

O'ZBEKISTON ALOA VA AXBOROTLASHTIRISH AGENTLIGI

TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI

Axborot texnologiyalari fakulteti
Axborotlashtirish va kutubxonashunoslik yo'nalishi

Himoyaga ruhsat
Kafedra mudiri

2010 y. «____» _____

ARM DIZAYN KO'RINISHLARINI TANLAB OLISH DASTURI

mavzusida

MALAKAVIY BITIRUV ISHI

Bitiruvchi _____	<u>Jurinov F. A.</u>
imzo	
Rahbar _____	<u>Payozov M. M.</u>
imzo	
Tarizchi _____	<u>Nosirov I. N.</u>
imzo	
HFH _____	<u>Qodirov F. M.</u>
maslaxatchi imzo	

TOSHKENT-2010

**O'ZBEKISTON ALOA VA AXBOROTLASHTIRISH AGENTLIGI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI**

**Axborot texnologiyalari fakulteti Axborot texnologiyalari kafedrası
5320200 Axborotlashtirish va kutubxonashunoslik yo`nalishi**

TASDIQLAYMAN

Kafedra mudiri

2010 yil « _____ » _____

Talaba JURINOV FARHOD Abduvahobovich
(familiyasi, ismi, otasining ismi)

“ARM dizayn ko`rinishlarini tanlab olish dasturi” mavzuidagi malaka-viy bitiruv ishiga

T O P S H I R I Q

1. Mavzu universitetning 2010 yil «12» «yanvar» dagi 26-07 sonli buyruq bilan tasdilangan
2. Ishni himoyaga topshirish muddati: 2010 yil 25 may
3. Ishga oid dastlabki ma`lumotlar: Mavzu tasdiqlandi, reja tuzib olindi va materiallar yig`ishga kirishildi.
4. Hisoblash-tushuntirish yozmalar mazmuni (ishlab chikadigan masalalar ro`yxati), Axborot resurs-markazlari va kutubhonalarni uch ulchovli grafikada ko`rinishlarini yaratish texnologiyalari va ularni ma`lumotlar bazasidan kerakligini idirib topish algoritmlari hamda dasturiy vositalarini o`rganish va ularni tahlil ilish. ARM dizayn ko`rinishlarini tanlab olish dasturining funksional strukturalarini ishlab chiish va ularni o`llash.
5. Grafik materiallar ro`yxati

-
6. Topshiriq berilgan sana: 2010 yil 15 yanvar

Rahbar

imzo

Topshiriq oldim

imzo

7. Ishning ayrim bo`limlari bo'yicha maslaxatchilar:

Bo`lim nomi	Maslahatchi	Imzo, sana	
		Topshiri berildi	Topshiri olindi
1. ARM dizaynini uch ulchovli grafikada yaratish afzalliklari va ularni tahlili 2. ARM dizaynini uchun bo'lgan idiruv dasturining funksional strukturasi, masalalar shemasi va malumotlarni kiritish algoritmlari 3. ARM dizayni uchun bo'lgan dasturning ishlatish qo'llanmasi	Payozov M.		

8.

Ishni bajarish grafigi

T/r	Ish bo`limlari nomi	Bajarish muddati	Rahbar (maslahatchi) imzosi
1.	ARM Dizayni ko'rinishlarini yaratadigan dasturiy vositalar.	2010 yil yanvar.	
2.	Ma'lumotlar ombori haqida tushuncha	2010 yil fevral-mart.	
3.	ARM dizayni uchun bo'lgan dasturning tuzilishi.	2010 yil aprel.	
4.	Mexnatni muxofaza qilish va texnika xavfsizligi	2010 yil may.	

Bitiruvchi _____
imzo

2010 yil "_____"

Rahbar _____
imzo

2010 yil "_____"

Ushbu BMIda - axborot resurs-markazlari hamda kutubhonashunoslik sohasida ishlovchi rahbar va xodimlar uchun ularning talabi asosida, maxsus dastur yordamida yangi ko'rinishdagi ARMLarni tanlab olishga yordam beradigan, dasturiy ta'minot haqida so'z yuritiladi. Unda sohaga oid tushunchalar, ARM ko'rinishlari uchun uch o'lchovli grafika xaida tushuncha, ularning afzalliklari o'rganilgan va tahlil ilingan hamda ARM dizaynini tanlab oladigan dasturiy ta'minotni yaratish texnologiyalari, ularni ko'rinishlari aks ettirilgan. ARM dizaynini tanlab oladigan dastur yaratilgan va ularni o'llash yo'llari ko'rsatilgan.

Настоящая выпускная работа посвящена для выбора сотрудников ИРЦ нового дизайна ИРЦ и поможет в выборе по потребительским спросом отражать в электронной странице внешний вид ИРЦ. В нее включены основные понятия специальности, понятие о трехмерной графике в ИРЦ, изучены ихние необходимости и анализировались и также включена программа подбора дизайна ИРЦ. Разработан программный пакет для подбора дизайна ИРЦ.

The present final work is devoted for a choice of employees Is IRC of new design IRC and will help with a choice on a consumer demand to reflect in electronic page appearance IRC. The basic concepts of a speciality, concept are included In it about the three-dimensional drawing in IRC, their necessities are studied and analyzed and also the program of selection of design IRC is included. The software package is developed for selection of design IRC.

Mundarija

1.	Qisartmalar va terminlarni ro'yhati
2.	Kirish
3.	1-Bob. ARM DIZAYNI UCHUN UCH O'LCHOVLI GRAFIK DASTUR TAHLILLARI 1.1. ARM dizaynini uch ulchovli grafikada yaratish afzalliklari va ularni tahlili. 1.2. ARM Dizayni ko'rinishlarini yaratadigan dasturiy vositalar. Xulosa:
4.	2-Bob. ARM DIZAYNI UCHUN BO'LGAN IDIRUV DASTURINING TAHLILI 2.1. ARM dizaynini uchun bo'lgan qidiruv dasturining funksional strukturasi, masalalar sxemasi va ma'lumotlarni kiritish algoritmlari 2.2. Ma'lumotlar ombari haqida tushuncha Xulosa:
	3-Bob. ARM DIZAYNI UCHUN BO'LGAN DASTURNI ISHGA TUSHIRISH 3.1. ARM dizayni uchun bo'lgan dasturning tuzilishi 3.2. ARM dizayni uchun bo'lgan dasturni ishlatish qo'llanmasi Hulosa
5.	4-Bob. MEHNAT MUHOFAZASI VATEHNIKA HAVFSIZLIGI 4.1 Shovqin va uning inson tanasiga ta'siri 4.2 Yong'in xavfsizligi Xulosa:
6.	Xulosa
7.	Foydalanilgan adabiyotlar ro'yhati
8.	Ilova

QISARTMALAR VA TERMINLAR RO'YHATI

EK – Elektron Kutubxona

ARM – Axborot Resurs Markazi

AKM – Axborot Kutubxona Markazi

WWW – Word Wide Web

HTML – Hyper Text Markup Language (Gipermatnni Belgilash Tili)

DOM – The Document Object Model

API – Application Programming Interface

PHP – Personal Home Page & Hypertext Processor

AT – Axborot qidiruv Tarmog'i

KIRISH

Zamonaviy texnologiyalardan foydalangan holda Ahborot-resurs markazlari hamda kutubxonalarning tashi va ichki ko'rinishlari yanada ham ko'rkam bo'lishi, aysidir ma'noda bir tarafdin kitobhonlarning sonini oshishiga, aholi atlamining ko'pchiliklari savodhonligi hamda ularning dunyoqarashi oshishiga olib keladi.

Respublikamiz prezidenti I.A. Karimovning 2006 yil 20 - Iyundagi "Respublika aholisini kutubxona axborot bilan ta'minlashni tashkil etish to'g'risida"gi arori ijrosi-ni ta'minlash, kutubxona jarayonlarini avtomatlashtirish sohasida faoliyat ko'rsatayotgan mutaxassislar oldiga bir ator hal ilinishi o'ta dolzarb masalalarni o'ymoda. Toshkent ahborot tehnologiyalar universiteti, ahborot tehnologiyalar fakulteti, ahborot tehnologiyalar kafedراس, Ahborotlashtirish va kutubhonashunoslik yo'nalishi professor o'ituvchilari, bu aror ijrosini bajarishda faol ishtirok etib kelmodalar. Jumladan, ARM lar va kutubhonashunoslikda, hamkorlikda elektron bibliografik resurslar yaratish va ulardan birgalikda foydalanishni yo'lga o'yish g'oyasiga katta ahamiyat berilmoda.

Hozirgi kunda global axborot tarmolarining keng taralib borishi bilan axborotlarga alohida munosabat hamda ulardan foydalanishga bo'lgan intilishlar ko'lami ham kengayib boradi.

Axborotlar oimining kundan-kunga ortib borishi, turli shakldagi axborotlarni yaratish, salash, idirish va ularni iste'molchilarga yetkazib berish kundan kunga murakkablashib bormoda. Axborotlardan foydalanuvchilar kerakli axborotlarni tezlik bilan idirib topish va o'zlariga mabul shaklda bu axborotlarga ega bo'lishni xohlaydilar.

Bugungi kundagi ARMLar va kutubhonalarining holatini ko'rkam shklga keltirish, bu dargohlarda yanada kitobhonlarning sonini olib kelishiga sabab buladi. Ular uchun ulay va ko'rkam dizayn, ularning shu dargihga o'rganib olishlari va u yerda yaratilayotgan darsliklarni, o'uv o'llanmalarni, ma'ruzalar matnlarini va bosha ko'plab vositalarni elektron shakldagi ma'lumotlardan foydalanishlariga yanada g'ayrat bag'ishlaydi. Shu sababli ham ARMLar va kutubxonalarni ichki va tashi tarafdin dizaynini yaratish kutubxonachilik sohasidagi o'z echimini kutib turgan dolzarb masalalar sirasiga kiradi. Bu hol tanlangan **mavzuning dolzarbligini** ko'rsatmoda.

Bitiruv malakaviy ishining masadi: ARM hodimlari va shu soha mutahassislarga ulay buladigan, ularga mos keladigan ARM dizayni ko'rinishlarini avtomatik tarzda topishni amalga oshiruvchi dasturiy ta'minotni yaratish.

Tadiot vazifalari:

Bitiruv malakaviy ishida quyidagi vazifalar amalga oshiriladi:

1. ARM dizaynlarini yaratadigan mavjud grafik dasturlarni urganib chiish.
2. ARM dizaynini idirib topadigan dasturiy ta'minotni yaratish.
3. Mavzuga oid tushuncha va atamalar o'rganib chiildi.
4. Ma'lumotlar ombori haqida tushunchaga ega bo'lish.
5. Yaratilgan dasturiy ta'minotning amalda o'llashni o'rganish.

Tadqiqot ob’ekti va predmeti: Tadiot ob’ekti sifatida Ahborot resurs markazi olindi. Tadiot predmeti sifatida esa, dizayn yaratish bo’yicha uch o’lchovli grafika turlaridan hisoblangan „Maya“ dasturi olingan bo’lib, idiruv dasturi sifatida C# dasturi olingan.

Tadiot natijalarining amaliy ahamiyati - Tadiot natijalaridan juda ko’plab dizaynga ega bo’lgan, uch o’lchovli grafik dasturda yaratilgan tayyor xoldagi axborot resurs markazlariga ega bo’lish mumkin. Tadiot natijasida yaratiladigan dastur xaidagi uslubiy qo’llanmalar, to’li matnli ma’lumotlar bazasi universitet talabalari hamda professor o’ituvchilari uchun juda muhim axborot manbai bo’ladi.

Tadiot natijasida yaratilgan ARM dizayn ko’rinishlarini tanlab olish dasturidan respublikamizning barcha oliy o’uv yurtlari talabalari va professor-o’ituvchilari, xamda axborot resurs markazlari va kutubxonachilar foydalanishlari mumkin.

Ishning axborot ta’minoti - Bitiruv malaka ishining Elektron kutubxona, o’uv qo’llanmalar, ARM lardagi axborot jarayonlarini avtomatlashtirish bo’yicha adabiyotlar, internet, 3d studio max, Maya, Autocad C dasturlash tili, ma’lumotlar ombori xaidagi adabiyotlardan foydalanilgan. Toshkent axborot texnologiyalari universiteti virtual kutubxonasi fondi, Alisher Navoiy nomidagi O'zbekiston Milliy kutubxonasi bo'limlari tomonidan tayyorlangan uslubiy bibliografik qo’llanmalar asosida shakllantirildi.

Битирув малакавий иш кириш, То’rt боб, хулосалар ва фойдаланилган адабиётлардан иборат.

Birinchi bob ahborot resurs markazlari va kutubhonalarning chiroyli ko’rinishga olib keladigan dasturiy vositalar, ularning turlari, va tahlillari, hamda ARM dizayn ko’rinishlarini rahbar hodimlar va foydalanuvchilar talabiga mos ravishda tanlab oladigan dasturni yaratadigan mavjud dasturlar haqida gap boradi.

Ikkinchi bob ARM dizaynini uchun bo’lgan idiruv dasturining funksional strukturasi, masalalar shemasi va malumotlarni kiritish algoritmlari hamda ma’lumotlar ombari haqida tushunchalar bilan o’z nihoyasiga yetadi.

Uchinchi bob bitiruv malaka ishidagi asosiy masadlardan biriga to’htaladi. Unda ARM dizayni uchun bo’lgan dasturning tuzilishi, ko’rinishi hamda ARM dizayni uchun bo’lgan dasturni ishlatish qo’llanmasi bilan yakunlanadi.

1-BOB. ARM DIZAYNI UCHUN UCH O'LCHOVLI GRAFIK DASTUR TAHLILLARI

1.1. ARM dizaynini uch ulchovli grafikada yaratish afzalliklari va ularni tahlili.

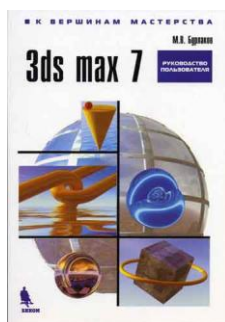
O'zbekiston respublikasi mustaillikka erishgandan so'ng, o'z yo'lidan yura boshladi. Prezident I.A. Karimov eng avvalo e'tiborni yoshlarga aratdi. Ularni yurish-turishi, bilim olishini doimiy nazorat ostiga oldi. Jumladan ta'limni isloh ilish, uni rivojlantirish masadida 1992-yil "Ta'lim to'g'risida" gi onun abul ilinishi bunga yaol misol bo'la oladi. Bu onun ishlab chiilgandan so'ng joylarda katta ishlar amalga oshirildi. Keyinchalik, ya'ni 1997-yil 29-avgustda ushbu onunning yangi taxriri "Kadrlar tayyorlash milliy dasturi" ishlab chiildi va president I.A.Karimovning 1997-yil 6-oktyabrida "Ta'lim tarbiya va kadrlar tayyorlash tizimini tubdan isloh ilish, barkamol avlodni voyaga etkazish" to'g'risidagi farmoniga binoan ta'limni III bosich asosida isloh ilish ko'zda tutilgan. Shu bosichlarning so'nggisida "ta'limni kompyuterlashtirish" jumlasida uchraydi. Shundan bo'lsa kerak hozirga kelib bu sohada yurtimizda juda ko'p ishlar amalga oshirildi. Yurtimizdagi deyarli barcha institut va universitetlarning hatto maktab va kollejlarning internet tarmog'idan bemalol foydalana olish imkoniyati borligi va buning yordamida o'z dargohlarining web saytlariga ega bo'lishi, elektron darsliklarning yaratilishi, dars o'tish uslublarining takomillashuvi, kompyuter va bosha zamonaviy texnikalardan foydalangan holda fan mavzularining yana ham boy mazmunda yoritilishi ta'lim sohasida o'lga kiritilgan yutulardan biridir. Aynisa darslarni zamonaviy dars o'tish uslublaridan biri multimediali ko'rinishida o'tilishi mavzularni tinglovchilar tomonidan yaxshi o'zlashtirilishiga katta turtki bo'lmoda.

Jahon miyosida nazar soladigan bo'lsak Kompyuter Grafikasi va Elektron Dizayn juda ham keng darajada rivojlangan. Ularni amalda allaachondan buyon o'llab kelishmoda. Lekin bu yaratilayotgan barcha dasturlar, roliklar, dizaynlarning asosiy ismi kinofilmlar, multafilmlar, web-saytlar uchun yaratilmoda. Maxsus saytlarda fanlarga mavzulashtirilgan multimediali darsliklarning kamligi, borlarini ham ani bir fanni mavzularini to'la yoritib bera olmasligi ta'lim sohasiga yanada ko'pro e'tibor berish lozimligini e'tirof etadi. Lekin elektron darsliklar ichiga kiritilgan matnli yozuvlar, rasmlardan hozirda grafikaning eng zamonaviy yo'nalishi bo'lgan roliklarning yaratilishi saytlarga va shunga tegishli bo'lgan narsalarga iziishni oshiradi. To'g'ri bu sohada ko'p ishlar amalga oshirilgan. Darsliklarning multimedia shaklida o'tilishi o'ituvchilarga bir ancha ulayliklar tug'diradi. Chunki, shu fanga mavzulashtirilgan ovozli dasturlar, rasmi ishlanmalar, harakatli yozuvlar (prizentatsiyalar), multiplikatsiya ko'rinishida yaratilgan filmlar, animatsiyalar bir tomondan o'ituvchi uchun engillik bo'lsa, ikkinchi tomondan o'uvchilar uchun iziarli va diatini o'ziga jalb iladi. Va mavzuni yaxshi o'zlashtirilishiga turtki bo'ladi. Ta'lim sohasining multimedia ko'rinishida o'tkazilishiga sabab ham shu. Chunonchi, o'uvchi televizorda o'yilayotgan multfilm va reklama roliklarini ko'rish o'rniga darsga boradi, bu esa o'z-o'zidan maktabga yanada iziishini oshiradi.

Animatsiyali rolik mavzuni har jihatdan o'rganib, keng amrab olishiga aramasdan uni ham kamchiliklari bor. Xususiyl holda olib araydigan bo'lsak, fizika va kimyo fanlari mavzulari uchun ishlangan roliklar har bir fizik, ximik jarayonni ko'z ilg'amas tomonlarigacha ko'rsatib bera oladi, tushuntirib bera oladi, bunga sabab tinglovchilarning ba'zi psixologik hususiyatlari e'tiborga olingan. Masalan: Harakatga e'tibor – Bu harakatlanmagan ob'ektga nisbatan harakatlangan ob'ektning tezro esda olishi. Bitta tasvirda 7 tadan 9 tagacha ob'ektlarni eslab olishi. Va bu tasvirda ob'ektlarni xossalari va ob'ektlar aro axborotni to'li amrab olishi, olaversa bu axborotni to'laligicha o'zgarmas holidan salab olishi. Lekin, rolik real emas o'uvchi bunda laboratoriya asboblarini ushlab ko'ra olmaydi, ya'ni bu barcha jarayonlar real fizik yoki ximik jarayonning aynan sun'iy ravishda yaratilgan formasi. Bu bir tomondan kamchilik deb atalsa, ikkinchi tomondan havfsizlik tushunchasiga borib taaladi. O'uvchi bunda laboratoriya ishini bajarish jarayonida asbobni buzib o'ymaydi, havfli laboratoriya ishi jarayonida bajaruvchiga xavf tug'dirmaydi. Eng muhimi alohida jihozlangan laboratoriya xonasini talab ilmaydi. olaversa darslarning bunday usulda o'tilishi zamon talablariga ham mos keladi. Ta'limni avtomatizatsiyalashga xizmat iladigan bunday dasturiy qo'llanmalar ko'p ulardan eng asosiysini keltirib o'tamiz: 3ds Max dasturiy qo'llanmasidir. Bu dasturlash vositasi 3 o'lchamli har anday animatsiyalar yaratish imkonini beradi.

Ahborot resurslarni dizaynini ana shunday dasturlar yordamida yaratish nihayotda ahamiyatlidir.

3d studio max bu - 3 o'lchamli grafika dasturi bo'lib, undan turli animatsiyalar ishlashda foydalaniladi. Unda uch o'lchamli filmlarni ishlash, ob'ektlarni yaratish, kooordinatalarni yasash va eng asosiysi uch o'lchamli dunyoga kirib borish imkonini beradi. Hozirda esa bu dastur yanada mukammallashib bormoda. Hozirgi vatda yaratilgan, yaratilayotgan reklama roliklari, multfilmlar va o'uv dasturlari shu dastur asosida yaratilmoda va "foydalanuvchilar" ga xavola etilmoda. 3ds max ni hozirda eng keng taralgan turlari 3ds max 4, 3ds max 5, 3ds max 7, 3ds max 8, 3ds max 9 nomlari bilan mashhur. Ularning ko'rinishi ham bir-biridan far iladi. Yani:



3ds max 7

Misollar keltirib o'tsak: "Final Fantasy", "Monstor Corporation", "Shrek", "Ice age" multfilmlari bu dastur imkoniyatlarini naadar kengligini ko'rsatib turibdi.



3ds max 8



3ds max 9

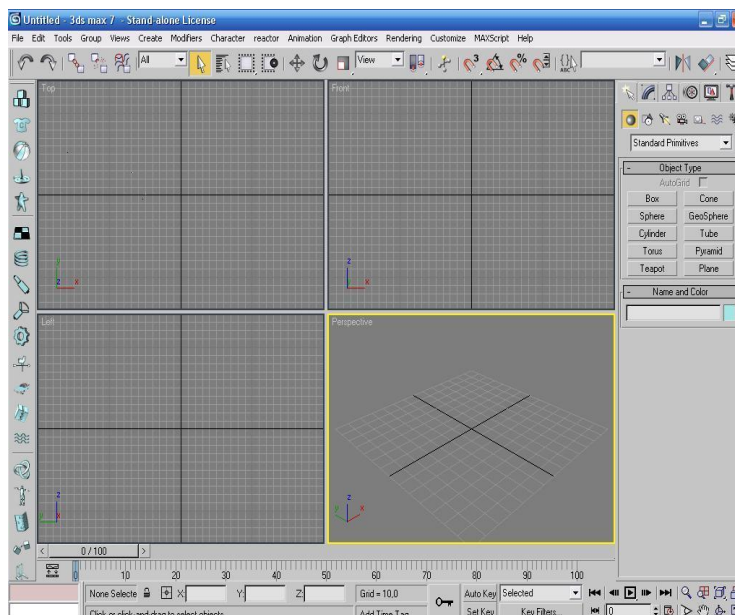
Bu dasturni ishga tushirish uchun Windows ishchi oynasidagi belgi orali kiriladi. So'ng uyidagi menyu: ishchi oyna hosil bo'ladi (1.1-rasm). Bu ishchi oynada: 4 ta ishchi oyna proeksiyalari mavjud.

Perspektive – asosiy ishchi oyna,

Front – tasvirni oradan ko'rsatuvchi ishchi oyna,

Left – tasvirni chap tomondan korsatuvchi ishchi oyna,

Top – tasvirni yuoridan ko'rsatuvchi ishchi oyna.



1.1-rasm.

Bu erda o'zgaruvchilar va buyrular umumiy ro'yhati berilgan (1.2-rasm). File, Edit, Tools, Group, Views, Create, Modifiers, Character, reactor, Animation, Graph Editor, Rendering, Customize, Maxscript, Help. Bosh menyu deyiladi.

File Edit Tools Group Views Create Modifiers Character reactor Animation Graph Editors Rendering Customize MAXScript Help

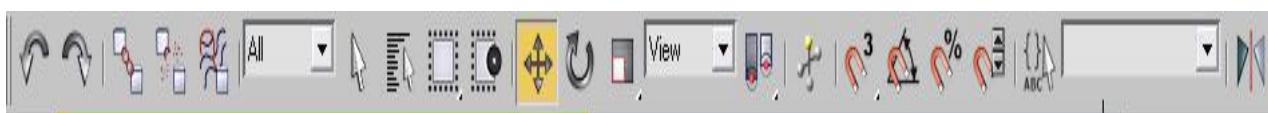
1.2-rasm.

Bosh menyuning keyin uskunalar doskasi joylashgan.

Bu panelda yaratiladigan ob'ektlar va ular bilan ishlasga mo'ljallangan operatorlar joylashgan (1.3-rasm). Bunda

1. Bajarilgan operatsiyani oraga aytarish (Undo).
2. Bajarilgan operatsiyaga aytish (Redo).
3. Uchinchi: Ob'ektga fizik hodisani bog'lash, yoki tanlash va bog'lash.
4. Tanlangan ob'ektni va bog'lanishni bekor ilish.
5. Space Warps fizik hodisalarni ob'ektlarga bog'lash uchun ishlatiladi.
6. Ob'ektni tanlash
7. Ishchi oynadagi barcha ob'ektlarni nomma-nom tanlash.
8. To'rtburchak, aylana... maydonli tanlash.
9. Tanlash filtri.

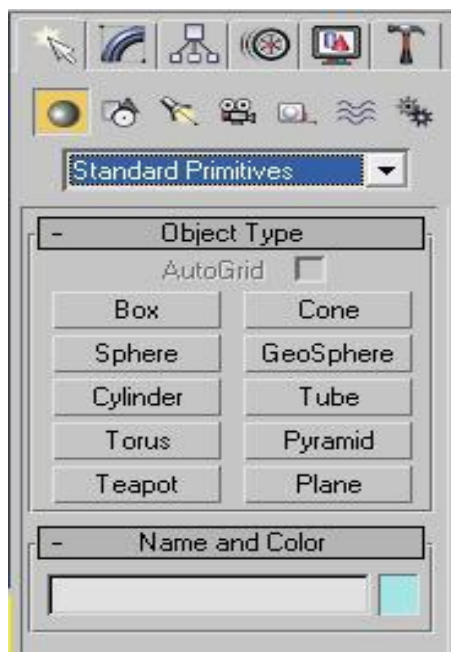
Va tanlash harakatlantirish, tanlash aylantirish, tanlash va o'lchovli masshtablash, koordinata sistemasi va bosha operatsiyalarni bajarish uchun uskunalar joylashgan.



1.3-rasm.

Asosiy ishchi oynani o'ng tomonida esa "Buyrular doskasi" joylashgan.

Birinchi atorda (1.4-rasm): Yaratish (создать), O'zgartirish (Изменить), Ierarxiya (иерархия), Harakat (Движение), Display (Дисплей), Hizmatlar (Сервис) buyrulari joylashgan. Ikkinchi atorda: aysi buyru tanlanishiga arab yaratish operatorlari joylashadi. Masalan: Yaratish operatorining create paneliga kiramiz. Bunda Standart primitives va unda yaratilishi mumkin bo'lgan ob'ektlar joylashgan. Unda nimalar yaratilishi uyidagi rasmda keltirilgan.



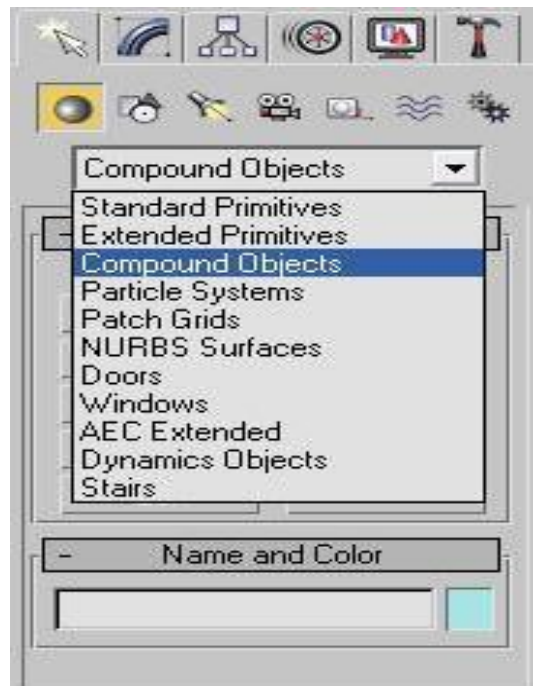
1.4-rasm.

Standart Primitives dan so'ng Extended primitives doskasi keladi (1.5-rasm). Bunda ham ko'pburchakli ob'ektlar yaratiladi va ulardan foydalaniladi.



1.5-rasm

Bu rasmda Create doskasidagi barcha buyrular keltirilgan (1.6-rasm). Bu erdagi barcha operatsiyalar ob'ekt yaratish. Ob'ektlarni hosil ilishda irish, bo'laklash, birlashtirish operatsiyalarni bajaruvchi funksiyalarni bajaradi.

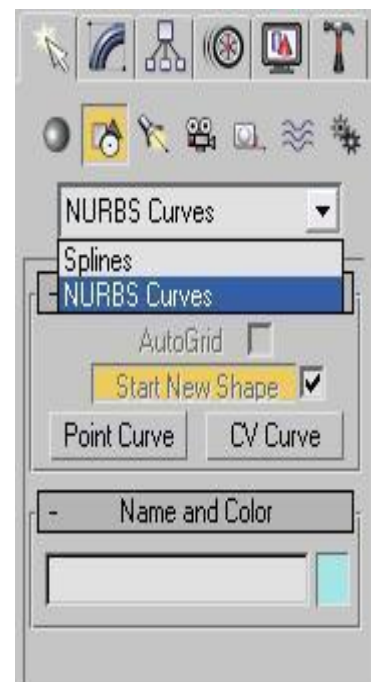


1.6-rasm.

Create operatoridan keyin SHAPES (1.7-a rasm) joylashgan shapes ikkiga: splines, NURBS curver bo'linadi. Spline: Line-chizi, Circle-aylana, arc-burchak, Ngon- n burchak, Text-yozuv, Section-ism, bo'lim, Rectangle-to'rtburchak, Ellipse- ellips, Donut-donut, star-yulduz, Helix-prujina. NURBS (1.7-b rasm) Curve: point curve va CV curve iyshi chizilar chizish uchun mo'ljallangan. Bu operatsiyalar yordamida har anday kerakli ob'ektni yasash mumkin.



1.7- a) rasm.



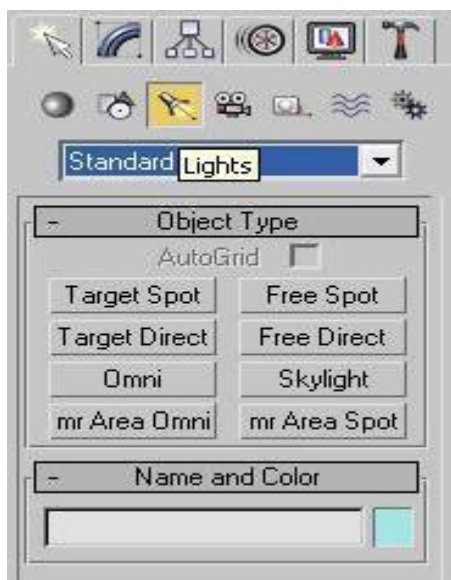
1.7- b) rasm

Shapes dan keyin Lights turadi (1.8-a,b rasm). Lights bu – chirolar bilan ishlashga mo'ljallangan. Bu panelda chiroqlarning sakkiz turi joylashgan ularning nomlarini esa uyidagi rasmdan bemalol bilib olish mumkin. Bulardan ba'zilarini sanab o'tamiz. Lights ikki turga bo'linadi: Standard va Photometric. Standard uyidagilardan tashkil topgan: Target spot, target direct, Omni, mr Area Omni, Free Spot, Free direct, Skylight, mr Area Spot.

Target spot - yorug'likni bir tomonga yo'naltirib beradi. Bunda asosan lazer yorug'lik manbalari yaratishda foydalaniladi.

Target direct – yorug'likni tekislikni ma'lum ismiga yo'naltirish uchun foydalaniladi. Bunda oddiy sharoitda lampochkalar yaratishda foydalaniladi.

Omni – yorug'likni har tomonga yo'naltiradi. Bunda asosan uyosh va shunga o'xshash ob'ektlarni yaratishda foydalaniladi. olgan yorug'lik manbalari ham huddi shunday ob'ektlarni yaratishda foydalaniladi. Faat ular doimo ozod harakat iladi. Photometric uyidagilardan tashkil topgan: Target Point, Target Linear, Target Area, Ies Sun, Free Point, Free Linear, Free Area, Ies Sky



1.8- a) rasm.



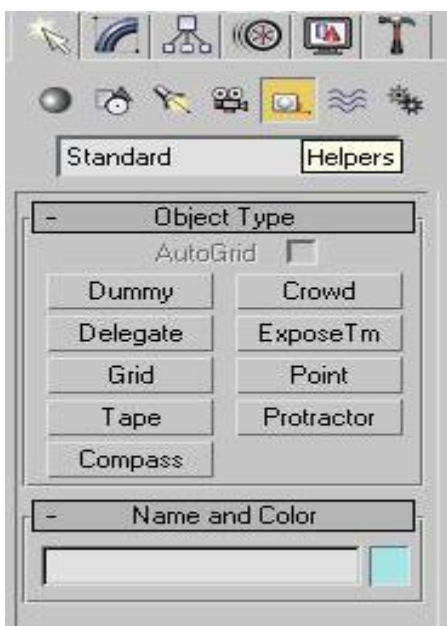
1.8- b) rasm.

Camera (1.9-rasm) Lights dan keyin keladi. Camera faat standartdan iborat bo'lib, unda Target va Free dan tashkil topgan. Target kamera olishni eng universal usuli bo'lib uni har anday sharoitda ishlatish mumkin. Va har anday ob'ektni istagan joydan kamera tushirish mumkin. Free kamera olishni cheklangan yoki, yo'naltirilgan usuli bo'lib, u faat o'rnatilgan joyidan turib suratga olish uchun mo'ljallangan. Uni faat klaviaturaning maxsus tugmalari yordamida kerakli yonalishlarga burish mumkin.

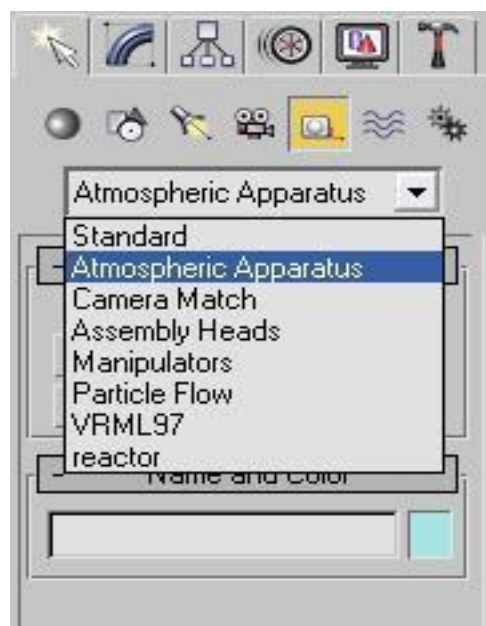


1.9- rasm.

Yordamchi (Helpers) dan keyin keladi (1.10-a,b rasm). Bu buyrular panelida deyarli barcha harakat operatsiyalari bajariladi. Bunda asosan oddiy operatsiyalar bilan uddalab bo'lmaydigan harakatlar amalga oshiriladi. Chunonchi, pardani harakatlantirish, kiyim kiydirish, iyin akrobat va shunga o'xshash amaliyotlar amalga oshiriladi.



1.10- a) rasm



1.10- b) rasm

Space Warps da ob'ektni o'zi bilan bog'li bo'lgan fizik jarayonlar amalga oshiriladi. Masalan biror ob'ektni portlatmochi yoki, parchalab tashlamochi bo'linsa Space Warps dan foydalanish mumkin. Space Warps ham o'zining bir necha turkumli parametrlariga ega: Forces, Deflectors, Geometric/deformable, Modifier-Based, Particles & Dynamics, reactor.

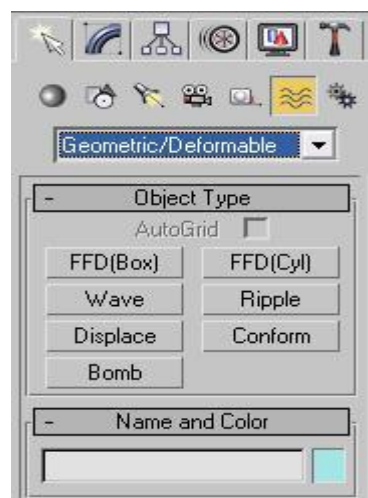


1.11- a) rasm



1.11- b) rasm

Space Warps (1.12-rasm) ning Geometric/ Deformable da asosiy funksiyalari joylashgan. FFD(box) da tayyor obektlarni turli shakllarga aylantirish mumkin. Wave esa to'linni harakarga keltiradi. Ya'ni uzuncho shakldagi to'linni hosil ilishda foydalaniladi. Bomb da esa biror ob'ektni parchalab tashlash uchun ishlatiladi. FFD(Cyl) da esa silindrsimon ob'ektlarni turli shakllarga keltirish uchun foydalaniladi. Ripple da aylanasimon to'linlar hosil ilish uchun o'llaniladi



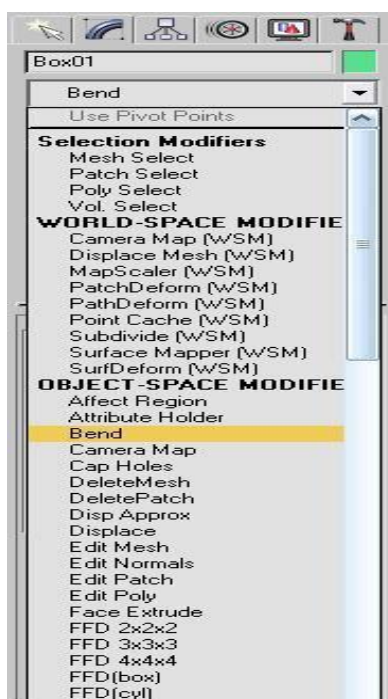
1.12-rasm.

Systems (1.13-rasm)da asosan skeletlar va suyaklar bilan ishlashga mo'ljallangan. U Bones, sunlight, Biped, ring array, daylight lardan tashkil topgan. Bones(suyak) lar bilan ishlashga mo'ljallangan bo'lib, unda turli xil hayvonlarni skelet maketlarini yaratishga mo'ljallangan. Sunlight uyoshning chiishi va botishiga mo'ljallangan yorug'likni o'zgarishiga mo'ljallangan. Ya'ni bunda Shar, Shimol, Janub, G'arb koordinatalari hisobga olingan. Biped odam skeletini tayyor xolda yaratilgan shaklidir. Unda odamning har bir suyak skeleti mukammal va mos ravishda hech anday oshicha bo'g'imlarsiz yaratilgan. Daylight uyoshning kun davomidagi harakat yo'nalishi ifodalangan.



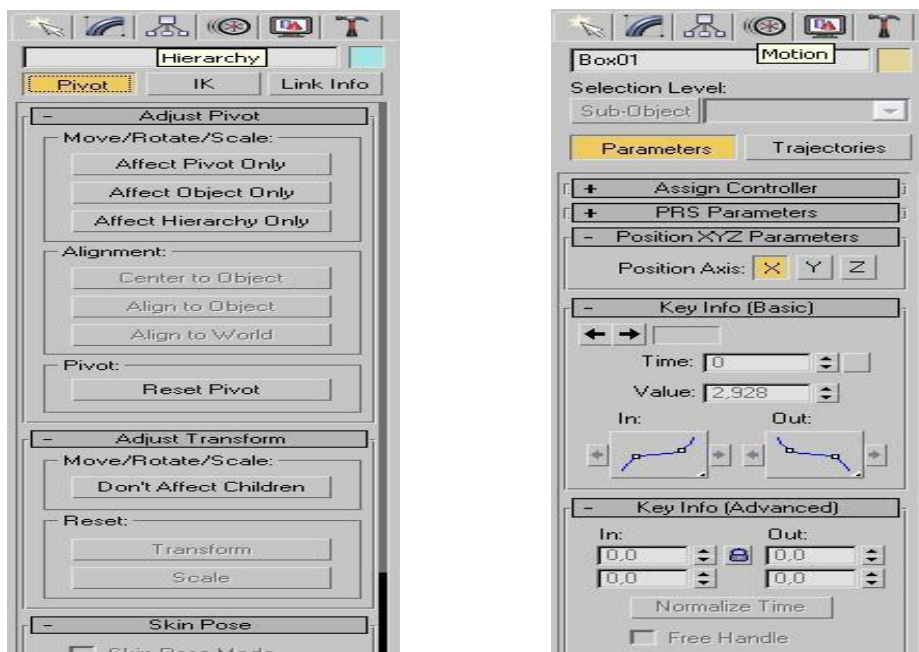
1.13-rasm.

Modfierda (1.14-rasm) ob'ektlarni o'zgartirish amallari bajariladi. Masalan: Bend ob'ektlarni egish uchun xizmat iladi. Lathe chizilarni radius bo'yicha aylantirish uchun xizmat iladi. Xullas bundagi har bir komandalar ob'ektni ma'lum bir masadda, o'z hususiyatiga suyangan holda o'zgartiradi.



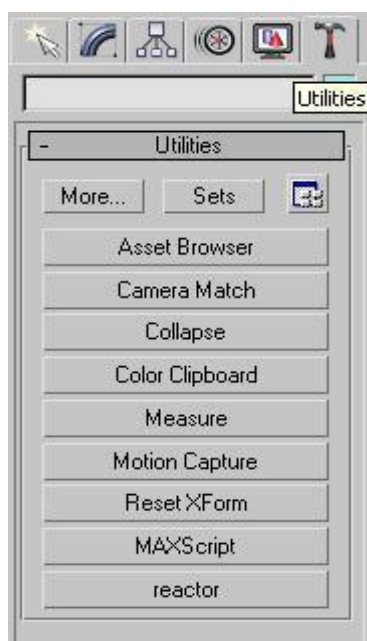
1.14-rasm.

Hierachy (1.15-rasm) iyin harakatlarni amalga oshirishda ma'lum bir o'shimcha imkoniyatlar beradi. Harakat ierarhiyasini ko'rsatib beradi. Motion ob'ektlarni ma'lum harakatlar bo'yicha harakatlantirish imkoniyatini beruvchi funksiya.



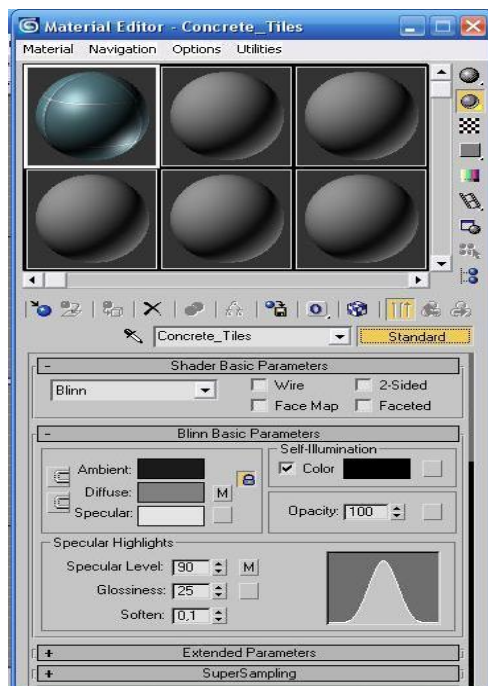
1.15-rasm.

Uskunalar panelida (1.16-rasm) o'shimcha imkoniyatlar beradigan funksiyalar joylashgan.



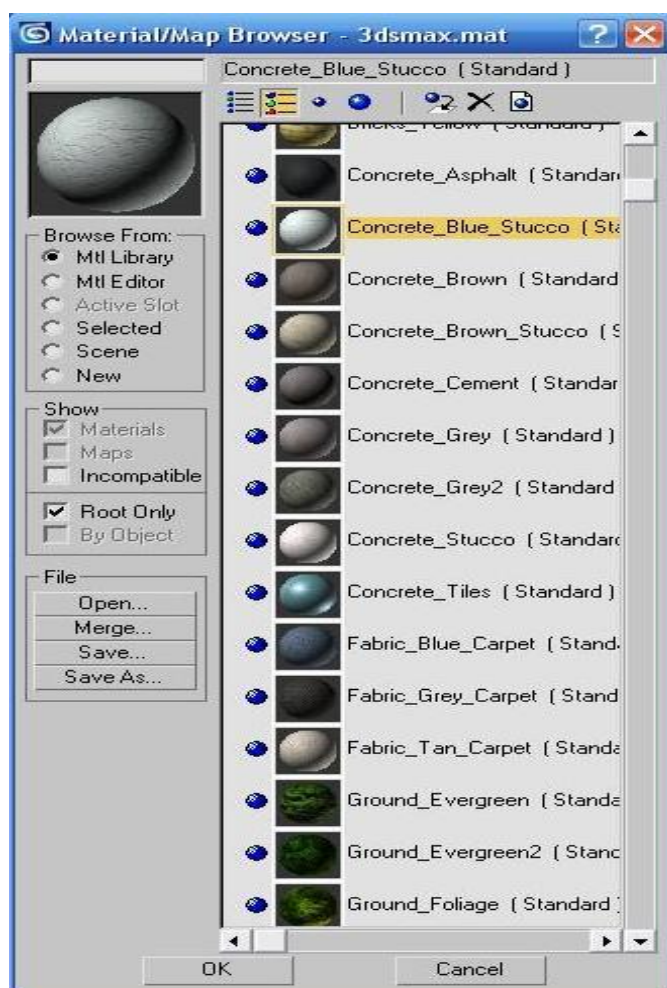
1.16-rasm.

Ob'ekt tayyor bo'lgandan so'ng uni xayotiy ob'ektlarga o'xshatish uchun unga material berish kerak bo'ladi. Huddi shu vazifani bajarish uchun Material Editor dan foydalaniladi (1.17-rasm). Material Editor ni ishchi oynaga chiarish uchun "M" tugmasini bosish lozim bo'ladi. "M" tugmasi bosilgandan so'ng ishchi oynada uyidagi rasm hosil bo'ladi.



1.17-rasm.

Bunda yasalgan ob’ektga materialni bog’lash uchun bo’sh material tanlanadi va “Standard” ga kiriladi. Bunda uyidagi rasm hosil bo’ladi.

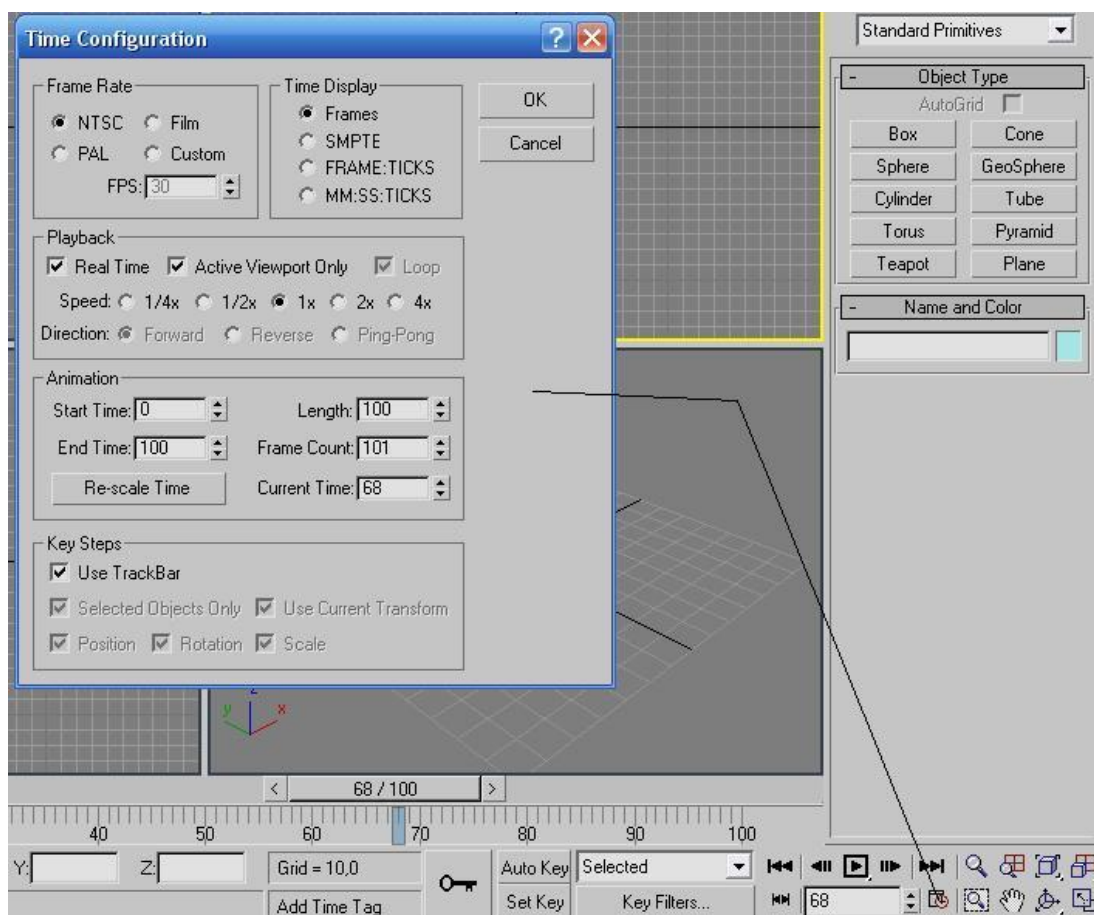


1.18-rasm.

Bu “Material/Map Browser” deb nomlanadi. Bunda “New” belgilangan bo’ladi. Bu belgini “Mtl Library” ga o’tkazamiz. Bu oynada materiallar o’z nomlari bilan ko’rsatiladi. Va kerakli materialni tanlashimiz, tanlaganimizdan so’ng, uni ob’ektga surib borib bog’lahimiz mumkin. Material Editor da asosiy materiallar va ularni yana ham o’zgartirish mumkin bo’lgan funksiyalari joylashgan.

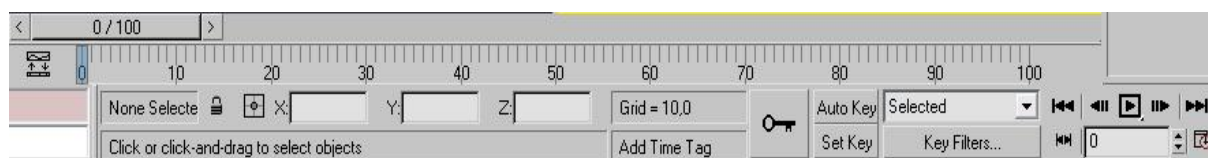
Bunda foydalanuvchi o’ziga kerakli bo’lgan materialni tayyorlab olishi va o’z yaratgan ob’ektiga bog’lashi mumkin.

Tayyorlangan ob’ektni harakatlantirib undan biror-bir rolik yaratishda albatta vat bilan hisoblashishga to’g’ri keladi. Vatni cho’zish yoki isartirish uchun Time Configuration ga murojat ilinadi (1.19-rasm). Time Configuration da Animation bo’limiga kiriladi. Va undan start time va end time ning iymatlari o’zgartiriladi.



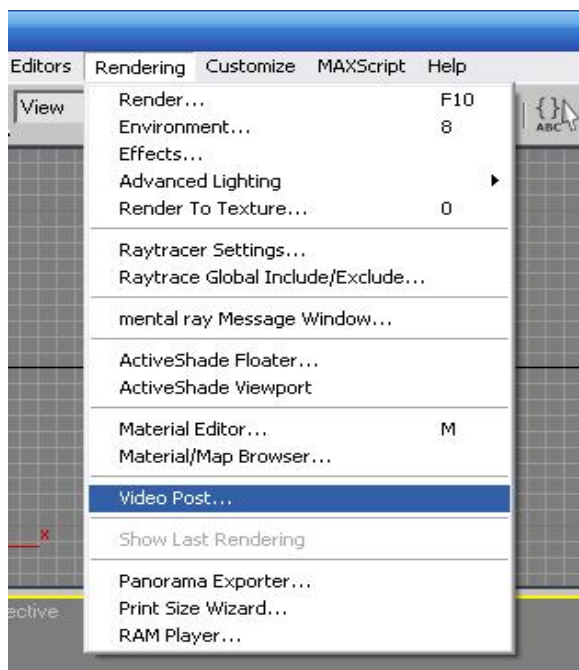
1.19-rasm.

Ob’ektni harakatlantirish uchun uyidagi rasmdan “auto key” bosiladi (1.20-rasm) va kerakli kadr tanlanib (0-100) , ob’ekt harakatga keltiriladi. Agar ishchi oynada bir necha ob’ektlarni harakatlantirish lozim bo’lsa va ularni harakatlantirish har xil bo’lsa kerakli ob’ektni harakatini cheklash uchun “set key” bosiladi va “Kalit” tugmasi bosiladi. Bu bilan shu ob’ektga kalit beriladi.



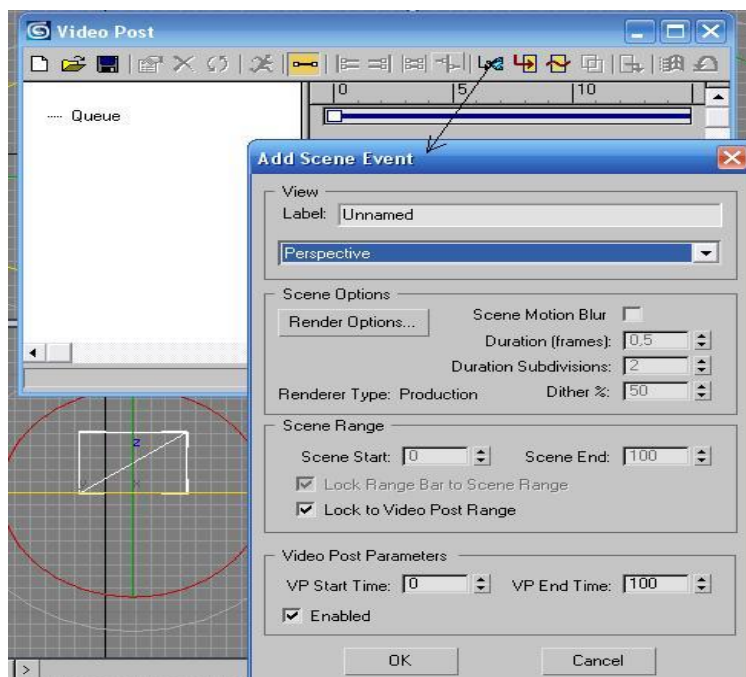
1.20-rasm

Tayyor bo'lgan ob'ektni harakatlantirib, unga kerakli bo'lgan barcha ishlarni bajarib bo'lgandan keyin uni "rolik" ko'rinishida salash uchun "Render" ilish kerak (1.21-rasm). Bosh Menyuda joylashgan Rendering bosiladi va undan Video Post... tanlanib bosiladi. Render ning asosiy vazifasi, tayyor bo'lgan rasmlarni tez suratga olishdir. Bu uyidagi rasmda keltirilgan.



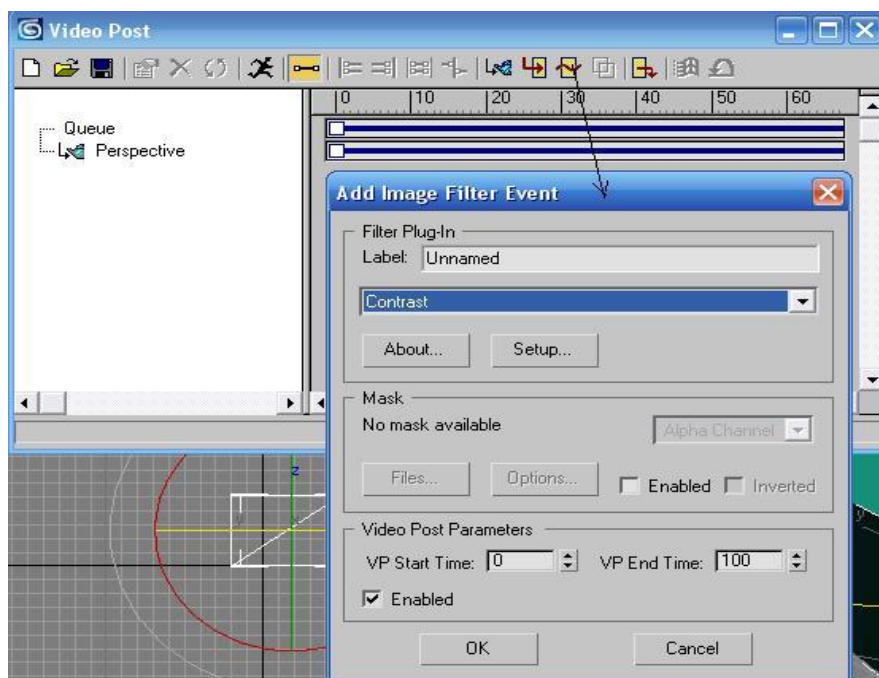
1.21-rasm.

Video Post ga kirgandan so'ng bieinchi bo'lib Add Scene Event tugmasi bosiladi (1.22-rasm). Undan aysi ishchi oyna (perspective, top front, left) ni yoki, aysi kamerani suratga olish lozimligi tanlanadi. Va yana kerakli amallar bajariladi. So'ngra OK tugmasi bosiladi.



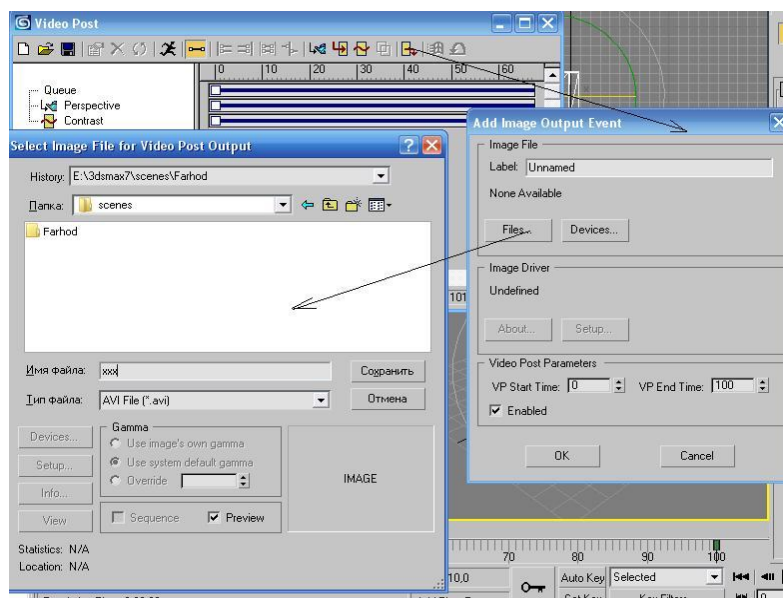
1.22-rasm.

Add Scene Event tugmasidan keyin Add Image Filter Event tugmasi bosiladi (1.23-rasm). Bunda suratga olishning o'ziga xos usullari ko'rsatilgan Contrast, Focus, glow shular jumlasidandir.



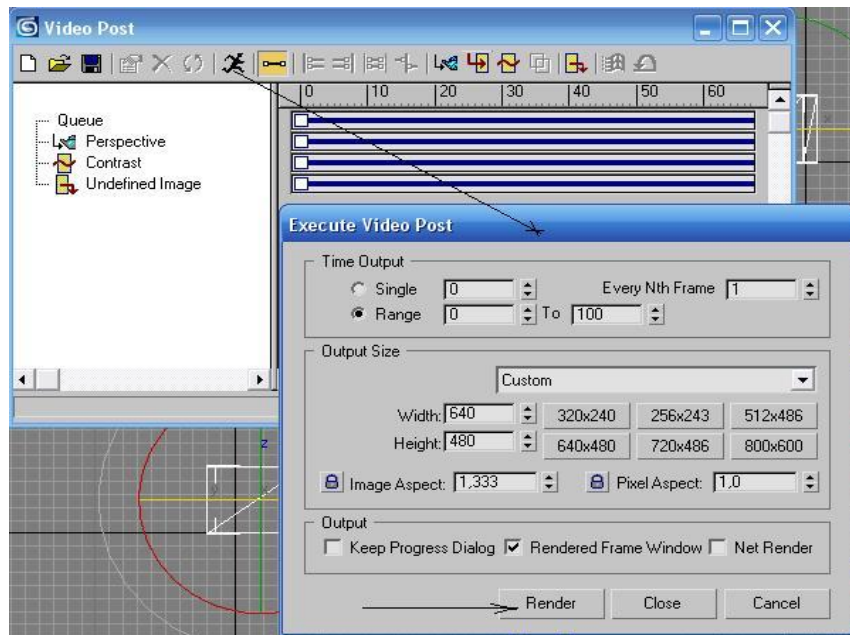
1.23-rasm.

Add Image Filter Event dan so'ng Add Image Output Event tugmasi bosiladi (1.24-rasm). Bu erda suratga olinayotgan rolikni aerga salash lozimligi ko'rsatiladi. Ya'ni Add Image Output Event da Files tugmasi bosiladi. Rolikni salanishi lozim bo'lgan joy ko'rsatiladi. So'ngra Fayl nomi va rolikni salanish formati ko'rsatiladi(avi.) va SAVE tugmasi bosiladi.



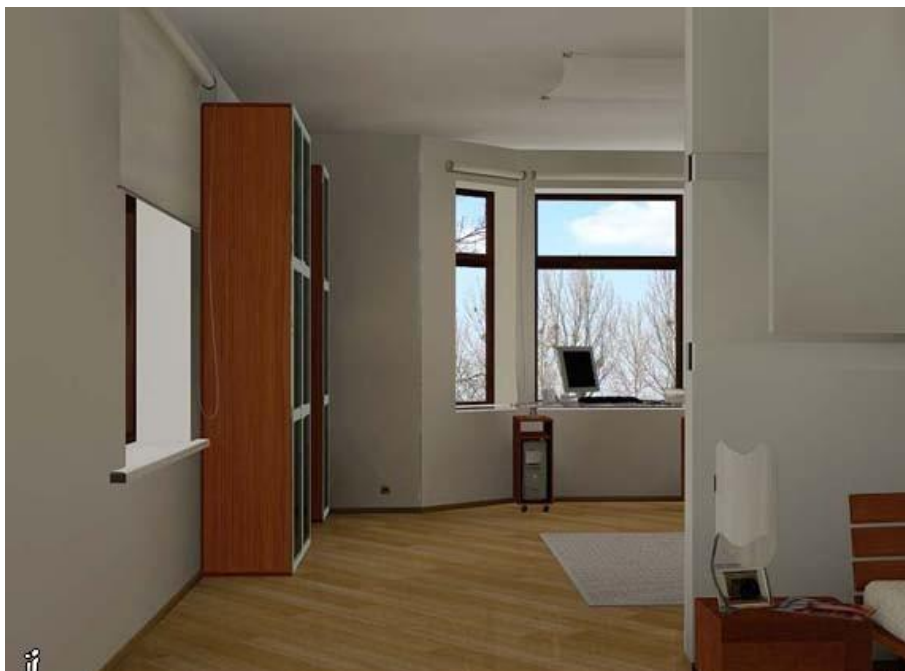
1.24-rasm.

Nihoyat hamma ishlarni bajarib bo'lgandan so'ng Execute Video Post tugmasi bosiladi va rolikni aysi o'lchamda suratga olish ko'rsatiladi, Pixel Aspect iymati kiritiladi. Va Render tugmasi bosiladi (1.25-rasm). Bu tugma bosilgandan so'ng ishchi oynada suratga olish ishlari boshlanadi.



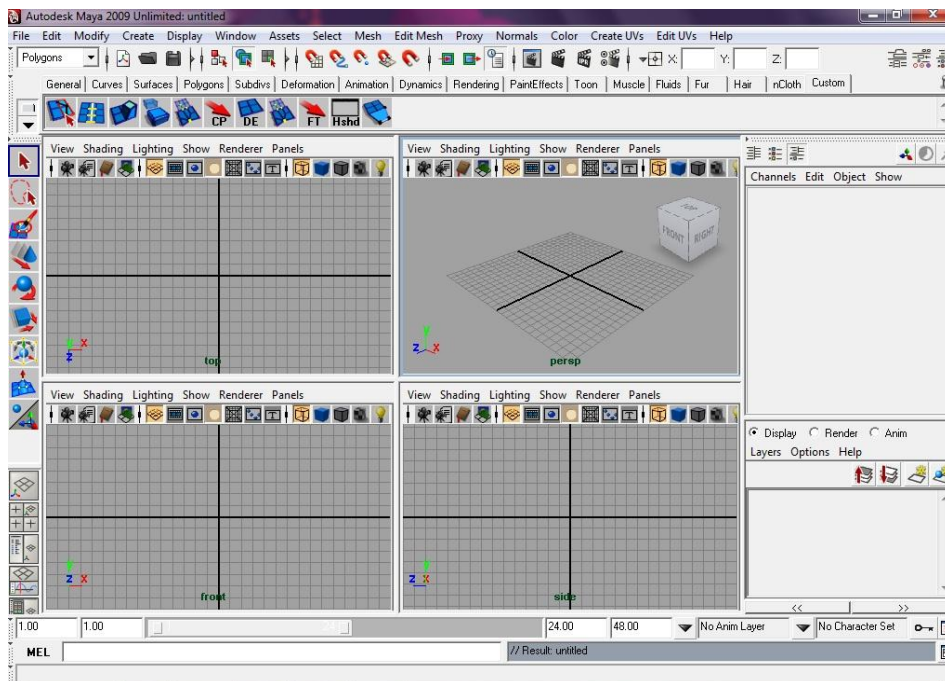
1.25-rasm.

Uch o'lchovli grafik dasturda yaratilgan ob'ektlar ko'rinishlari nihoyatda go'zal ko'rinishda bo'ladi.

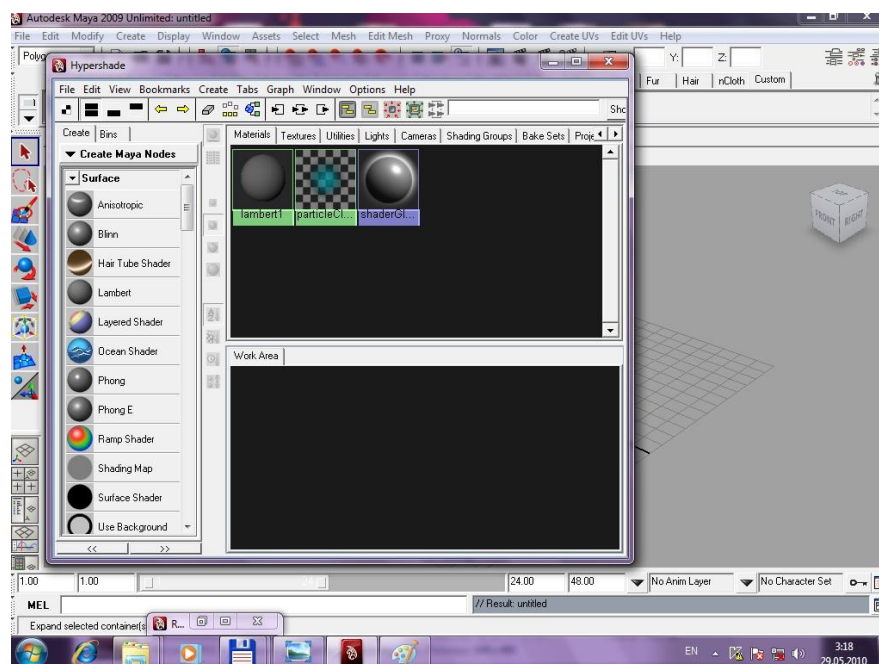


1.26-rasm ARM ning ichki ko'rinishi.

Keyingi uch o'lchovli grafik dasturlardan biri bu Mayadir. Maya ham biron ob'ektlar uchun kuchli dizayn chiarishda muhim ahamiyat kasb etadigan dasturlardan hisoblanadi. Uning ko'rinishi uyidagicha.



1.27-rasm. Mayaning ko'rinishi



1.28-rasm. Mayaning ko'rinishi



1.29-

rasm. Maya yordamida tayyorlangan ARM ning tashi ko'rinishi

Yana bir uch o'lchovli grafikani o'llab uvvatlovchi dasturlardan biri bu AutoCAD dir.

Bu dastur ham Autodesk kompaniyasi tomondan yaratilgan bo'lib u ikki hamda uch o'lchovli grafikani o'llab uvvatlaydi. Kompaniya bu dasturni 1982 yildan beri ishlaib kelmoda.

AutoCAD inglizcha - Computer-Aided Design so'zdan olingan bo'lib, Avtomatlashtirilgan loyiha ishlari uchun mo'ljallangandir.

Ahborot resurs markazlari hamda kutubhonalarni dizaynlari uchun bu dastur ham muhim ahamiyat kasb etadi.

Uning ko'rinishi uyidagicha.



1.30-rasm. AutoCAD ning ko'rinishi

1.2. ARM Dizayni ko'rinishlarini yaratadigan dasturiy vositalar.

Biz Informatika fanidan turli darajadagi test savollarini tuzish uchun Microsoft firmasining Visual Studio dasturining C# (Si sharp) dasturlash tilidan hamda MICROSOFT SL SERVER 2005 (Structured uery Language Server, odatda "sikvel" deyiladi) ma'nosi Tarkiblangan so'rovlar tildan foydalanamiz.

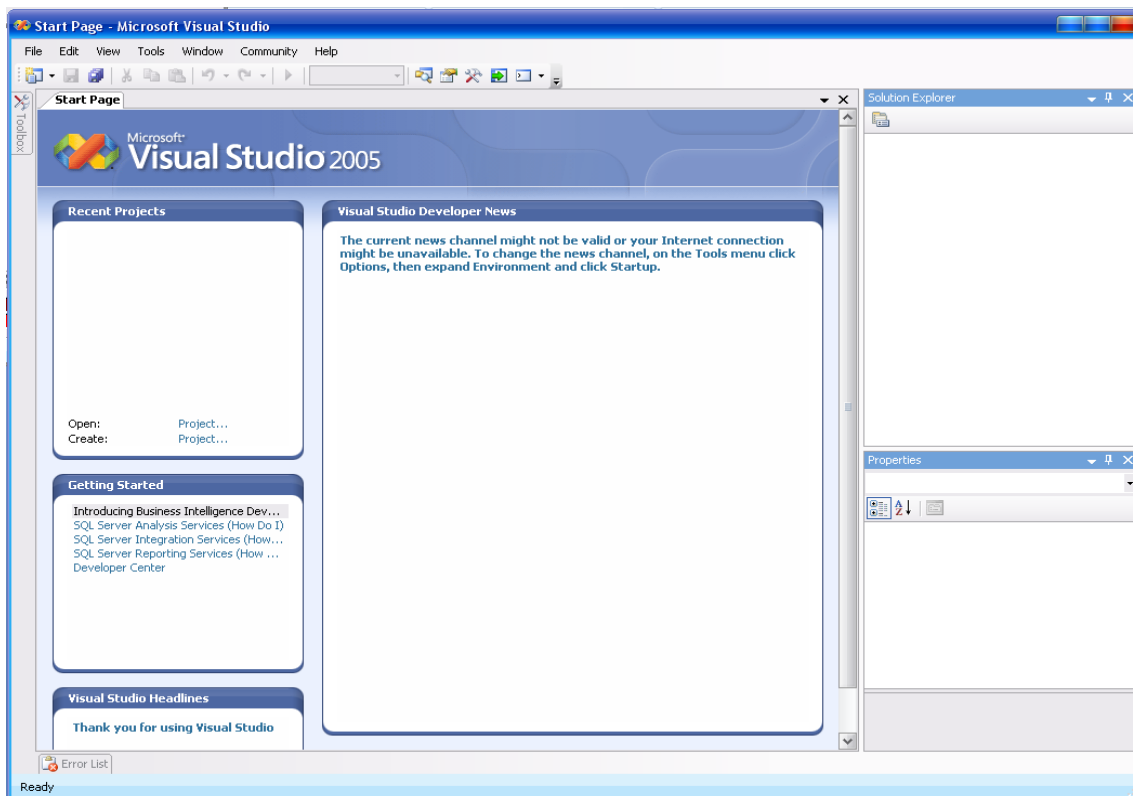
Masalaning dasturini tuzish uchun bizning kompyuterimizga birinchi navbatda Microsoft Visual Studio 2005 (yangi versiyalari ham bo'laveradi), Microsoft SL Server 2005 va ADO.NET texnologiyalari o'rnatilgan bo'lishi lozim.

Bu dasturlarni ishga tushirish uchun birinchi bo'lib uyidagi ketma – ketliklarni amalga oshiramiz.

PUSK ga kirib Microsoft Visual Studio 2005 dasturlash tiliga kirib (1-rasmdagi) ketma-ketlikni amalga oshiramiz

uyidagi oyna hosil bo'ladi: (2-rasm) da keltirilgan.

Bu oynadan dastlabgi ishchi oynasini ya'ni "Proyekt" yaratish uchun (3-rasm) da ko'rsatilganidek "File" nomli tugma "New" "Project" tugmalari bosiladi yoki klaviaturadan Ctrl+Shift+N ni bosib ham (yangi proyekt) yaratsa bo'ladi bo'ladi.

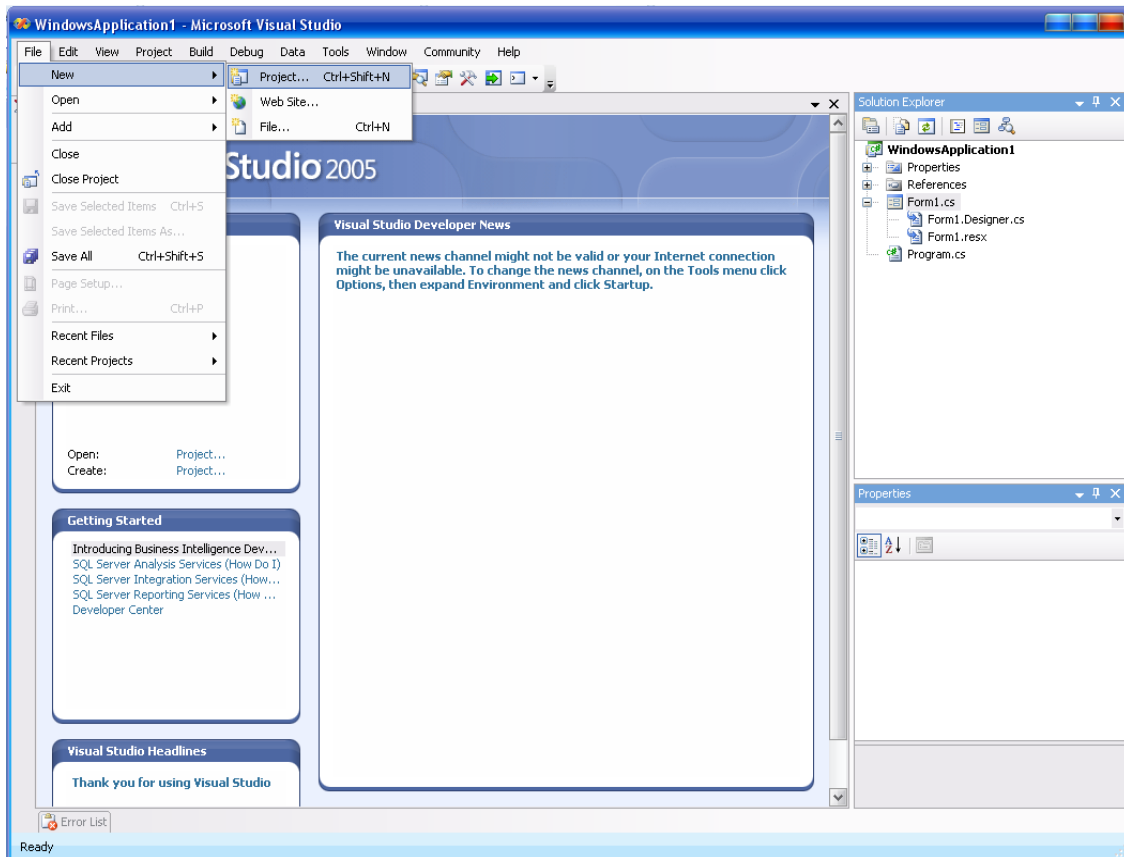


1-rasm

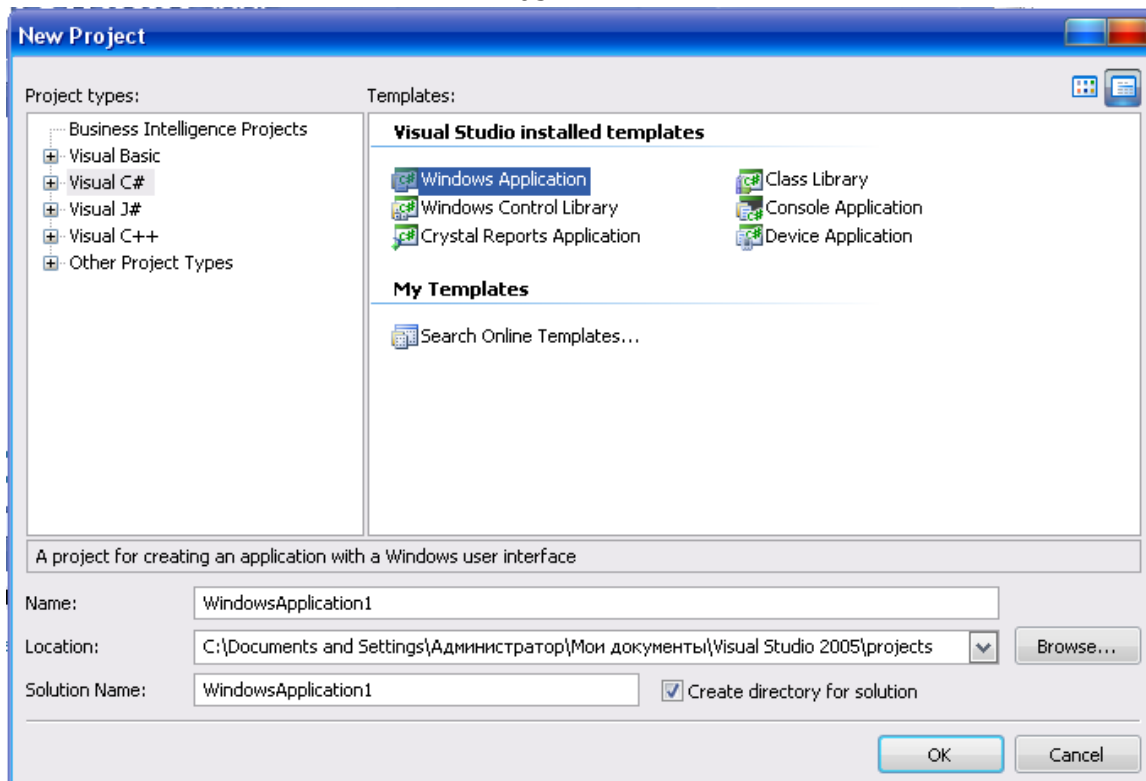
New Project nomli oyna hosil bo'ladi. Bu oynadan Project types ismidan "Visual C#" tanlab Templates ismidan "Windows Application" ni belgilab "OK" tugmasini bosamiz.

"Form1" nomli yangi forma hosil bo'ladi (5-rasm) da keltirilgan

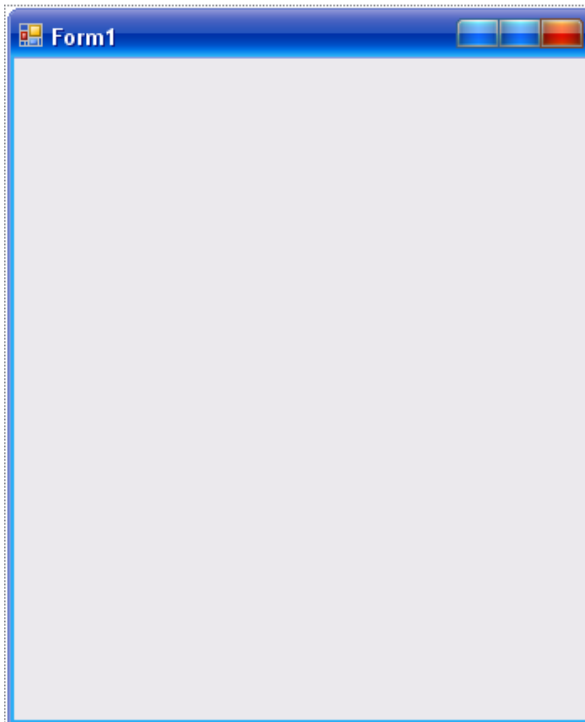
Bu formani View belgisidan "Toolbox" ni tanlaymiz yoki klaviatoriadan Ctrl+ Alt+x ni bosib ham hosil ilsa bo'ladi.



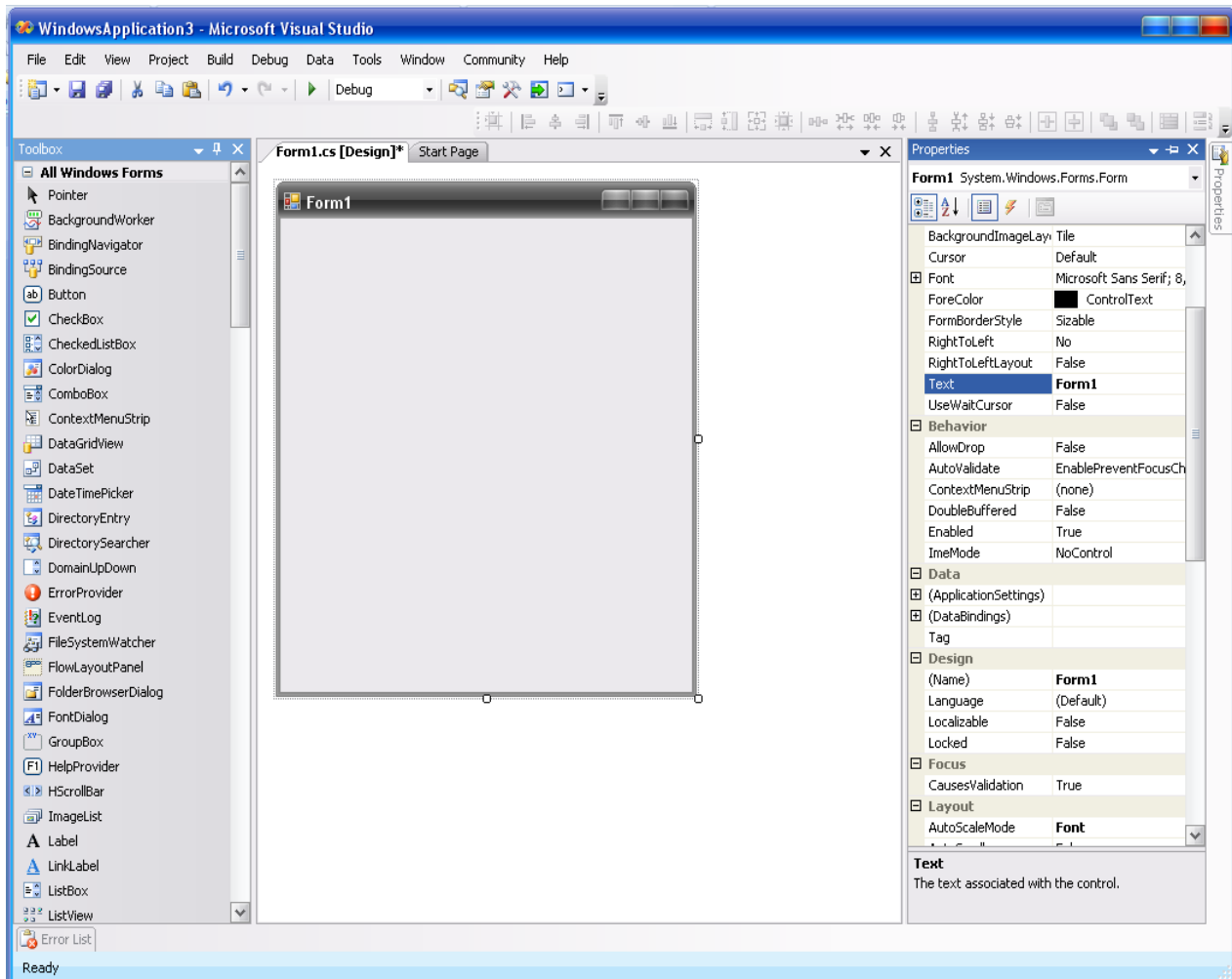
2-rasm



3-rasm



4-rasm

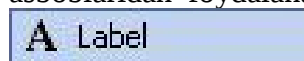


5-rasm

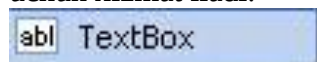
Foydalanuvchi ma'lumotlarini kirituvchi muloot interfeysini yaratish va uning dasturiy kodini kiritish.

Yangi formadan biz foydalanuvchi uchun muloot interfeysi yaratishga xizmat iladi.

Asboblard joylashgan ("Toolbox") ismidan ya'ni chap atordan controllerni ochib uning kerakli asboblardan foydalanamiz



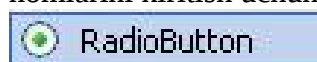
Bu komponenta Formaga Familya, Ism, Guruh va Yo'nalish ismlarini kiritish uchun xizmat iladi.



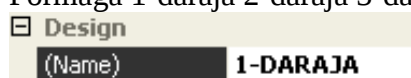
Formadagi belgilangan ismlarga kerakli axborotni kiritish joyini hosil iladi.



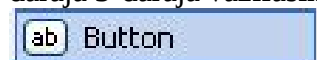
yaratilgan <A Labelga> ya'ni Familya, Ism, Guruh va Yo'nalish nomlarini kiritish uchun (7-rasm) da ko'rsatib o'tilgan



Bu kontrollerlar dastlabki holatda belgilanmagan bo'lishligini ko'rsatadi va Formaga 1-daraja 2-daraja 3-daraja controllarlarini chiqarishda xizmat iladi.



Radio button ornatgan controllarga nom berish ya'ni 1-daraja 2-daraja 3-daraja vazifasini bajaradi.



Formada " OK " tugmasini o'rnatadigan komponenta



Form1 ustiga sichonchaning chap tugmasini bosib Text dan Form1 nomini Ma'lumotga o'zgartiramiz .

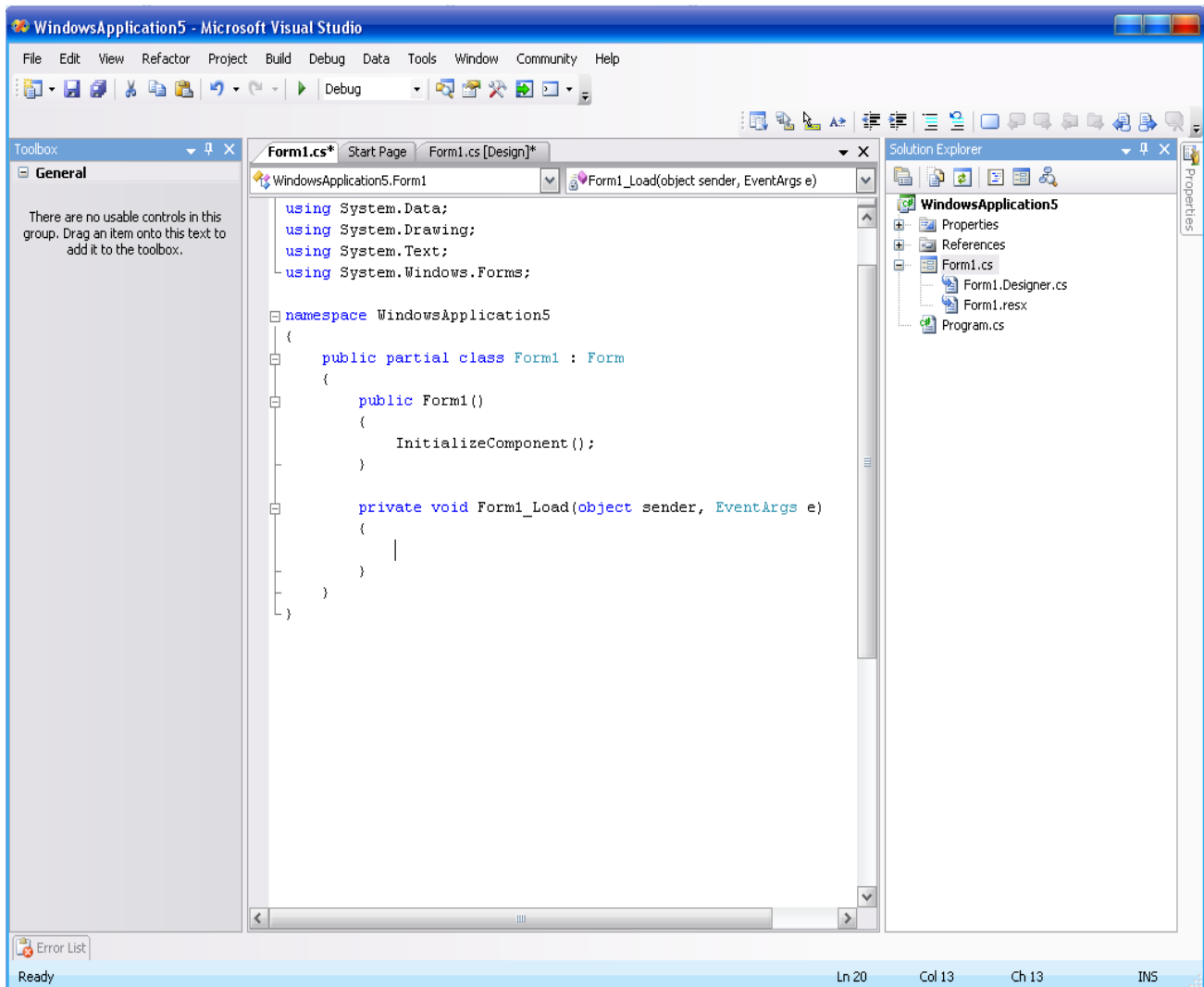
Talaba, Nazorat yozuvlari ham A label da o'rnatiladi.Yuorida ko'rsatilgan vazifalarni to'li bajargandan keyin (7-rasm) dagi oyna hosil bo'ladi.

Bu oynaga orasiga fon berish uchun <BackgroundImage> tanlanib <Localresource> belgilanib birorta rasm tanlanadi va <ok> tugmasi bosiladi.

Yana bosha ozgartirishlar kiritish ham kiritish mumkin "Properties" ismi va asboblard joylashgan Toolbox ismidan foydalangan holda.

Biz yaratgan Forma oddiy va ixcham ilib yatilgan.

Endi Forma ustiga sichonchaning chap tugmasi 2 marta bosganda dasturning kodi yoziladigan yoki kiritiladigan (8-rasm) oyna ociladi:



Avvallari, biz dasturchilik faoliyatimizni boshlaganimizda, bizda dasturiy ta'minotni yaratish uchun kerakli bo'lgan dasturlash tillari sanoli va noulay edi. Hozirda esa, buning aksi, ya'ni, bizda tanlash imkoniyati juda yuori, shuning uchun ba'zan biz dasturlash tillarining ay biridan foydalanishni ham bilmay olamiz. Hozirgi kunda o'nlab (balki yuzlab) dasturlash tillari yaratilgan bo'lib, bular bizga dasturiy ta'minotni yaratishimizda juda o'l kelmoda. Masalan, Object Pascal, Modula, Visual Basic, COBOL, LISP, Piton, C/C++ va boshalar.

Yain yillar ichida esa dunyo maydoniga yana bir dasturlash tili bo'lmish –Microsoft kompaniyasi tomonidan .NET texnologiyasi asosida urilgan C# (si sharp) chidi. Bu dasturlash tili nafaat Microsoft Windows dasturlarini, balki, Web dasturlarni ham yaratishda muhim ahamiyat kasb etadi.

Yangi projektini yaratish uchun aysi dasturlash tilidan foydalanagan ma'ul

Juda ko'plab dasturcilar o'zlariga beradigan bu savolni, menimcha, uyidagicha o'zgartirish to'g'riro bo'ladi: "Yangi projektim uchun aysi texnologiyadan foydalanganim ma'ul".

U yoki bu texnologiyani bir yoki bir nechta dasturlash tillarida o'llash mumkin. Yangi tuzilajak projektga texnologiyani tanlashdagi xatolik, ko'p vat va kuchni ketkazib, oxiri berk ko'chaga kirib ketishga olib keladi. Shu paytning o'zida texnologiyani to'g'ri tanlab, lekin dasturlash tilini xato tanlash ham xuddi shu xatolikning takrorlanishiga olib keladi.

Ko'rib turganingizdek, hozirgi kungacha umuman bir- biriga mos kelmaydigan yoki isman mos keladigan dasturlash tillari va texnologiyalari dunyo yuzini ko'rib ulguran. Oddiy dasturni tuzishda bir dasturlash tili va texnologiyani bilish yetarli, ammo Internet (Web) dasturlarni tuzish esa ko'plab dasturlash tillarni bilishni talab etadi.

Oddiy va Web dasturlarni yaratishga mo'ljallangan Microsoft .NET yangi texnologiyasi murakkab zamonaviy dasturlarni tuzishni o'ta yengillashtirib beradi. Dasturcilar uchun Microsoft

.NET odatiy Microsoft Windows ning dasturlarini, Microsoft Windows uchun tizimli (sistemali) xizmatlarni, bundan tashari Web dastur va Web xizmatlarini o'z ichiga oladi.

Microsoft .NET texnologiyasi uyidagi dasturlash tillarini o'llab- uvvatlaydi:

Microsoft C#

Microsoft Visual Basic .NET

Managed C++

Microsoft Visual J# .NET

Java Script .NET

Bulardan tashari Microsoft .NET ning ichiga Microsoft Visual Studio.NET dastur ishlab chiuvchi dasturiy tillar jamlanmasini va Microsoft.NET Framework dasturlarni ishga tushiruvchi kompilyator ham kiradi.

Microsoft .NET dasturlarini ishga tushiruvchi Microsoft .NET Framework platformasi, dasturcilarga juda ko'plab imkoniyatlar yaratib beradi. Bu isman bosha dasturlash tillarida tuzilgan dasturlarni ham o'iy oladi, ya'ni, dasturlash tilini tanlamaydi. Bosha platformalarga araganda Microsoft .NET Framework platformasida juda ko'plab klasslar kutubxonasi mavjud bo'lib, bular oddiy va Web dasturlarni tuzishda juda o'l keladi. O'zida bir necha minglab klasslarni jamlagan bu kutubxona, siz tuzayotgan dasturni ishlatishga tayyor va ulay. Microsoft .NET Framework platformasi siz oldin yaratgan modullarni ishlatish imkonini beradi, bundan tashari oldin yaratilgan dastur kodiga murojaat ilib yangi komponentalar o'shish mumkin. Eski dasturlarga bu platforma yordamida o'zgartirish kiritib, bu dasturga yangi funksiyaviylik berish mumkin.

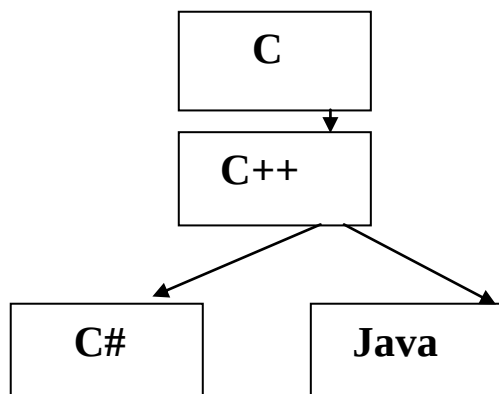
C# dasturlash tili va bu tilga moslashtirilgan Microsoft .NET Framework platformasida izoh yozadigan operator va atributlar dasturning kodini tushunarliro ilishga yordam beradi. Bu aynisa, juda katta projeklarda o'l keladi. Agarda bu projektlarga izoh o'yib ketilmasa, bu projektlar haiiy bir tushunarsiz "chumoli izi" ga o'xshab oladi.

C# sistemasini asosan uyidagi ismlardan iborat: bular dasturni yozish redaktori, C# tili va standart kutubxonalardir. C# dasturi ma'lum bir fazalardan o'tadi. Birinchisi dasturni yozish va tahrirlash, ikkinchisi preprocessor amallarini bajarish, kompilyatsiya, kutubxonalardagi obyekt va funksiyalarni dastur bilan bog'lash (link), xotiraga yuklash (load) va bajarish (execute).

C# dasturi 90 yillarning oxirida ishlab chiilib **Microsoft .NET** ning bir ismiga aylandi. Al'fa versiya sifatida 2000 yildan boshlab ishlatila boshladi .C # bosh arxitektori butun dunyo dasturchilari ichida birinchilar atorida turgan va butun dunyo tomonidan tan olingan **Anders Hejlsberg** bo'ldi. Uning 1980 yillarda chiqarilgan Turbo Paskal dasturi orali ham tanishimiz mumkin.

C# bevosita C, C++ va Java bilan bog'li. Chunki bu uchta til dasturlash olamida eng mashhur tillardir. Bundan tashari profisanal dasturchilar C va C++ ni va juda ko'pchilik Java tilida ish yuritadi.

Biz C# ning kelib chiish genealogik daraxtini ko'rib chisak.



2.1 rasm C# rivojlanish tarixi.

Rasimda C va C++ tillari C# ning asosini tashkil ilishi ko'rib turibmiz. Lekin C# va Java o'zaro o'zgacha ravishda bog'langan. Ularning kelib chiishi C va C++ bo'lsada o'zaro bir biridan fariladi. C# tili ham obe'ktga mo'ljallangan tillar sirasiga kiradi.

C# dasturlash tilining asosiy ismi

C# sistemasi asosan uyidagi ismlardan iborat. Bular dasturni yozish redaktori, C# tili va standart kutubhonalardir. C# dasturi ma'lum bir fazalardan o'tadi. Birinchisi dasturni yozish va tahrirlash, ikkinchisi preprocessor amallarini bajarish, kompilyatsiya, kutubhonalardagi ob'ekt va funksiyalarni dastur bilan bog'lash (link), hotiraga yuklash (load) va bajarish (execute).

C# dasturlash tilida arifmetik amallar

Ko'p programmalar ijro davomida arifmetik amallarni bajaradi. C# dagi amallar uyidagi jadvalda berilgan. Ular ikkita operand bilan ishlatdi. C# dagi amal Arifmetik operator Algebraik ifoda C# dagi ifodasi o'shish + h+19 h+19

Ayirish - f-u f-u

Ko'paytirish * sl s*l

Bo'lish / v/d, vod v/d

Modul olish % k mod 4 k%4

C# da avslarning ma'nisi huddi algebradagidekdir. Undan tashari bosha bosha algebraik ifodalarning ketma-ketligi ham odatdagidek. Ko'paytirish, bo'lish va modul olish operatorlari ijro ko'radi. Agar bir necha operator ketma-ket kelsa, ular chapdan o'nga arab ishlanadi. Bu operatorlardan keyin esa o'shish va ayirish ijro etiladi.

C# dasturlash tili bosharuv ifodalari

Bu bo'limda biz strukturali dasturlashning asosiy prinsip va ismlarini ko'rib chiamiz. Ma'lum bir dasturni yozish uchun belgilangan adamlarni bosib o'tish kerak. Masala anilangandan so'ng uni yechish uchun mo'ljallangan algoritm tuziladi. Keyin esa psevdokod yoziladi. Psevdokod algoritmda bajariladigan adamlarni ko'rsatadi. Bunda faat bajariladigan ifodalar ko'rib chiiladi. Psevdokodda o'zgaruvchi e'lonlari yoki bosha ma'lum bir dasturlash tiliga mansub bo'lgan yordamchi amallar bo'lmaydi. Psevdokodni yozish dasturlashni ancha osonlashtiradi, algoritm mantig'ini tushunishga va uni rivojlanritishga katta yordam beradi. Misol uchun bir dasturning rejasi va psevdokodi 3-4 oy yozilgan bo'lsa va yuori darajada detallashtirilgan bo'lsa, ushbu dasturning C# yoki bosha tildagi kodini yozish 2-3 hafta vat oladi halos. Bu yozilgan programmada hato ancha kam bo'ladi, uni keyinchalik takomillashtirish arzonga tushadi. Hozirgi paytda dastur o'zgarishi fafulotda hodisa emas, balki zamon talabidir.

if strukturalari

Biz shartga ko'ra bir necha harakat yo'lidan bittasini tanlaymiz. Misol uchun agar bolaning yoshi 7 ga teng yoki katta bo'lsa u maktabga borishi mumkin bo'lsin. Buni C# da if ni o'llab yozaylik.

```
if (yosh >= 7)
```

```
maktab();
```

Bu yerda shart bajarilishi yoki bajarilmasligi mumkin. Agar yosh o'zgaruvchisi 7 ga teng yoki undan katta bo'lsa shart bajariladi va maktab () funksiyasi chaqiriladi. Bu holat true (to'g'ri) deyiladi. Agar yosh 7 dan kichik bo'lsa, maktab() tashlab o'tiladi. Yani false (noto'g'ri) holat yuzaga keladi. Biz shart ismini mantiy operatorlarga asoslanganligini ko'rib chiqaymiz. Aslida esa shartdagi ifodaning ko'rinishi muhim emas – agar ifodani nolga keltirish mumkin bo'lsa false bo'ladi, noldan farqli javob bo'lsa, musbatmi, manfiymi, true holat paydo bo'ladi va shart bajariladi.

Bunga o'shimcha ilib o'tish kerakki, C# da mahsus bool tipi mavjud. Bu tipdagi o'zgaruvchilarning yordamida bul (mantiy) arifmetikasini amalga oshirish mumkin. bool o'zgaruvchilar faqat true yoki false qiymatlarini olishlari mumkin.

if/else strukturalari

If ni o'llaganimizda ifoda faqat shart haqiqat bo'lgandagina bajariladi, aks holda tashlanib o'tiladi. if/else yordamida esa shart bajarilmaganda (false natija chiqqanda) else orali bosha bir yo'lidan borishni belgilash mumkin. Misolimizni takomillashtirsak. Bola 7 yosh yoki undan katta bo'lsa maktabga, 7 dan kichik bo'lsa bog'chaga borsin.

```
if (yosh >= 7)
```

```
maktab(); //nuta-vergul majburiydir
```

```
else
```

```
bog'cha();
```

Yuqorida if ga tegishli bo'lgan blok bitta ifodadan (maktab()) iborat. Shu sababli nuta-vergul o'yilishi shart. Buni aytib o'tishimizning sababi, masal Pascaldagi hech narsa o'yilmasligi shart. C# da bitta ifodaga turgan joyga ifodalar guruhini { } avslarda olingan holda o'yiladi.

Switch strukturalari

If-else-if yordami bilan bir necha shartni test qilishimiz mumkin. Lekin bunday yozuv nisbatan o'qishga qiyin va ko'rinishi o'qilmay bo'ladi. Agar shart ifoda butun son tipida bo'lsa yoki bu tipga keltirilishi mumkin bo'lsa, biz switch (tanlash) ifodalarini ishlatamiz.

Switch strukturalari bir necha case etiketlaridan (label) va majburiy bo'lmagan default etiketidan iboratdir. Etiket bu bir nomdir. U dasturning bir nutasidagi o'yiladi. Programmaning bosha yeridan

ushbu etiketga o'tishni bajarish mumkin. O'tish yoki sakrash goto bilan amalga oshiriladi, switch blokida ham o'llaniladi.

C# dasturlash tilida ilk yaratilgan dasturimiz

```
/*Ekranga yozuv chiarish*/  
using System;  
namespace ConsoleApplication1  
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
            Console.WriteLine("Talabalarga alangali salom!");  
        }  
    }  
}
```

Ekranda: Talabalarga alangali salom!

Yozuvi hosil bo'ladi.

Dasturni satrma- satr tahlil ilaylik: C# da ikki tur sharxlar mavjud. /* bilan boshlanib, */ bilan tugaydigani bir necha satrni egallashi mumkin. Ya'ni bu belgilar orasida olgan hamma yozuv sharx hisoblanadi. Bu tur sharx C dan olgan. C# yangi ko'rinishdagi sharxlar ham kiritilgan. Bu // bilan boshlanadi va kuchi shu satr oxirigacha salanadi. Sharxlar yoki boshacha ilib aytganda kommentariylar kompilyator tomonidan hisobga olinmaydi va hech anday mashina ijro kodiga aylantirilmaydi. Sharxlar kerakli joyda, funksiyalardan oldin, o'zgaruvchilar e'lonidan keyin yozilganda, dasturni tushunish ancha osonlashadi va keyinchalik dastur ishlash mantig'ini esga solib turadi. using System; Bu juda katta kutubxona bo'lib, unda har xil funktsiyalar va klasslar mavjud. namespace ConsoleApplication1 esa ushbu klass aysi nom ichida joylashganligini bildirish uchun moljallangan. Void main() har bir C# dasturining ismidir. main dan keyingi () avslar C# ning funksiya deb ataluvchi blokining boshlanganini bildiradi. C# dasturi bir yoki bir necha funksiyalardan iborat. Va shulardan ani bitta funksiya main deb atalishi shart. Bunda main dastur ichida keladigan birinchi funksiya bo'lmasligi ham mumkin. Operatsion sistema dastur ijrosini main() funksiyasidan boshlaydi. main() dan oldin kelgan int esa main funksiyasidan aytish iymati tipini belgilaydi. Bunda int integer, yani butun son deganidir. main() ning aytargan iymati operatsion sistemaga boradi. { avs funksiya va bosha bloklar tanasini boshlaydi. Blokni yopish uchun } avsi ishlatiladi.

Console.WriteLine("Talabalarga alangali salom!"); satri C# da ifoda deb ataladi. C# dagi har bir ifoda ; (nuta-vergul) bilan tugatilishi shart. Orticha ; bo'sh ifoda deyiladi. Uni o'yish dastur tezligiga ta'sir ilmaydi. Kirish va chiish (Input/Output), ya'ni dasturga kerakli ma'lumotlarni kiritish va ular ustida dastur tomonidan bajarilgan amallar natijalarini olish C# da oim ob'ektlari orali bajarilishi mumkin. Lekin kirish/chiishni C dagi kabi funksiyalar bilan ham amalga oshirsa bo'ladi. C# falsafasiga ko'ra har bir kirish/chiish jihozi (ekran, printer, klaviatura...) baytlar oimi bilan ishlagandek abul ilinadi.

Yuoridagi ifoda bajarilganda bizning "Talabalarga alangali salom!" gapimiz standart chiish oimi obykti WriteLine ga (writeline - console out) jo'natiladi. Normal sharoitda bu oim ekranga ulangandir. C# da satrlar (string) o'shtirtnolar (") orasida bo'ladi. Bitta harfli literallar esa bitta tirno - apostrof (') ichiga olinadi. Misol uchun: 'A','\$'. Bitta harf yoki belgini o'shtirtno ichiga olsa u satr kabi abul ilinadi. Console.WriteLine operatori oimga kiritish operatori deyiladi. Dastur ijro etilganda Console.ReadLine() operatorining o'ng tomonidagi argument ekranga yuboriladi. Bunda ekranga o'shtirtno ("...") ichidagi narsa bosib chiariladi. Lekin e'tibor bersak, \n belgisi bosilmadi. \ (teskari kasr - backslash) belgisi maxsus ma'noga ega. U o'zidan keyin kelgan belgi oim buyrug'i yoki manipulyatori ekanligini bildiradi. Shunda \ belgisi bilan undan keyin kelgan belgi buyru ketma-ketligida ayplanadi.

Bularning ro'yxatini beraylik:

- \n - Yangi satr. Kursor yangi ator boshidan joy oladi.
- \t - Gorizontol tabulyatsiya (kursor bir-necha harf o'nga siljiydi).
- \v - Vertikal tabulyatsiya (bir-necha satr tashlanib o'tiladi).
- \r - aytish. Kursor ayni satr boshiga aytadi, yani yangi satrga o'tmaydi.
- \a - Kompyuter dinamiki chalinadi.
- \\ - Ekranga teskari kasr belgisini bosish uchun o'llaniladi.
- \'' - Ekranga o'shtirno belgisini bosish uchun o'llaniladi.

return 0; (return - aytmo) ifodasi main() funksiyasidan chiishning asosiy yo'lidir. 0 (nol) iymatining aytarilishi operatsion sistemaga ushbu dastur normal bajarilib tugaganini bildiradi. return orali aytadigan iymat tipi funksiya e'lonidagi aytish tipi bilan bir xil bo'lishi kerak. Bizda bu e'lon int main(){...} edi. Va 0 int tipiga mansubdir. Bundan keyin return orali aytarilayotgan ifodani avs ichiga olamiz. Misol uchun return (6). Bu avslar majburiy emas, lekin bizlar ularni dasturni o'ishda ulaylik uchun kiritamiz.

C# Dasturida Arifmetik Amallar

Ko'p dasturlar ijro davomida arifmetik amallarni bajaradi. C# dagi amallar uyidagi jadvalda berilgan. Ular ikkita operand bilan ishlatdi. C# dagi amal Arifmetik operator Algebraik ifoda C# dagi ifodasi

o'shish + h+19 h+19

Ayirish - f-u f-u

Ko'paytirish * sl *sl

Bo'lish / v/d, v/d, v/d

Modul olish % k mod 4 k%4

Bularning ba'zi birlarining xususiyatlarini ko'rib chiaylik. Butun sonli bo'lishda, ya'ni bo'luvchi ham, bo'linuvchi ham butun son bo'lganda, javob butun son bo'ladi. Javob yaxlitlanmaydi, kasr ismi tashlanib yuborilib, butun ismining o'zi oladi.

Modul operatori (%) butun songa bo'lishdan kelib chiadigan oldini beradi. x%y ifodasi x ni y ga bo'lgandan keyin chiadigan oldini beradi. Demak, 7%4 bizga 3 javobini beradi. % operatori faat butun sonlar bilan ishlaydi. Vergulli (real) sonlar bilan ishlash uchun "math.h" kutubxonasidagi f mod funksiyasini o'llash kerak. C# da avslarning ma'nosi xuddi algebradagidekdir. Undan tashari bosha-bosha algebraik ifodalarning ketma-ketligi ham odatdagidek. Ko'paytirish, bo'lish va modul olish operatorlari ijro ko'radi. Agar bir necha operator ketma-ket kelsa, ular chapdan o'nga arab ishlanadi. Bu operatorlardan keyin esa o'shish va ayirish ijro etiladi.

MANTIY SOLISHTIRISH OPERATORLARI

C# bir necha solishtirish operatorlariga ega. Algebraik ifoda C# dagi operator C# dagi ifoda Algebraik ma'nosi tenglik guruhi == x==y x tengdir y ga teng emas != x!=y x teng emas y ga solishtirish guruhi > x>y x katta y dan < x<y x kichkina y dan katta-teng >= x>=y x katta yoki teng y ga kichik-teng <= x<=y x kichik yoki teng y ga ==, !=, >= va <= operatorlarni yozganda oraga bo'sh joy o'yib ketish sintaksis hatodir. Yani kompilyator dasturdagi hatoni ko'rsatib beradi va unituzatilishini talab iladi. Ushbu ikki belgili operatorlarning belgilarining joyini almashtirish, masalan <= ni =< ilib yozish ko'p hollarda sintaksis hatolarga olib keladi. Gohida esa != ni != deb yozganda sintaksis hato vujudga ham, bu mantiy hato bo'ladi. Mantiy hatolarni kompilyator topa olmaydi. Lekin ular dastur ishlash mantig'ini o'zgartirib yuboradi. Bu kabi hatolarni topish esa ancha mashaatli ishdir (! operatori mantiy inkordir). Yana bosha hatolardan biri tenglik operatori (= =) va tenglashtirish, iymat berish operatorlarini (=) bir-biri bilan almashtirib o'yishdir. Bu ham juda ayanchli oibatlariga olib keladi, chunki ushbu hato aksariyat hollarda manti xatolariga olib keladi.

BOS HARUV IFODALARI

Bu bo'limda biz strukturali dasturlashning asosiy prinsip va ismlarini ko'rib chiqamiz. Ma'lum bir dasturni yozish uchun belgilangan adamlarni bosib o'tish kerak. Masala anilangandan so'ng uni yechish uchun mo'ljallangan algoritm tuziladi. Keyin esa psevdokod yoziladi. Psevdokod algoritmda bajariladigan adamlarni ko'rsatadi. Bunda faat bajariladigan ifodalar ko'rib chiqiladi. Psevdokodda o'zgaruvchi e'lonlari yoki bosha ma'lum bir dasturlash tiliga mansub bo'lgan yordamchi amallar bo'lmaydi. Psevdokodni yozish dasturlashni ancha osonlashtiradi, algoritm mantiqini tushunishga va uni rivojlantirishga katta yordam beradi. Misol uchun bir dasturning rejasini va psevdokodini 3-4 oy yozilgan bo'lsa va yuqori darajada detallashtirilgan bo'lsa, ushbu dasturning C# yoki bosha tilidagi kodini yozish 2-3 hafta vaqt oladi halos. Bu yozilgan dasturda hato ancha kam bo'ladi, uni keyinchalik takomillashtirish arzonroq tushadi. Hozirgi paytda dastur o'zgarishi faqatgina hodisa emas, balki zamon talabidir.

DASTUR IJRO STRUKTURALARI

Asosan dasturdagi ifodalar ketma-ket, navbatiga ko'ra ijro etiladi. Gohida bir shart bajarilishiga ko'ra, ijro bosha bir ifodaga o'tadi. Navbatdagi emas, dasturning bosha yerida joylashgan ifoda bajariladi. Yani sakrash yoki ijro ko'chishi vujudga keladi. 60-yillarga kelib, dasturlardagi ko'pchilik hatolar aynan shu ijro ko'chishlarining rejasiz ishlatilishidan kelib chiqishi ma'lum bo'ldi. Bunda eng katta aybdor deb bu ko'chishlarni amalga oshiruvchi goto (...ga bor) ifodasi belgilandi. goto dastur ijrosini deyarli istalgan yerga ko'chirib yuborishi mumkin. Bu esa dasturni o'ishni va uning strukturasini murakkablashtirib yuboradi. Shu sababli "strukturali dasturlash" atamasi "goto ni yo' ilish" bilan tenglashtiriladi. Shuni aytib o'tish kerakki, goto kabi shartsiz sakrash amallarini bajaruvchi ifodalar bosha dasturlash tillarida ham bor. Tadqiqotlar shuni ko'rsatdiki, istalgan dastur goto siz yozilishi mumkin ekan. goto siz yozish uslubi strukturali dasturlash deb nom oldi. Va bunday dastur yozish metodi katta itisodiy samara beradi. Strukturali dasturlash asosi shundan iboratki, har bir dastur faatgina uch xil bosharuv strukturalaridan iboratdir. Bular ifodalarni ketma-ket ijro etish strukturasini (sequence structure), tanlash strukturasini (selection structure) va amalni ayta ijro etish strukturasidir (repetition structure). Ifodalarni ketma-ket ijro etish strukturasini C# tomonidan ta'minlanadi. Normal sharoitda C# ifodalari dasturdagi navbatiga ko'ra bajariladi. Tanlash buyrulari uchta. Bular if, if/else va switch dir. ayta ijro etish buyrulari guruhiga ham uchta a'zo bor, bular while, do/while va for. Bularni har birini keyinroq tahlil qilib chiqamiz.

Yuqoridagi buyrular nomlari C# dasturlash tilining mahsus so'zlaridir. Dasturchi bu so'zlarni o'zgaruvchi yoki funksiyalar nomi sifatida o'llashi ta'yanlanadi. uyida C# ning ajratilgan so'zlarining to'liq ro'yxati berilgan.

C# va C ga tegishli:

auto do goto signed unsigned break double if sizeof void case else int static volatile char enum long struct while const extern register switch continue float return typedef default for short union.

Faat C# ga arashli:

asm explicit operator this virtual bool false private throw wchar_t catch friend protected true class inline public try const_cast mutable reinterpret_cast typeid delete namespace static_cast typename dynamic_cast new template using.

C# dagi yetita bosharuv strukturasini aytib o'tdik. Ular bittagina boshlanish nutasiga va bittagina chiish nutasiga egadirlar. Demak biz bu dastur bo'laklarini ketma-ket ulab ketishimiz mumkin. Bosharuv strukturalarining bu kabi ulanishini devorning g'ishtarini ustma-ust alashga ham taoslasak bo'ladi. Yoki biz bu bloklarni bir-birining ichiga joylashtirishimiz mumkin. Bu kabi o'llashish ikkinchi uslub bo'ladi. Mana shu ikki yo'l bilan bog'langan yetita blok yordamida biz istalgan dasturimizni yoza olamiz.

if{} else{} operatorining strukturasini.

Biz shartga ko'ra bir necha harakat yo'lidan bittasini tanlaymiz. Misol uchun agar bolaning yoshi 7 ga teng yoki katta bo'lsa u maktabga borishi mumkin bo'lsin. Buni C# da if ni o'llab yozaylik.

```
if (yosh >= 7)
    maktab();
```

Bu yerda shart bajarilishi yoki bajarilmasligi mumkin. Agar yosh o'zgaruvchisi 7 ga teng yoki undan katta bo'lsa shart bajariladi va maktab() funksiyasi chairiladi. Bu holat true (to'g'ri) deyiladi. Agar yosh 7 dan kichik bo'lsa, maktab() tashlab o'tiladi. Yani false (noto'g'ri) holat yuzaga keladi. Biz shart ismini mantiy operatorlarga asoslanganligini ko'rib chian edik. Aslida esa shartdagi ifodaning ko'rinishi muhim emas – agar ifodani nolg'a keltirish mumkin bo'lsa false bo'ladi, noldan farli javob bo'lsa, musbatmi, manfiymi, true holat paydo bo'ladi va shart bajariladi.

Bunga o'shimcha ilib o'tish kerakki, C# da mahsus bool tipi mavjud. Bu tipdagi o'zgaruvxilarning yordamida bul (mantiy) arifmetikasini amalga oshirish mumkin. bool o'zgaruvxilar faat true yoki false iymatlarini olishlari mumkin.

if ni o'llaganimizda ifoda faat shart haiat bo'lgandagina bajariladi, aks holda tashlanib o'tiladi. if/else yordamida esa shart bajarilmaganda (false natija chianda) else orali bosh'a bir yo'ldan borishni belgilash mumkin. Misolimizni takomillashtirsak. Bola 7 yosh yoki undan katta bo'lsa maktabga, 7 dan kichkina bo'lsa bog'chaga borsin.

```
if (yosh >= 7)
    maktab(); //nuta-vergul majburiydir
else
    bogcha();
```

Yuorida if ga tegishli bo'lgan blok bitta ifodadan (maktab()) iborat. Shu sababli nuta-vergul o'yilishi shart. Buni aytib o'tishimizning sababi, masal Pascald'a hech narsa o'yilmasligi shart.

C# da bitta ifoda turgan joyga ifodalar guruhini {} avslarda olingan holda o'ysa bo'ladi.

Masalan:

```
if (yosh >= 7)
{
    Console.WriteLine("Maktabga!\n");
    maktab();
}
else
{
    Console.WriteLine( "Bog'chaga!\n" );
    bogcha();
}
```

Aslida har doim {} avslarni o'yish yaxshi odat hisoblanadi; keyinchalik bir ifoda turgan joyga o'shimcha ilinganda avslardan biri unutilib olmaydi. Strukurali dasturlashning yana bir xarakterli joyi

shundaki tabulyatsiya, bo'sh joy va yangi satrlar ko'p o'llaniladi. Bu dasturni o'ishni osonlashtirish uchun ilinadi. C# uchun bo'sh joyning hech ahamiyati yo', lekin dasturni tahrir ilayotgan odamga buyrular guruhini, bloklarni tabulyatsiya yordamida ajratib bersak, unga katta yordam bo'ladi. Yuoridagini uyidagicha ham yozish mumkin:

```
if(yosh>=7){Console.WriteLine ("Maktabga!\n");  
maktab()}else{Console.WriteLine ("Bog'chaga!\n");bogcha();}  
Biro buni o'ish ancha murakkab ishdur.
```

C# da if/else strukturasiga o'xshash : shart operatori (conditional operator) ham bordir. Bu C# ning bittagina uchta argument oluvchi operatori. Uch operand va shart operatori shart ifodasini beradi. Birinchi operand orali shartimizni beramiz. Ikkinchi argument shart true (haiat) bo'lib chiandagi butun shart ifodasining javob iymatidir. Uchinchi operand shartimiz bajarilmay (false) olgandagi butun shart ifodasining iymatidir. Masalan:

```
bool bayro;  
int yosh = 10;  
bayro = ( yosh >= 7 true : false );
```

Agar yosh 7 ga teng yoki katta bo'lsa, bool tipidagi o'zgaruvchimiz true iymatini oladi, aks tadirda false bo'ladi. Shart operatori avslar ichida bo'lishi zarur, chunki uning kuchi katta emas. Javob iymatlar bajariladigan funksiyalar ham bo'lishi mumkin:

```
yosh >= 7 maktab() : bog'cha();
```

if/else strukturalarini bir-birining ichida yozishimiz mumkin. Bunda ular bir-biriga ulanib ketadi.

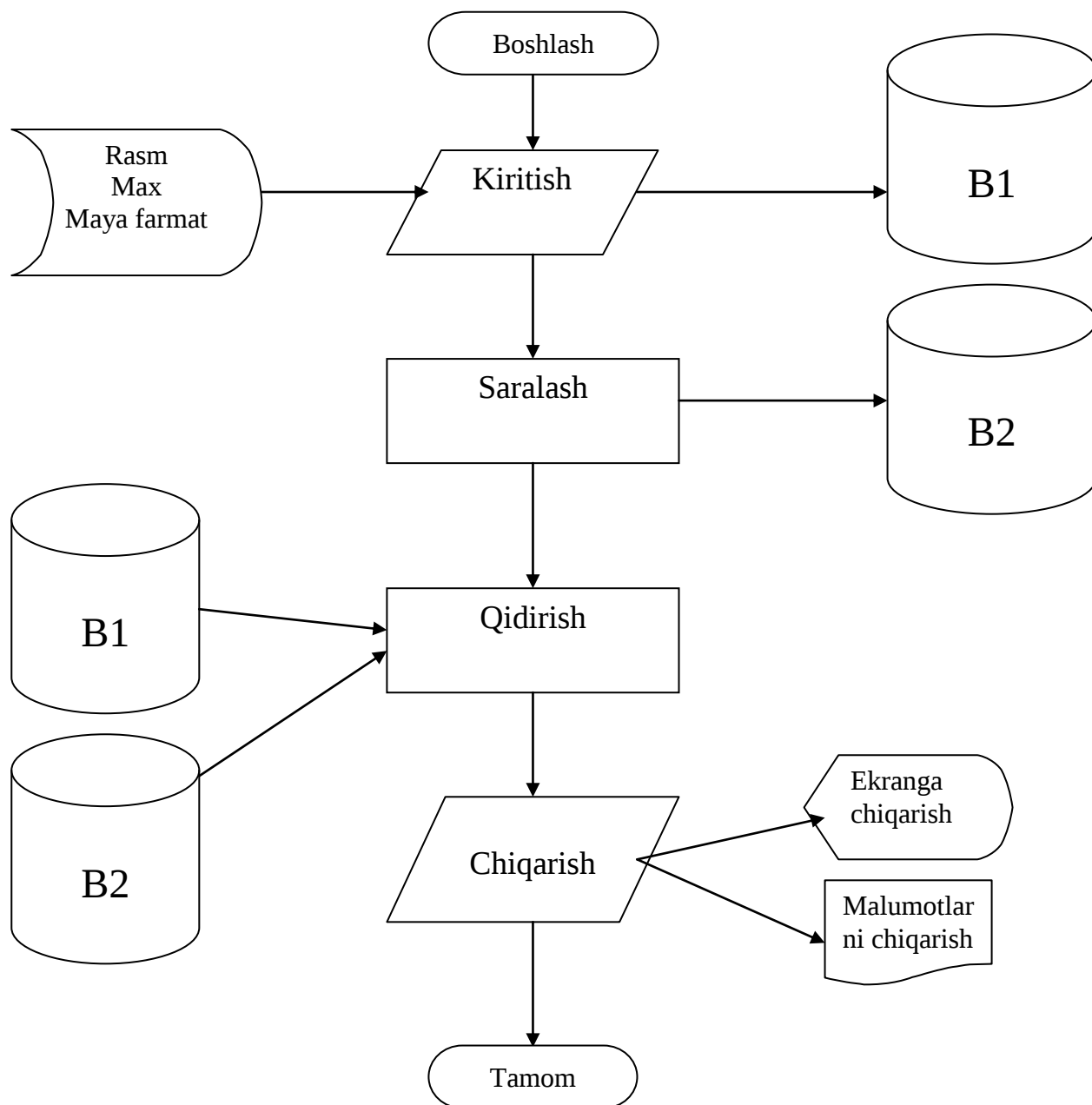
Xulosa

Xulosa ilib aytadigan bo'lsak, birinchi bob yakunida ARM dazaynlarini yaratadigan dasturlarni tahlil ilib olib, ularning birida ARM dizayn ko'rinishlarini yaratish masadga muvofi bo'lib unda 3d max uch o'lchovli grafik dasturning ishlatish tartiblari tushuntirilib berib, bobning ikkinchi ismida ARM dizayn ko'rinishlarini tanlab oladigan dasturiy vositalardan biri C# tili haqida ma'lumot berib o'tildi.

2-BOB. ARM DIZAYNI UCHUN BO'LGAN IDIRUV DASTURINING TAHLILI

2.1. ARM dizaynini uchun bo'lgan idiruv dasturining funksional strukturasi, masalalar shemasi va malumotlarni kiritish algoritmlari

Masalalarni dasturlar sxemasini umumiy ko'rinishi.



Masalalarni dasturlar sxemasini umumiy ko'rinishining izoxi.

Boshlanish- dasturni boshlanishi;

Video, rasm, max- Bu kiritiladigon ma'lumot. Bu ma'lumotlarni biz diskdan,fleshkadan va bosha narsalar orali kiritamiz;.

Kiritish- Ma'lumotlar kiritiladi;

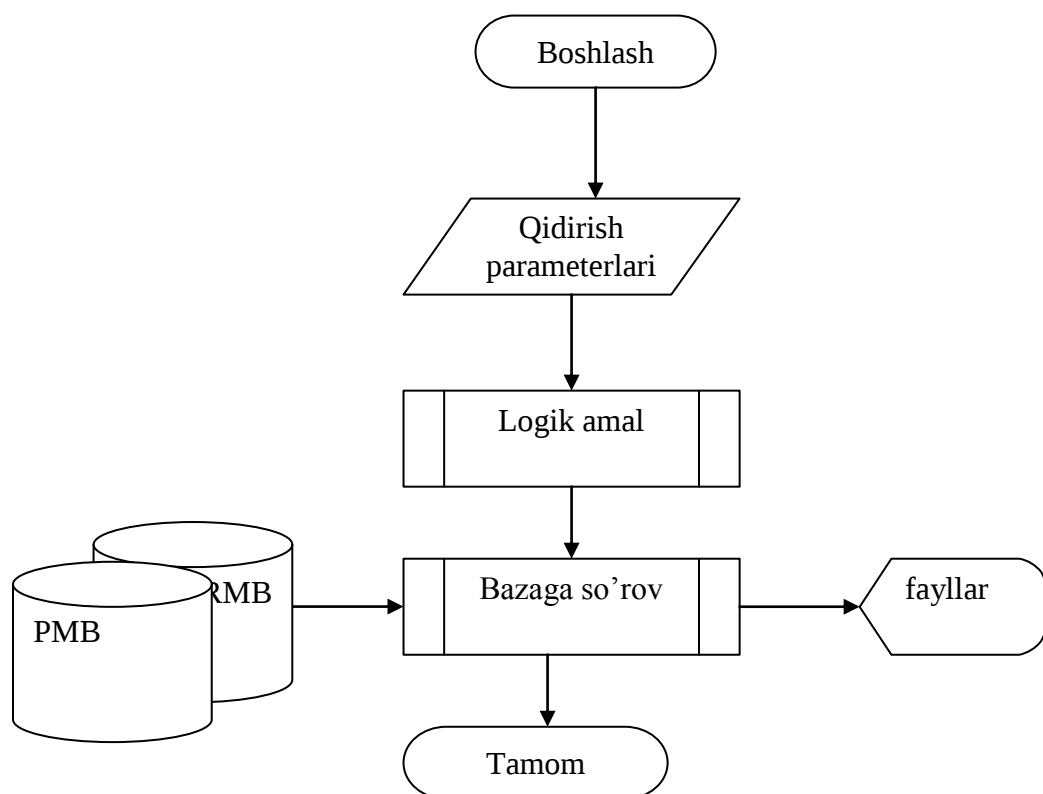
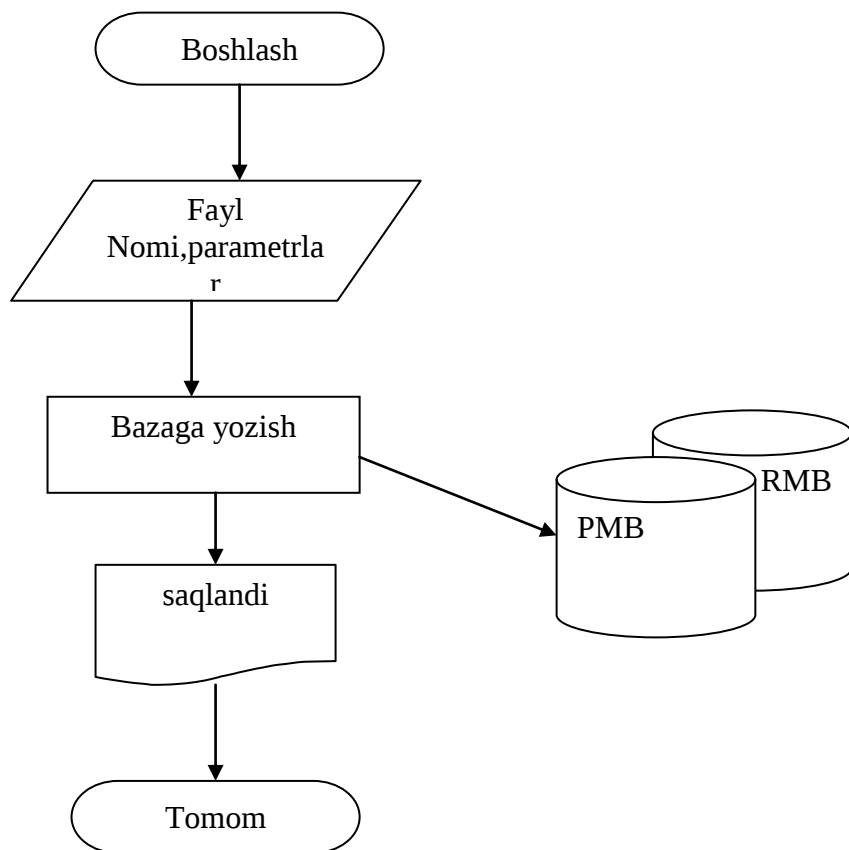
B1- Umumiy ba'za bumda barcha ma'lumotlar masalan:videosi, rasmlar, max salanadi;

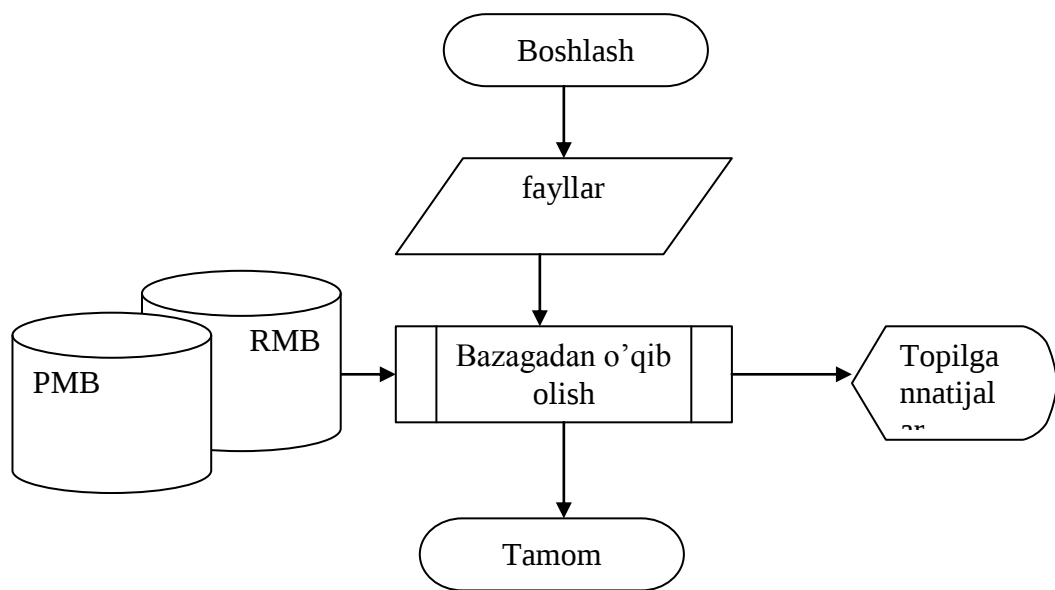
Saralash-Bunda ma'lumotlar saralanadi;

B2-Bunda saralangan ma'lumotlar salanadi.;

idirish-Bunda ma'lumotlar idiriladi; chiarish jarayonida ekranga chiarish va raspechatka ilib chiarish mumkin.

Tamom-dasturni tugatilishi.

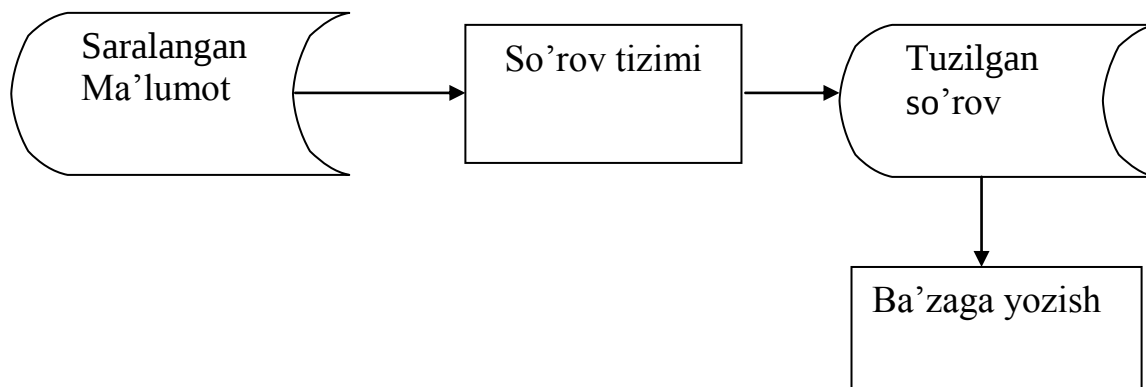




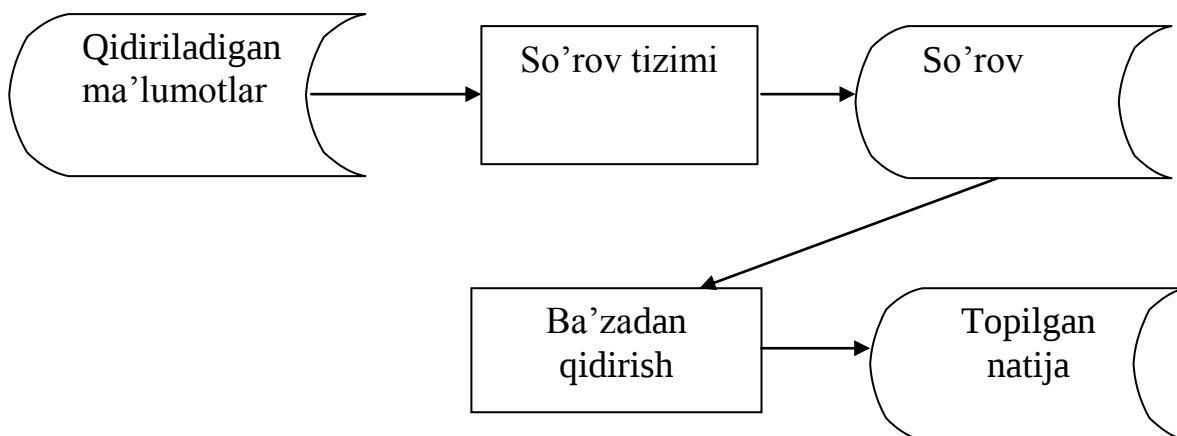
Kiritish tizim osti masalalar sxemasi.



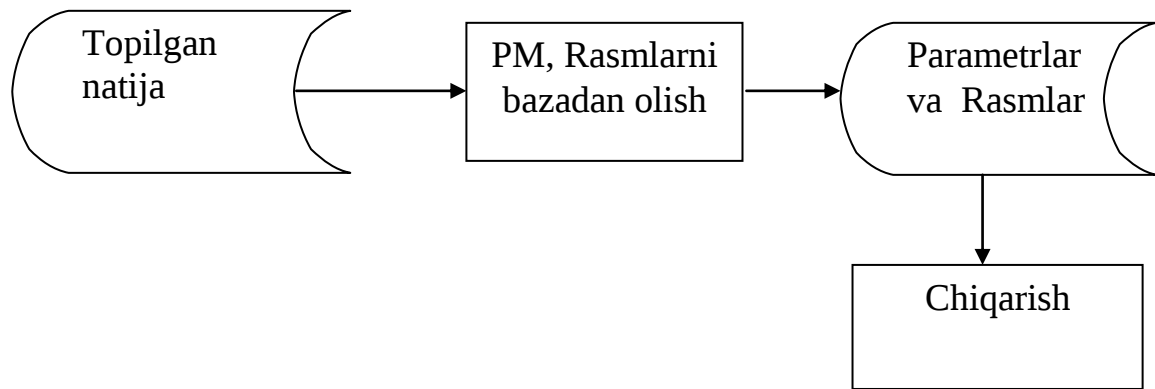
Bazaga yozish tizim osti masalalar sxemasi.



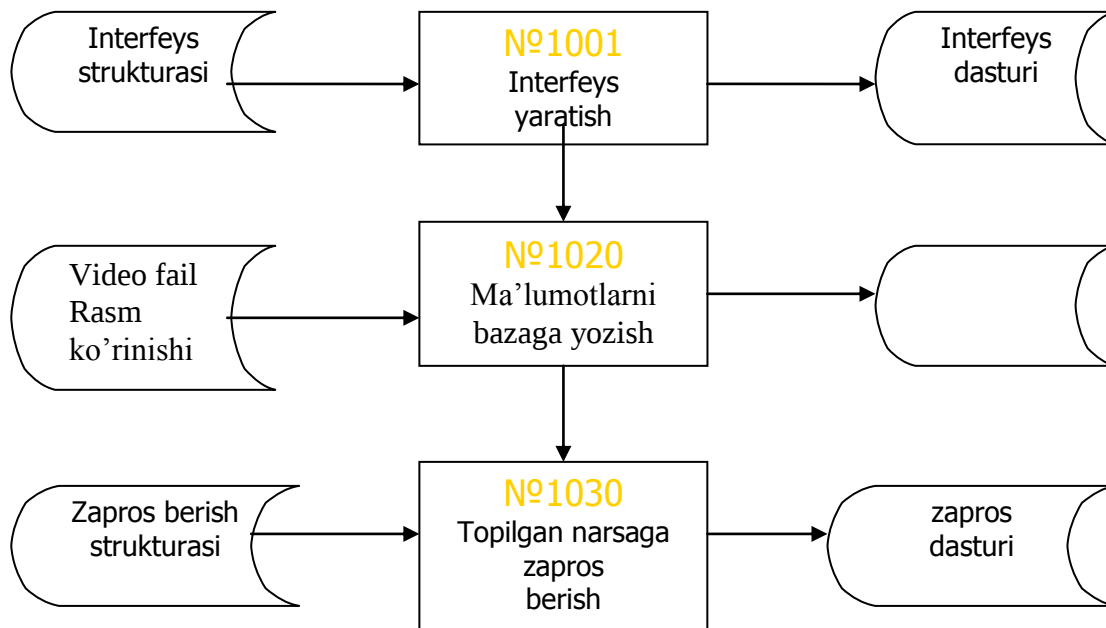
Qidirish tizim osti masalalar sxemasi.



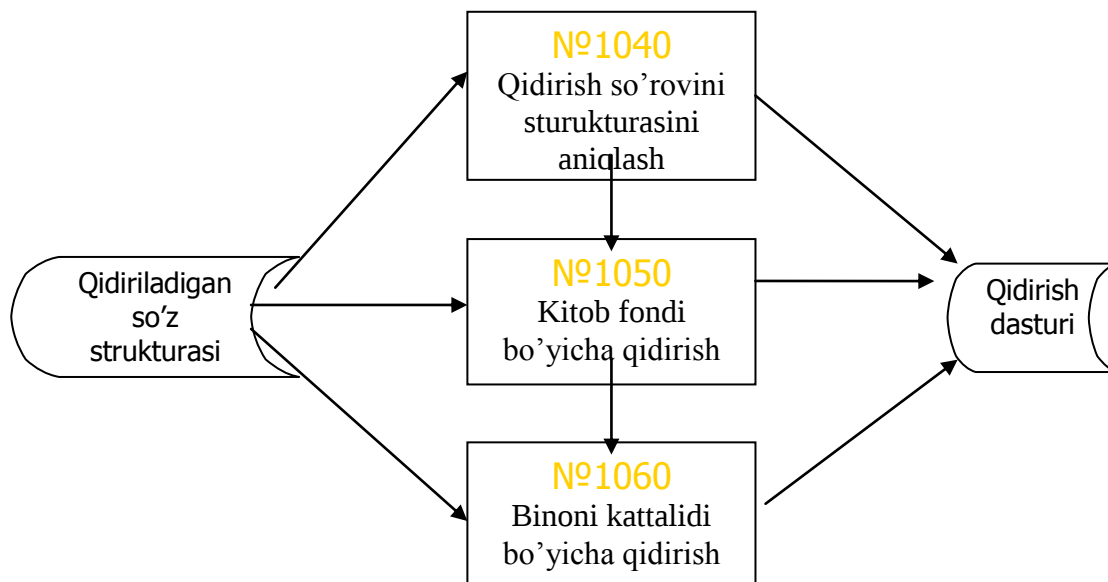
Chiqarish tizim osti masalalar sxemasi.



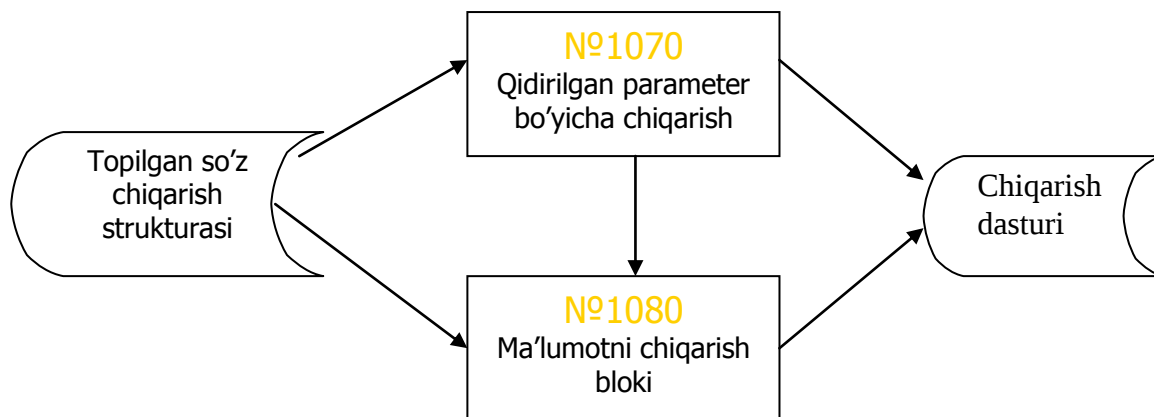
Ma'lumotlarni kiritish blokining masalalar sxemasi.



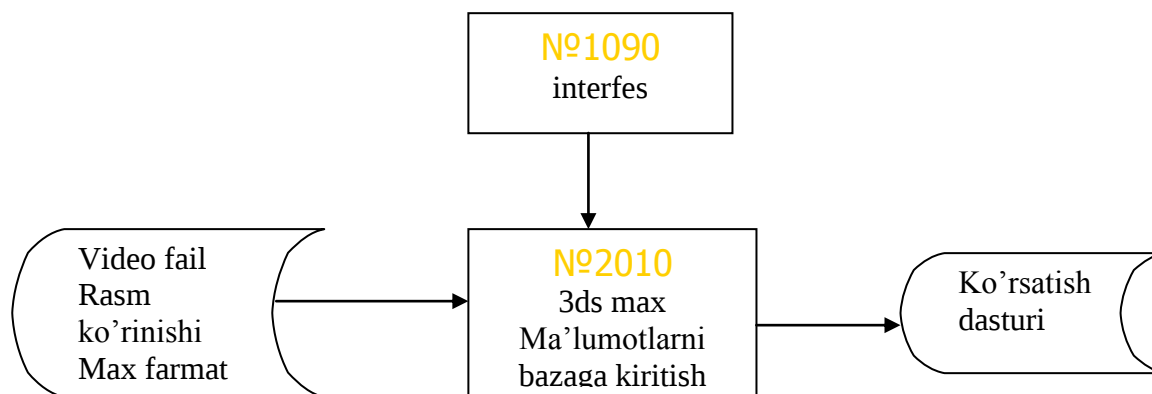
idirish tizimi blokining masalalar sxemasi.



Ma'lumotlarni chiarish bloki masalalari sxemasi



3ds max ilingan ma'lumotlarni kiritish



Masalalarni ko'rsarkichlari

Masalani kodi	Masalani nomi	Masalani masadi	Kiritiladigan ma'lumot	Chiish ma'lumoti
№1001	Interfeys yaratish	Dasturni interfeysini yaratish	Interfeysni yaratish	Dastur interfeysi
№1020	Ma'lumotlarni bazaga yozish	-	Bino haqida ma'lumot	Registrasiya ilingani
№1030	Topilgan narsaga zaptos berish	Topilgan obektmi tavurdagi ko'rinishi	-	-
№1040	idirish so'rovini sturukturasini anilash	Topilgan obektga zaptos berish	Binoga zaptos berish	Zaptosni chiarish
№1050	Kitob fondi bo'yicha idirish	idirish so'rovini sturukturasini anilash	Bino summasi	EB ni Kitob fondi chiarish
№1060	Binoni kattalidi bo'yicha idirish	Kitob fondi bo'yicha idirish	-	-
№1070	idirilgan parameter bo'yicha chiarish	E bazadan 3ds max ma'lumotlarini chiarish	Kalit so'zni yozish	EB ni kalit so'zi bo'yicha chiarish

№1080	Ma'lumotni chiarish bloki	Kiritilalarni ochib berish	-	-
№1090	interfes	idirib topilgan EK ni chiarib berish	-	Eb chiarish
№2010	3ds max Ma'lumotlarni bazaga kiritish	bergan zaprosni chiarib berish	-	Zaprosni chiarish

2.2. Ma'lumotlar ombari haqida tushuncha

Keng ma'noda Ma'lumotlar ombori (MO) deganda real dunyoning konkret ob'ektlari xaidagi ma'lumotlar to'plamini tushunish mumkin. Lekin ma'lumotlar xajmi oshib borishi bilan bu masalalarni xal etish murakkablashadi. Yuzaga kelgan muammo ob'ekt va ma'lumotlarni strukturalash, ya'ni tizimga solish yo'li bilan xal ilinadi. Ob'ekt — bu mavjud va farlanishi mumkin bo'lgan narsadir. Ob'ektlarga tegishli bir ator ma'lumotlar borki, ularning to'plami MO bo'la oladi. Masalan, har bir akademik litsey yoki kasb-xunar kolleji — bu ob'ektlar bo'lsa, ulardagi o'kuvchilar xaidagi ma'lumotlar to'plami MOga misol bo'la oladi.

Har anday jiddiy MOning yaratilishi uning loyixasini tuzishdan boshlanadi. MO loyixalovchisining asosiy vazifasi ob'ektlar va ularni tavsiflovchi parametrlarni tanlash, ma'lumotlar orasidagi bog'lanishlarni o'rnatishdan iborat.

MOni yaratish jarayonida, foydalanuvchi ma'lumotlarni turli belgilar buyicha tartiblashga va belgilarning turli birikmalari buyicha zarur ma'lumotlarni (tanlanmani) tez topish uchun imkoniyatlar yaratilishiga harakat iladi. Bu ishlarni ma'lumotlar strukturalangan (tuzilmalangan) bo'lgandagina bajarish mumkin.

Strukturalash — bu ob'ektlar va ma'lumotlarning o'zaro bog'lanishini tasvirlash usullari xaidagi kelishuvni kiritishdir.

1-misol: Strukturalanmagan ma'lumotlar.

Shaxsiy ish N 16493; **Aliev Karim Ergashevich**; tug'ilgan sana 1 yanvar 1979 yil; Shaxsiy ish N 16498; **Boiev Dilmurod Raxmatullaevich**; tug'ilgan sana 5 dekabr 1985 yil; Shaxsiy ish N16595; **Zokirov Anvar Rashidovich**; tug'ilgan sana 15 may 1984 yil.

2-misol: Strukturalangan ma'lumotlar.

Шахсий иш №	Фамилияси	Исми	Отасининг исми	Тугилган сана
16493	Алиев	Карим	Эргашевич	01.01.79
16498	Бокиев	Дилмуров	Рахматуллаевич	05.12..85
16598	Зкиров	Анвар	Рашидович	15.05.84

Zamonaviy MO texnologiyasida MOni yaratish, unga xizmat ko'rsatish va foydalanuvchilarni MO bilan ishlashga imkon yaratish maxsus dasturiy uskunalar yordamida amalga oshiriladi. Bunday dasturiy uskunalar majmuasi ma'lumotlar omborini bosharish tizimlari (MOBT) deb ataladi.

MOBT — MOni yaratish, uni dolzarb holatda ushlab turish, kerakli axborotni topishni tashkil etish va bosha xizmat ko'rsatish uchun zarur bo'ladigan dasturiy va til vositalari majmuasidir.

MOBT misoli sifatida uyidagilarni keltirish mumkin:

- **DBASE dasturi**;
- **Microsoft Access**;
- **Microsoft Fox Pro for DOS**;
- **Microsoft Fox Pro for WINDOWS**;
- **Paradox for DOS**;
- **Paradox for WINDOWS**.

MO bilan ishlashga kirishishdan oldin ma'lumotlarni tasvirlash modelini tanlab olish kerak. U uyidagi talablarga javob berishi lozim:

- axborotni ko'rgazmali tasvirlash;
- axborotni kiritishda soddalik;
- axborotni izlash va tanlashda ulaylik;
- bosha omborga kiritilgan ma'lumotdan foydalanish imkoniyatining mavjudligi;
- MOning ochiligini ta'minlash (yangi ma'lumotlar va maydonlar ushish, ularni olib tashlash imkoniyatlari va xokazo).

MO bitta yoki bir nechta modellarga asoslangan bo'lishi mumkin. Har anday modelga uzining xossalari (parametrlari) bilan tavsiflanuvchi ob'ekt sifatida arash mumkin. Shunday ob'ekt ustida biror amal (ish) bajarsa bo'ladi. MO modellarining uchta asosiy turlari mavjud: relyatsion, ierarxik va semantik tarmo.

Relyatsion (lotin tilidagi relatio — munosabat suzidan olingan) modelda ma'lumotlarni salash uni tashkil etuvchi ismlari orasidagi munosabatlarga asoslangan. Eng sodda xolda u ikki ulchovli massiv yoki jadvaldan iborat bo'ladi. Murakkab axborot modellari ana shunday jadvallarning o'zaro bog'langan to'plamidan iborat.

MOning ierarxik modeli pastki pogonadagi yuori pogonadagiga buysinish tartibida joylashgan elementlar to'plamidan iborat bo'ladi va agdarilgan daraxt(graf)ni tashkil etadi. Ushbu model satx, tugun, bog'lanish kabi parametrlar bilan tavsiflanadi. Uning ishlash tamoyili shundayki, uyi satxdagi bir nechta tugunlar boglanish yordamida yuoriro satxdagi bitta tugun bilan boglangan bo'ladi. Tugun — bu ierarxiyaning berilgan satxida joylashgan elementning axborot modelidir.

MOning semantik tarmo modeli ierarxik modelga uxshashdir. U xam tugun, satx, bog'lanish kabi asosiy parametrlarga ega. Lekin semantik tarmo modelida turli satxdagi elementlar orasida «erkin», ya'ni «har biri xamma bilan» ma'noli bog'lanish kabul ilingan.

Ko'pchilik MOlar jadval tuzilmasiga ega. Unda ma'lumotlar adresi satr va ustunlar kesishmasi bilan anilaniadi. MOda ustunlar — maydonlar, satrlar esa yozuvlar deb ataladi. Maydonlar MOning tuzilmasini, yozuvlar esa, unda joylashgan ma'lumotlarni tashkil etadi.

Maydonlar — MO tuzilmasining asosiy elementlaridir. Ular ma'lum xususiyatlarga ega bo'ladilar. Har anday maydonning asosiy xususiyati uning uzunligidir. Maydon uzunligi undagi belgilar soni bilan ifodalanadi.

Maydonning yana bir xususiyati, uning nomidir. Maydonda uning nomidan tashkari yana imzo xususiyati xam mavjud. Imzo — ustunning sarlavxasida aks ettiriladigan axborotdir. Uni maydon nomi bilan aralashtirib yubormaslik lozim. Agar imzo berilmagan bo'lsa sarlavxada maydon nomi yozib uyiladi. Turli tipdagi maydonlar turli masadlarda ishlatiladi va turli xossalarga ega bo'ladi.

Maydonlarning xususiyatlari bilan tanishib chiamiz:

1. Oddiy matn maydoni. Belgilar soni 255 dan oshmasligi kerak.
2. MEMO — katta ulchamli matn maydoni. Belgilar soni 65535 dan oshmasligi shart. Oddiy matn va MEMO maydonida xisob ishlarini bajarib bo'lmaydi.
3. Sonli maydon. Sonli ma'lumotlarni kiritishga xizmat iladi va xisob ishlarini bajarishda foydalaniladi. Bu maydon 1,2,4,8 va 16 baytli bo'lishi mumkin.
4. Sana va vat maydoni. Bu maydon sana va vatni bichimlangan xolda salab o'yish imkonini beradi (01.06.01 20:29:59). 8 bayt ulchamga ega.
5. «Pul birligi» nomi bilan ataluvchi maydon. Bu maydondan xisob-kitob ishlarini yuritishda foydalaniladi.
6. Xisoblagich maydoni. Bu maydon 4 bayt uzunlikka va avtomatik ravishda ma'lum songa oshib borish xususiyatiga ega. Ushbu maydondan yozuvlarni nomerlashda foydalanish ulaydir.
7. Mantiy amal natijasini salovchi maydon. Bu maydon «rost» (true) yoki «yolg'on» (false) iymatni salaydi. Maydon ulchami 1 bayt.
8. OLE — nomi bilan yuritiluvchi maydon. Bu maydon Excel jadvalini, Word xujjatini, rasm, ovoz va bosha shu kabi ma'lumotlarni ikkilik sano sistemasida salaydi. Maydon ulchami 1 G baytgacha.
9. Gipersilka maydoni. Bu maydon belgi va sonlardan iborat bo'lib, biror fayl yoki saytga yo'l ko'rsatadi.
10. iymatlar ro'xatidan iborat bo'lgan maydon. Bu maydon bir ancha iymatlardan iborat bo'lgan ruyxatdan tanlangan ani bir iymatni salaydi.

Jadvallar orasidagi munosabatlar ishonchli ishlashi va bir jadvaldagi yozuv orali ikkinchi jadvaldagi yozuvni topish uchun jadvalda aloxida maydon — unikal maydon bo'lishini ta'minlash kerak.

Unikal maydon — bu iymatlari takrorlanmaydigan maydondir.

Microsoft Office keng taralgan ofis ishlarini avtomalashtiruvchi dasturlar paketidir. Uning tarkibiga kiruvchi Access nomli dasturlar majmuasi hozirda MOBT sifatida keng o'rganilmoda va o'llanilmoda.

MOning dastlabki oynasi sodalligi va tushunarligi bilan ajralib turardi. Undagi oltita ilova, dastur ishlaydigan oltita obektni tasvirlaydi. Bular «Таблицы» (jadvallar), «Запросы» (so'rovlar), «Формы» (shakllar), «Отчеты»(hisobotlar),«Макросы» (makrosslar), «Модули»(modullar).

Ularning har biri haqida issledovaniyacha to'xtalib o'tamiz;

1-«Таблицы» (jadvallar)-MOning asosiy obekti. Unda malumotlar salanadi.

2-«Запросы»(so`rovlar)-bu ob'ekt ma'lumotlarga ishlov berish, jumladan, ularni saralah, ajratish, birlashtirish, o`zgartirish kabi vazifalarni bajarishga mo`ljallangan.

3-«Формы» (shakllar)-bu ob'ekt malumotlarni tartibli ravishda oson kiritish yoki kiritilganlarni ko`rib chiish imkonini beradi. Shakl tuzilishi bir ancha matnli maydonlar, tugmalardan iborat bo`lishi mumkin.

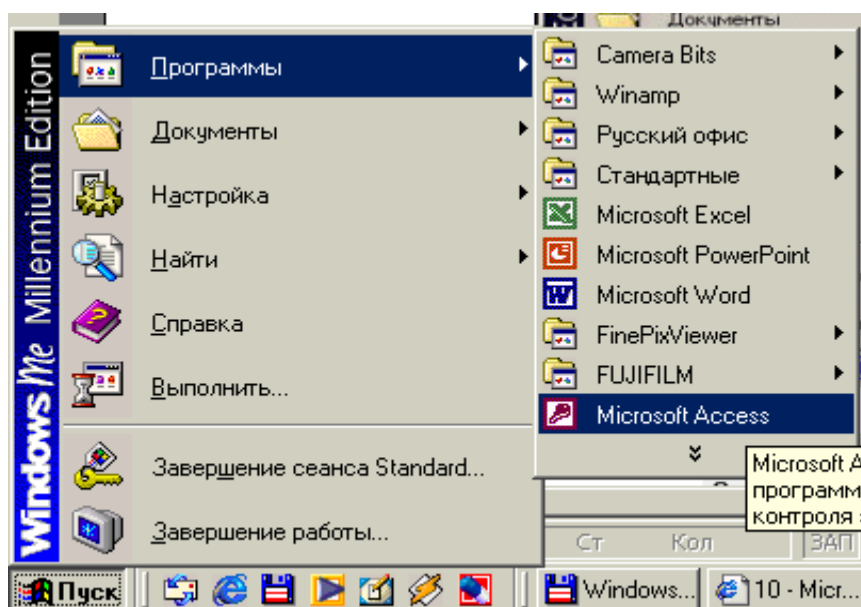
4-«Отчети»(hisobotlar)-bu ob'ekt yordamida saralangan malumotlar ulay va ko`rgazmali ravishda og`ozga chop etiladi.

5-«Макроси»(makroslar)-makrobuyrulardan iborat ob'ekt. Murrakab va tez-tez murojat ilinadigan amallarni bitta makrosga guruxlab, unga jaratilgan tugmacha belgilanadi va ana shu amallarni bajarish o`rniga ushbu tugmacha bosiladi. Bunda amallar bajarish tezligi oshadi.

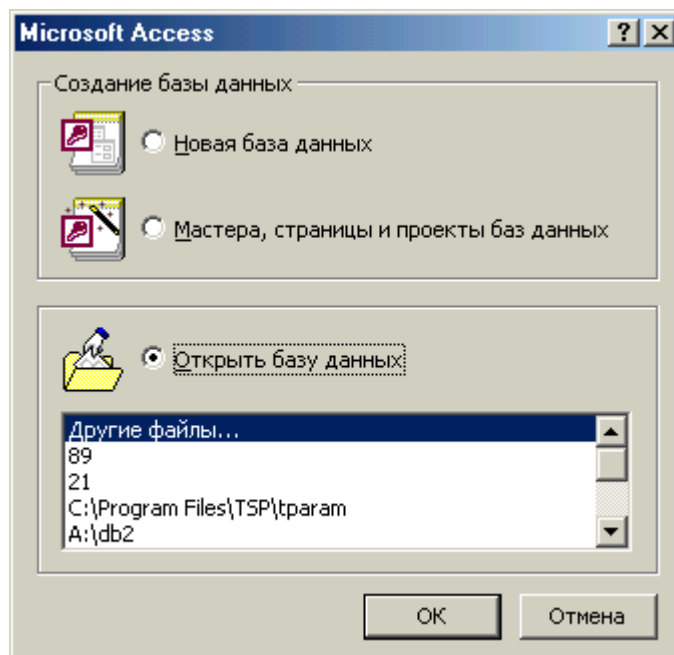
6-«Модули» (modullar)-Microsoft Access dasturining imkoniyatini oshirish masadida ichki Visual Basic tilida yozilgan dasturlarni o`x ichiga olivchi ob'ekt.

Bundan tashari «Страницы»(saxifalar) nomli alohida ob'ekt ham mavjud. Bu ob'ekt HTMLkodida bajarilgan, WEB-sahifada joylashtiriladigan va tarmo orali mijozga uzatiladigan alohida ob'ektdir.

Biror ma'lumotlar omborini loyihalash va yaratish uchun Microsoft Access dasturini ishga tushirish kerak.Buning uchun WINDOWS oynasining masalalar panelidagi < Пуск >tugmachasi ustiga sichoncha ko`rsatkichini olib borib chap tugmachasini bosamiz va < Программы > bo`imiga o`tib, Microsoft Access ismini tanlab olamiz.



Dastur ishga tushgandan keyin uyidagi oyna xosil bo`ladi (10.4-rasm):



MOning dastlabki oynasida yuorida sanab o'tilgan 6 ta asosiy ob'ektlarning ilovalaridan tashkari, yana 3 ta buyruk tugmachalari mavjud. Bo'lar: **«Открыт»** (Ochish), **«Конструктор»** (Tuzuvchi), **«Создат»** (Yaratish) tugmachalaridir (10.2-rasm).

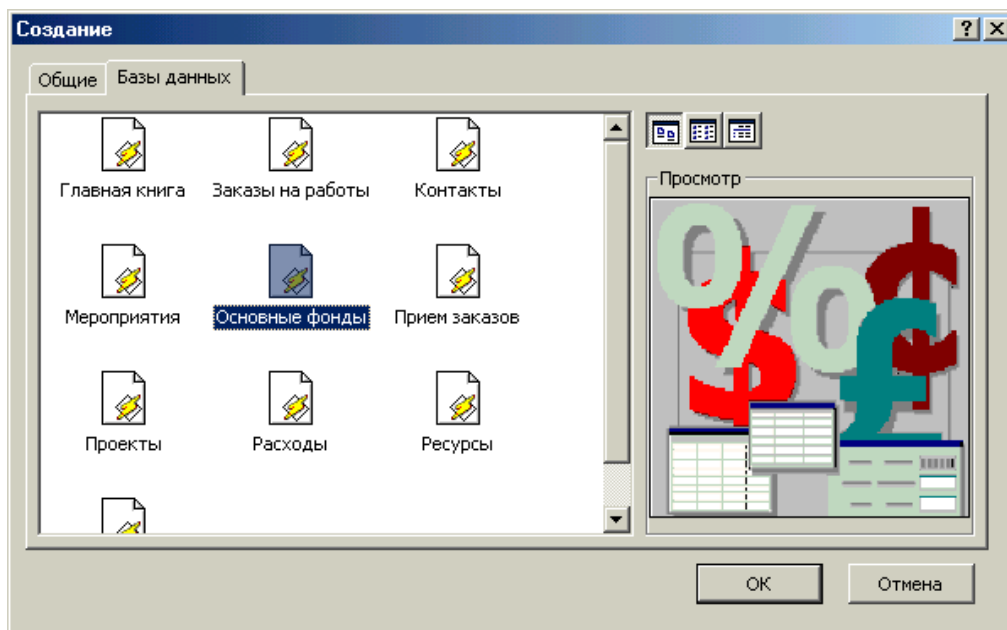
«Открыт» (Ochish) tugmachasi tanlangan ob'ektni ochadi. **«Конструктор»** (Tuzuvchi) xam tanlangan ob'ektni ochadi, lekin u ob'ektning tuzilmasinigina ochib, uning mazmunini emas, balki tuzilishini tug'rilash imkonini beradi. Agar ob'ekt jadval bo'lsa, unga yangi maydonlar kiritish yoki mavjud maydonlarning xossalarini o'zgartirish mumkin. **«Создат»** (Yaratish) tugmachasi yangi ob'ektlarni: jadvallar, so'rovlar, shakllar va xisobotlarni yaratish uchun ishlatiladi.

Biror MOni yaratishdan oldin albatta uning loyixasini ishlab chiish lozim. Buning uchun MOning tuzilmasini anilab olish kerak bo'ladi. MOning yaxshi tuzilmasi talablarga mos keladigan, samarali MOni yaratish uchun asos bo'ladi.

MS Accessda MOni yaratishning ikki usuli mavjud. Ulardan biri bush bazani yaratib, so'ngra unga jadvallar, shakllar, xisobotlar va boshqa ob'ektlarni kiritishdan iborat. Bu usul ancha engil va ulay bo'lgani bilan MOning har bir elementini aloxida anilashga tugri keladi. Shuning uchun ikkinchi usuldan Ko'pro foydalanishadi. Unda **«Мастер»** (Usta) yordamida barcha kerakli jadvallar, shakllar va xisobotlarga ega bo'lgan ma'lum turdagi MO birdaniga yaratiladi, sungra tegishli o'zgartirishlarni bajarish mumkin. Bu boshlang'ich MOni yaratishning eng sodda usulidir.

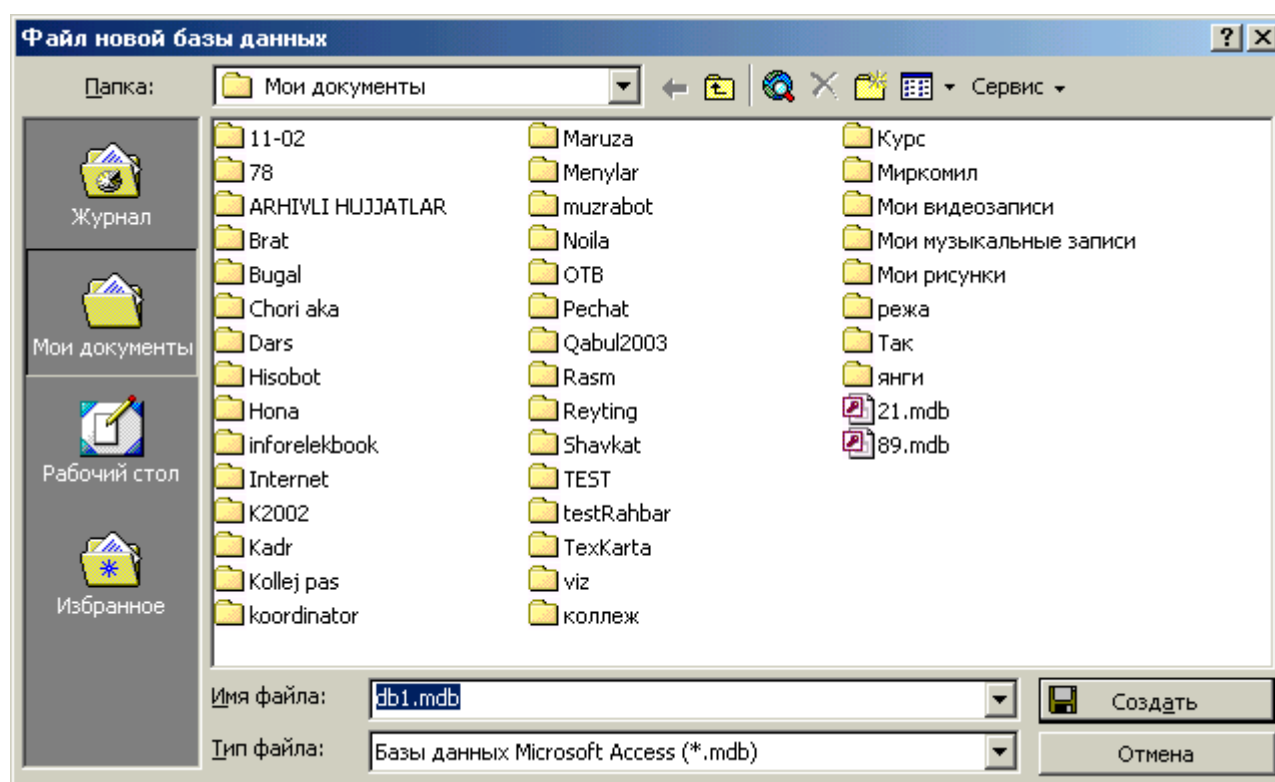
MOni **«Мастер»** (Usta) yordamida yaratish.

MS Access ishga tushirilgandan keyin paydo bo'lgan oynadan (10.4-rasm) **«Запуск мастера»** (Ustani ishga tushirish) parametrini tanlab, OK tugmachasini bosamiz. Agar MO oldindan ochilgan bo'lsa yoki dastlabki muloot oynasi yopilgan bo'lsa, uskunarlar panelidagi **«Создат базу данных»** (MOni yaratish) tugmachasini bosish kerak.



2. Sichoncha ko'rsatkichini kerakli MONing shabloni (andazasi) ustiga joylashtirib, chap tugmachasini ikki marta bosish kerak (10.5-rasm).

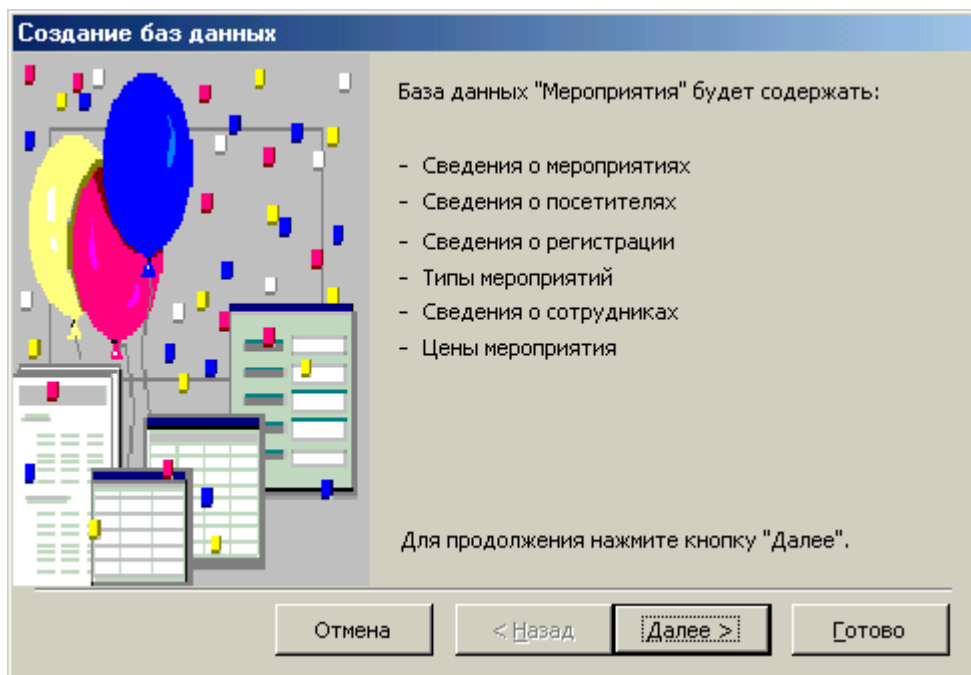
3. Ochilgan «Файл новый базы» (Yangi baza fayli) muloot oynasidagi «Папка» (Jild) ruyxatidan, yaratilayotgan MONi salab uymochi bo'lgan papkani tanlash, «Имя файла» (Fayl nomi) maydonida MONing nomini kiritish va «Создат» (Yaratish) tugmachasini bosish kerak (10.6-rasm).



4. Keyingi muloot oynasida Usta yaratilayotgan MO anday axborotni salashi kerakligi xaida ma'lumot chiqaradi. Ushbu muloot oynasining uyi ismida o'yidagi tugmachalar joylashgan:

«Отмена» (Bekor ilish) — Ustaning ishini tuxtatadi;

«Назад» (Oraga) — Usta ishida bitta oldingi kadanga kaytadi;

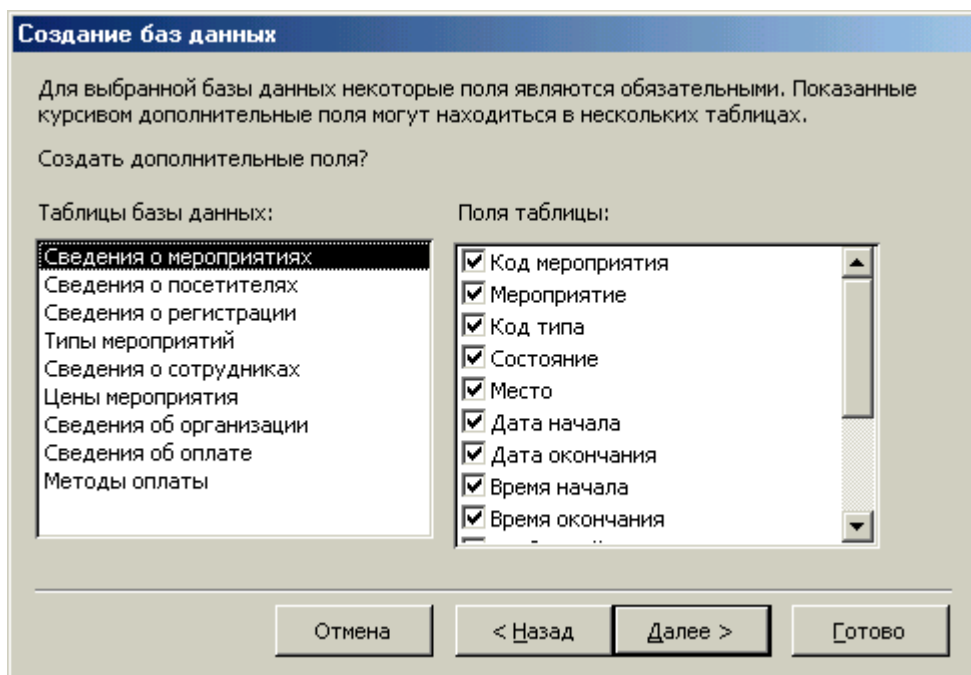


«Далее» (Keyinga) — Usta ishida keyingi kadamga utadi;

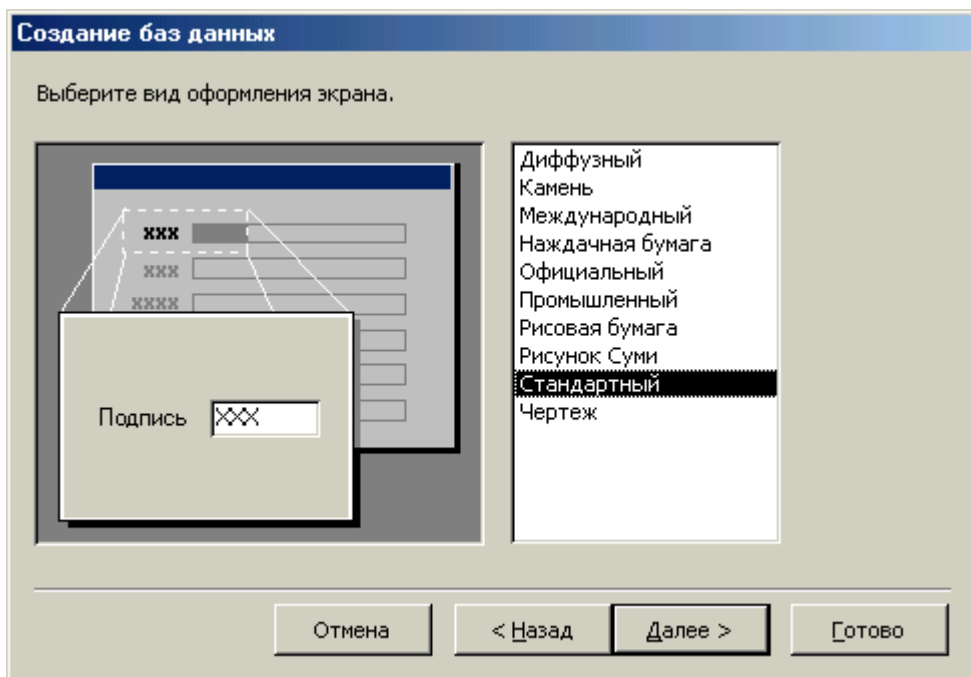
«Готово» (Tayyor) — Tanlangan parametrlı MOni yaratish ustasini ishga tushiradi. Ushbu tugmachani bosishdan oldin MOda salanadigan axborot ekranga chıariladi (10.7-rasm).

5. Ishni davom ettirish uchun «Далее» (Keyinga) tugmachasi bosiladi.

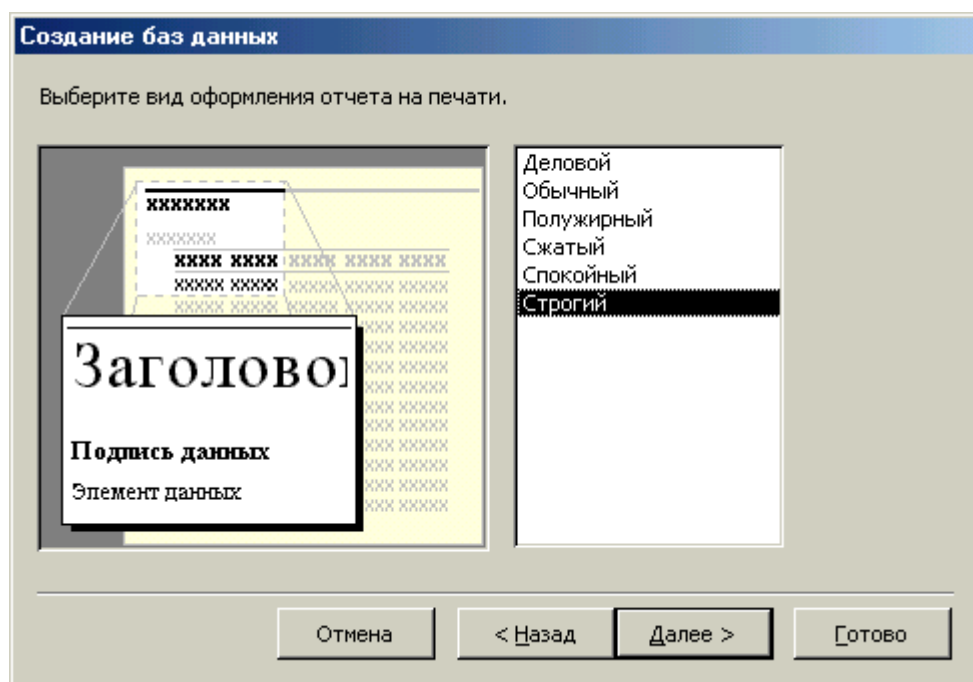
6. Ochiladigan muloot oynasi (10.8-rasm) ikkita ruyxatdan iborat bo'ladi.



Ulardan biri MO jadvallari ruyxati, ikkinchisi — tanlangan jadvalning maydonlari ruyxati. Ushbu ruyxatda jadvalga kiritilayotgan maydonlar belgilangan bo'ladi. Odatda deyarli barcha maydonlar belgilanadi (juda kam ishlatiladigan maydonlardan tashkari). Maydonchalar uchun bayrocha belgisini ('-belgisi) o'rnatish yoki olib tashlash bilan jadvalga maydonlarni kiritish yoki kiritmaslik mumkin. Shundan sung «Далле» (Keyinga) tugmachasini bosish kerak.

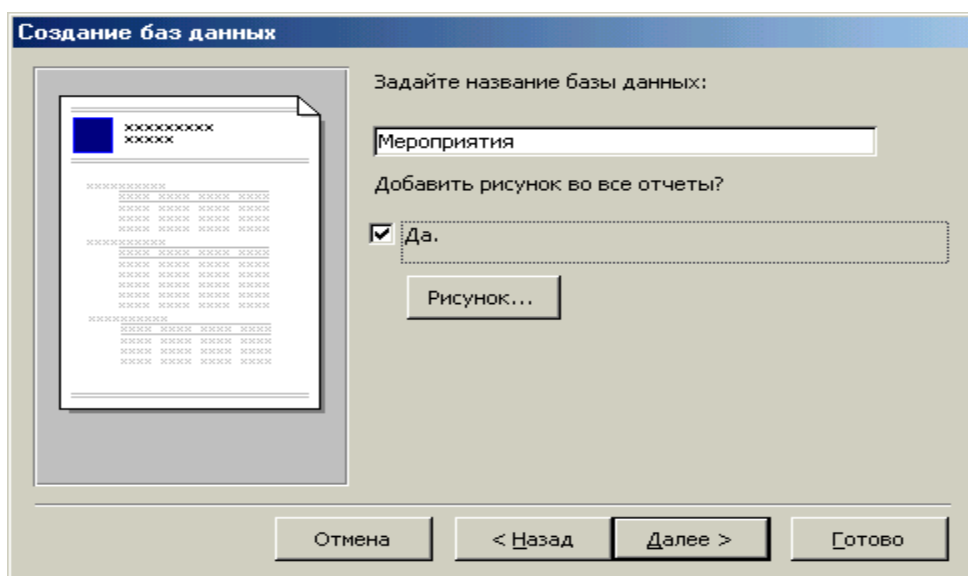


7. Ustaning keyingi adamida taklif ilinayotgan namunalardan ekranni jixozlashni tanlab olish va yana «Далее» (Keyinga) tugmachasini bosish kerak (10.9-rasm).



8. Usta ishining keyingi bosichida MO uchun yaratilayotgan xisobotlar Ko`rinishini anilash mumkin.

9. Ochilgan navbatdagi muloot oynasi xisobotga sarlavxa o'yish va rasm belgilash imkonini beradi (10.10 va 10.11-rasmlar). Ular keyingi barcha xisobotlarda tegishli joyda paydo bo'ladi. Agar rasm kerak bo'lsa «Да» (Ua) yozuvining oldiga bayrocha o'rnatish kerak. Unda «Рисунок» (Rasm) tugmachasini ishlatish mumkin bo'ladi. Bu tugmacha bosilganda «Выбор рисунка» (Rasmni tanlash) oynasi ochiladi.

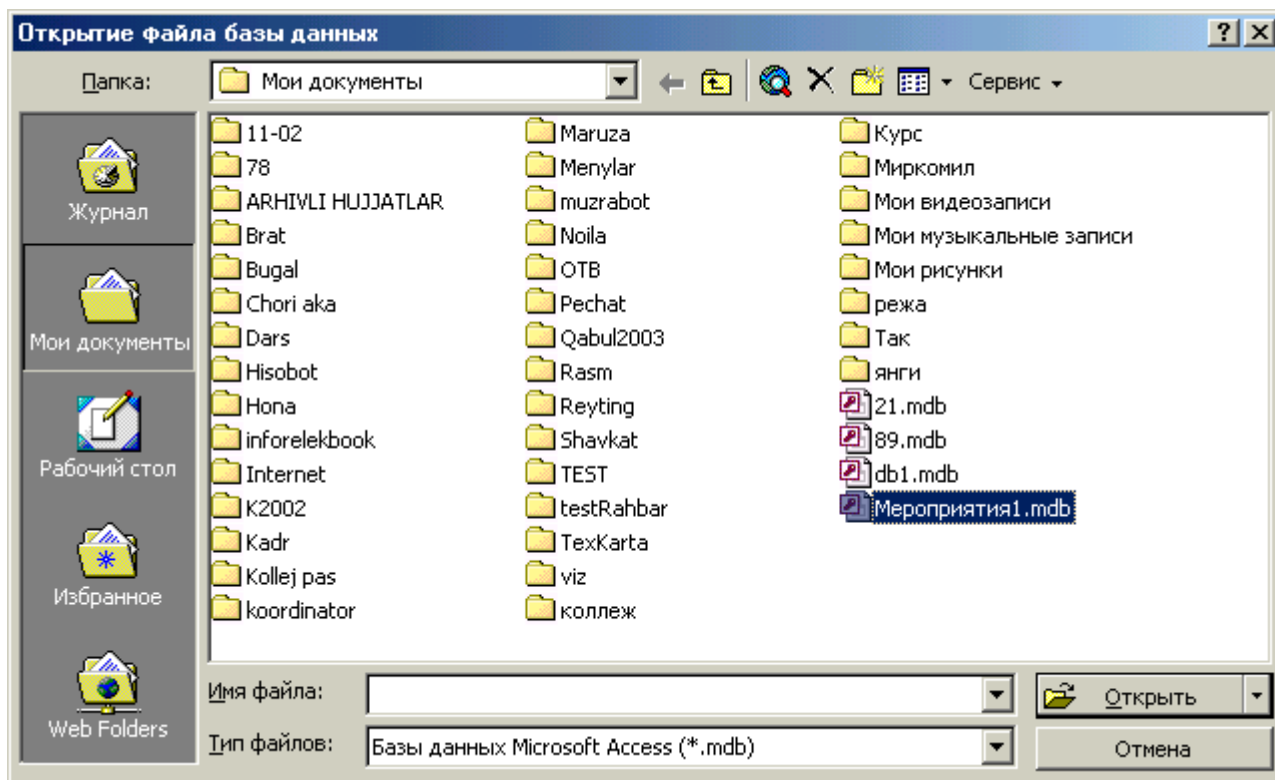


10. Oxirgi oynada «Готова» (Tayyor) tugmachasini bosish ustani MOni tuzish uchun ishga tushirib yuboradi va u avtomatik ravishda yuorida belgilangan parametrlri MOni yaratadi.

Yangi ma'lumotlar omborini Ustaning yordamisiz, mustail ravishda yaratish mumkin. Buning uchun MS Access ishga tushirilgandan keyin paydo bo'lgan oynadan «Новая база данных» (Yangi MO) parametrini tanlab, OK tugmachasini bosamiz. Agar MO oldindan ochilgan bo'lsa yoki ishga tushirish oynasi yopi bo'lsa, uskunalar panelidagi «Создат базу данных» (MOni yaratish) tugmachasini bosish va sichoncha ko'rsatkichini yangi MO belgisi ustiga olib borib, tugmachasini ikki marta bosish kerak. Shundan sung «Имя файла» (Fayl nomi) atoriga ombor nomini yozamiz va «Создат» (Yaratish) tugmachasini bosamiz. Natijada bo'sh bo'lgan MO tanasini xosil ilamiz.

MOni ochishning ikki usuli mavjud. Uni Access MOBTni ishga tushirish jarayonida yoki u bilan ishlash jarayonida ochish mumkin.

MOni Access bilan ishlash jarayonida ochish uchun «Файл» menyusida «открыт» (Ochish) buyrugini tanlash kerak. Shundan so'ng ochilgan oynadan foydalanib (10.12-rasm), uyidagi ishlar bajarilishi kerak:



1. Adreslar panelida yorli ustida sichoncha belgisini joylashtirib tugmachasini bosish yoki «**Папка**» (Jild) maydonida kerakli MO joylashgan disk yoki jildni tanlash.
2. Jildlar ruyxatida kerakli jild ustida ikki marta sichoncha tugmachasini bosib, MO joylashgan jildni ochish.

Agar kerakli MO topish imkoni bo'lmasa «Сервис» tugmachasini bosish va «**Найти**» (Topish) buyrugini tanlash kerak. «**Найти**» (Topish) muloot oynasida izlash uchun kushimcha shartlarni kiritish, so'ngra kerakli parametr ustida sichoncha tugmachasini bosish kerak. MOni fakat ukish, ya'ni taxrirlamasdan ko'rib chiish uchun ochganda «**Открыт**» (Ochish) tugmachasi yonidagi strelkali tugmachani bosish kerak va «Открыт для чтения» (Uish uchun ochish) variantini tanlash lozim. Access MOBTni ishga tushirishda ekranda muloot oynasi paydo bo'ladi. Buni siz yaxshi bilasiz. Undagi «**Открыт базу данных**» (MOni ochish) bo'limini tanlash va taklif etilayotgan barcha mavjud Molar ruyxatidan kerakli MOni sichoncha tugmachasini MOning yozuvi va nomi ustida bosish bilan ochish mumkin.

Bizga ma'lumki, MOni mustail ravishda yaratganda u Bo'sh bo'ladi. Uning ob'ektlarini foydalanuvchining o'zi yaratishiga tug'ri keladi. uyida biz MO ob'ektlaridan jadvallar, so'rovlar, shakllar va xisobotlar yaratishni ko'rib chiamiz.

Jadvallar yaratish

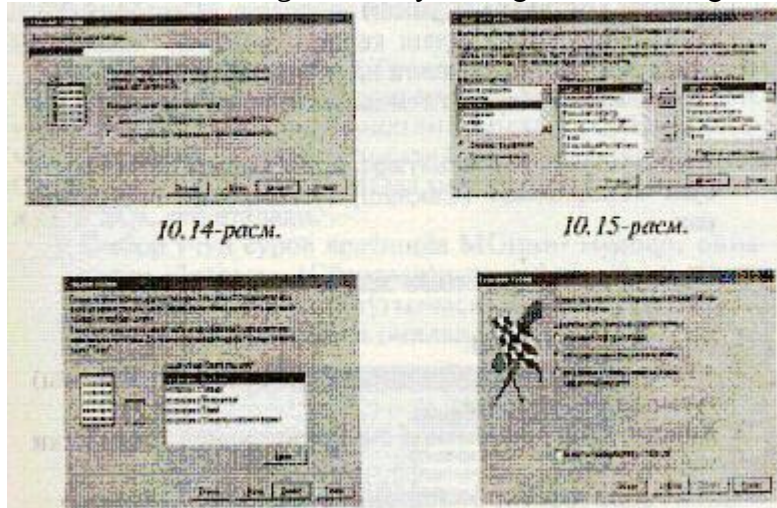
Bush jadval yaratish. Microsoft Accessda bush jadvalni yaratishning to'rt usuli mavjud:

- MOni tulaligicha yaratadagan MOning ustasini ullah. Bunday usta yangi omborni yaratadi, xolos. Uning yordamida MOga yangi jadvallarni, shakllarni yoki xisobotlarni o'shib bo'lmaydi.
- Jadvallar ustasi oldindan anilangan jadvallardan yaratilayotgan jadval uchun maydonni tanlash imkonini beradi.
- Jadval holatida ma'lumotlarni bevosita bo'sh jadvalga kiritish.
- Конструктор holatida jadval maketining barcha parametrlarini anilash.

Jadval yaratishda ishlatilgan usulning turidan kat'i nazar, har doim jadval maketini o'zgartirish, masalan, yangi maydonlar ushish, iymatlarini o'rnatish va boshalarni bajarish uchun Конструктор holatini o'llash imkoniyati mavjud.

Jadvallar ustasi yordamida jadval yaratish

1. MO oynasiga utish. Bir oynadan ikkinchi oynaga utish uchun F11 tugmachasini bosish kerak.
2. «Таблица» (Jadval) ilovasida «Создат» (Yaratish) tugmachasini bosish.
3. «Тастер таблица» (Jadvallar ustasi) elementida sichoncha tugmachasini ikki marta bosish. Natijada uyidagi oyna xosil bo'ladi (10.13-rasm):
4. Jadvallar ustasining muloot oynasidagi ko'rsatmalarga rioya ilish (10.14-10.17 rasmlar).

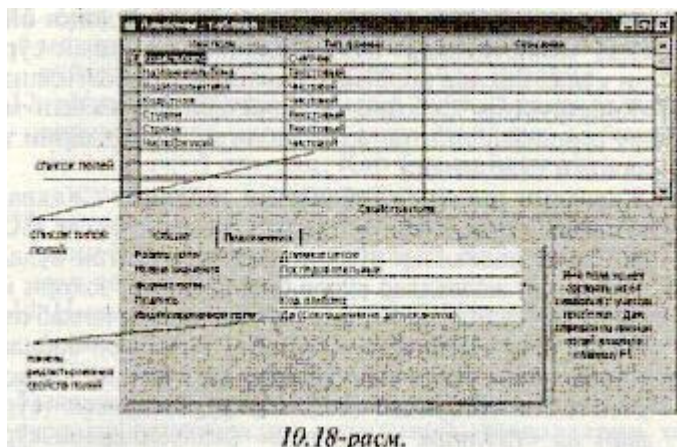


Eslatma: kerak bo'lgan xolda jadvallar ustasi bilan ish tugagandan sung, xosil bo'lgan jadvalni Конструктор holati yordamida uzgartirish yoki kengaytirish mumkin.

1. MO oynasiga o'tish.
2. «Таблица» (Jadval) ilovasida «Создат» (Yaratish) tugmachasini bosish.
3. «Режим таблицы» (Jadval holati) elementida sichoncha tugmachasini ikki marta bosish. Natijada ekranda 20 ta ustun va 30 ta satrdan iborat bush jadval xosil bo'ladi. Aloxida ko'rsatma berilmasa, ustunlar «Поле 1», «Поле 2» va xokazo nomlarini oladi.
4. Har bir ustun nomini o'zgartirish uchun uning oldingi nomi ustida sichoncha tugmachasini ikki marta bosish, yangi nomni ularga uyiladigan barcha talablarga rioya ilgan xolda kiritish va ENTER tugmachasini bosish kerak.
5. Agar jadval 20 tadan ortik ustunga ega bo'lsa, yangi ustunlarni o'shish mumkin. Buning uchun yangi ustun o'yilishi kerak bo'lgan joyning chap tomonidagi ustunning ung tomonida sichoncha tugmachasini bosish va «Вставка» (uyish) menyusida «Столбец» (Ustun) buyrugini tanlash kerak.
6. Ma'lumotlarni jadvalga kiritish. Bunda har bir ustunga ma'lum turdagi ma'lumotlarni kiritish lozim.
7. Barcha ustunlarga ma'lumotlarni kiritib bo'lgandan sung «Сохранит» (Salash) tugmachasini bosish kerak.

Конструктор holatida jadvalni tez yaratish

1. MO oynasiga utish.
2. «Таблица» (Jadval) ilovasida «Создат» (Yaratish) tugmachasini bosish.
3. Конструктор elementida sichoncha tugmachasini ikki marta bosish.
4. Jadvalda har bir maydonni anilash (10.18-rasm).
5. Kalit maydonlarni anilash.
6. Uskunalar panelidagi «Сохранит» (Salash) tugmachasini bosish, so'ngra jadval nomini kiritish kerak.



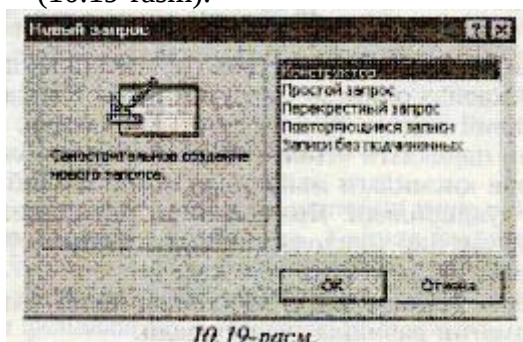
10.18-рasm.

So'rovlar yaratish

Amaliyotda dastlabki yaratilgan jadvaldan yozuvlarning bir ismini (ma'lum mezonlar buyicha) tanlab olish va tartiblash zarurati Ko'plab tutilib turadi. Tanlash mezonlari bir ator shartlar majmuasi bilan anilanishi mumkin.

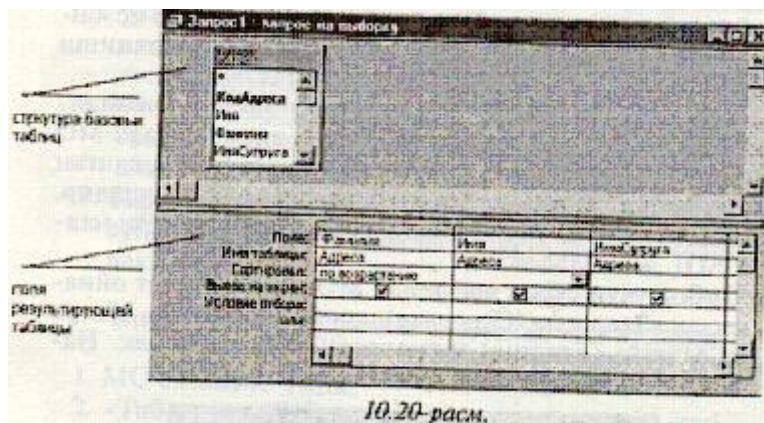
Tanlab olish so'rovlari. So'rovlarning eng Ko'p uchraydigan turi bu tanlab olish so'rovlaridir. Uning maksadi MO ma'lumotlari orasidan berilgan shartlarga buysunadigan ma'lumotlarni tanlab, natijaviy jadvalni chop etishdir. So'rovlarni yaratish uchun MOda maxsus so'rovlar tili mavjud. U SL deb ataladi.

1. Ombor uchun so'rov yaratishda MONing muloot oynasidan «Запросы» (So'rovlar)ilovasini tanlash, sungra «Создат» (Yaratish) tugmachasini bosish kerak. Natijada uyidagi oyna ochiladi (10.19-rasm).



10.19-рasm.

2. Ochilgan «Новые запросы» (Yangi so'rov) muloot oynasida «Конструктор» bo'limini tanlash bilan so'rovni o'lda ishlash holatida yaratish usuli belgilanadi.
3. « Конструктор » holatida so'rovni yaratish, MOdan mana shu so'rov asosini tashkil etadigan jadvallarni tanlashdan boshlanadi.
4. Jadvalni tanlash «Добавление таблицы» (Jadvalni o'shish) muloot oynasida bajariladi. Unda MOdagi barcha jadvallar ruyxati aks ettirilgan bo'ladi.
5. Tanlangan jadvallar so'rov blankasining yo'ori ismiga yozib uyiladi. Buning uchun jadvallar tanlab olingandan sung «Добавит» (o'shish) tugmachasi bosiladi.
6. «Добавление таблицы» (Jadvalni kushish) oynasida uchta ilova «таблица» (jadvallar), «запросы» (so'rovlar) va «таблицы запросы» (jadvallar va so'rovlar) borligiga e'tibor berish kerak.
7. Namuna buyicha so'rov blankasi ikkita panelga ega. Yuori panelda so'rovga asos bo'ladigan jadvallar maydonlarining ruyxati joylashadi (10.20-rasm).



8. uyi panel satrlari so`rov tuzilmasini, ya'ni so`rov natijasida olingan ma'lumotlar joylashadigan natijaviy jadval tuzilmasini anilaydi.

9. uyi paneldagi «Поле» (Maydon) satri blankining yuori ismidagi maydonlar nomini olib utish bilan tuldiriladi. Yaratiladigan natijaviy jadvalning har bir maydoniga namuna buyicha so`rov blankining bitta ustuni mos keladi.

10. «Имя» (Nomi) satri maydonlar olib o'tilayotganda avtomatik ravishda tuldiriladi.

11. Agar «Сортировка» (Saralash) satrida sichoncha tugmachasi bosilsa, saralash turlarini ko'rsatuvchi ruyxatni ochadigan tugmacha paydo bo'ladi.

12. «Условие отбора» (Tanlash sharti) satrida natijaviy jadvalga kiritish uchun yozuvlarga uyiladigan mezon (shart) yoziladi. Uar bir maydon uchun uzining tanlash sharti bo'lishi mumkin.

13. So`rovni ishga tushirish «Вид» (Ko'rinish) tugmachasini bosish orali bajariladi. Unda natijaviy jadval xosil bo'ladi.

14. Natijaviy jadvaldan chiish va namuna buyicha so`rov blankasida yangi so`rovni yaratishga kaytish uchun «Вид» (Ko'rinish) tugmachasini takroran bosish kerak.

**Uzgartirish so`rovlari.** Tanlab olish so`rovlarining barcha turlari vaktinchalik natijaviy jadvallarni xosil iladi. Bunda bazadagi jadvallar uzgarishsiz koladi. Shunga karamasdan MOni yaratuvchilari uchun so`rovlarning maxsus guruxi mavjudki, ular uzgartirish so`rovlari deyiladi. Uzgartirish so`rovlari — bu, bir amalni bajarish bilan bir nechta so`rovlarga uzgartirish kiritadigan so`rovdur. Uning 4 turi mavjud: yuotish, yangilash, yozuvlar kushish va jadval yaratish so`rovlari.

Yuotish so`rovlari bir yoki bir nechta jadvaldan yozuvlar guruxini yuotadi. Yuotish so`rovlari orali yozuvni tulaligicha yuotish mumkin. Uning ichidagi ayrim jadvallarni aloxida yuotib bulmaydi.

Yangilash so`rovlari bir yoki bir nechta jadvaldagi yozuvlar guruxida umumiy uzgartirishlar kiritadi. Ushbu so`rov mavjud jadvallardagi ma'lumotlarni uzgartirish imkonini beradi.

Yozuvlar kushish so`rovi bir yoki bir nechta jadvaldagi yozuvlar guruxini bosha bir yoki bir nechta jadvalning oxiriga kushadi.

Jadval yaratish so`rovi bir yoki bir nechta jadvalning barcha ma'lumotlari yoki ularning bir ismi asosida yangi jadvalni yaratadi.

Hulosa

Ushbu bob o'zidan keyin keladigan bobga tayyorgarlik sifatida yondoshishini bob so'ngida ko'rishimiz mumkin. Bunday deyilishiga sabab, keyingi bobda biz ARM dizayn ko'rinishlarini tanlab oladigan dasturning yaratilishi haqida gap yuritamiz.

Bu bobda ana shu dasturga tayyorgarlik sifatida ARM dizayni uchun qidiruv dasturining funksional strukturasi, masalalar sxemasi va dastur uchun ma'lumotlarni kiritish algoritmlari ishlab chiqildi.

Har hil ko'rinishda yaratilgan ARMLar dizayni ma'lumotlar omborida saqlanadi va shu yerdan dasturga uzatiladi. Bobning ikkinchi qismida esa ana shu ma'lumotlar ombari haqidagi ma'lumotlar bilan nihoya topadi.

3-BOB. ARM DIZAYNI UCHUN BO'LGAN DASTURNI ISHGA TUSHIRISH

3.1. ARM dizayni uchun bo'lgan dasturning tuzilishi.

Dastur ishga tushishi bilan uyidagi oina hosil bo'ladi. Bu uyidagi rasmda keltirilgan (3.1.1-rasm).



3.1.1-rasm.

Ushbu interfeysni yaratish uchun dasturga uyidagi kodlar ketma-ketligi yoziladi.

```
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);  
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;  
this.BackColor = System.Drawing.Color.Gainsboro; // oinaning rangi tanlangan  
this.ClientSize = new System.Drawing.Size(592, 373); // oinaning displaydagi o'lchamlari
```

kiritilgan.

```
this.ControlBox = false;  
this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.FixedDialog;  
this.MaximizeBox = false;  
this.MinimizeBox = false;  
this.Name = "Form1";  
this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen; // dastur ishga  
tushirilganda oinaning dastlabki hosil bo'lish o'rni tanlangan.
```

```
this.Text = "MAIN"; // bu atorda oinaga main deb nomlandi.
```

Main oinasiga button3 hamda Panel1 kontrollari tashlangan.

```
this.Controls.Add(this.button3);  
this.Controls.Add(this.panel1);
```

Button3 bu main oinasini yopish uchun ishlatiladi. Uning dizayni uyigacha yozilgan.

```
this.button3.ForeColor = System.Drawing.Color.Red;  
this.button3.Location = new System.Drawing.Point(554, 0);  
this.button3.Name = "button3";  
this.button3.Size = new System.Drawing.Size(28, 25);
```

```
this.button3.Text = "X";
this.button3.Click += new System.EventHandler(this.button3_Click);
```

Bu tugma bosilganda uyidagi hodisa bajariladi yani oynani yopish uchun shu kodlardan foydalanamiz.

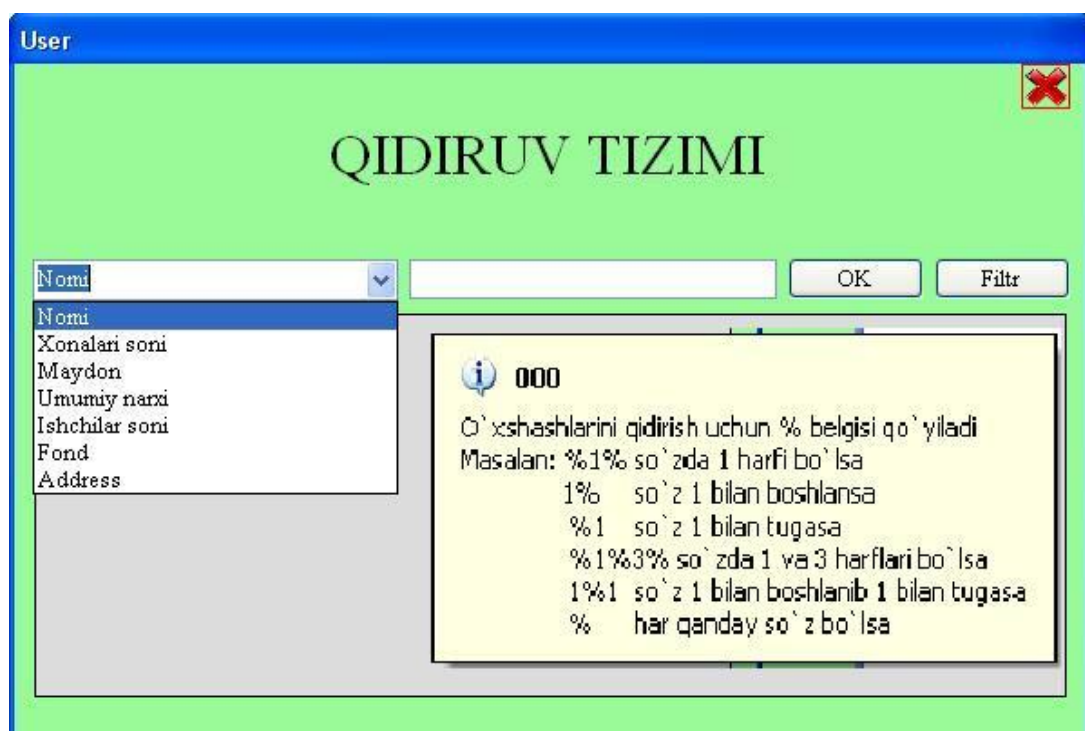
```
private void button3_Click(object sender, EventArgs e)
{
    Close();
}
```

oyna yopildi.

User va Administrator tugmalari Panel1 kontrolining ichiga hosil ilingan bo'lib, ular bosilganda uyidagi hodisa ichidagi kod bajariladi.

```
private void button1_Click(object sender, EventArgs e)
{
    new User().ShowDialog();
}
```

Ushbu hodisa User oinasini ishga tushiradi (3.1.2-rasm).



3.1.2-rasm.

Formadagi “idiruv tizimi” yozuvi chiishi uchun label kontroli kodi uyidagi ko'rinishda yoziladi.

```
this.label1.Location = new System.Drawing.Point(12, 28);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(569, 46);
this.label1.TabIndex = 0;
this.label1.Text = "idiruv tizimi";
this.label1.TextAlign = System.Drawing.ContentAlignment.MiddleCenter;
```

Foydalanuvchi kerakli parametrlar bo'yicha idiruvni tashkil ilib OK tugmasi bosilganda uyidagi hodisa bajariladi.

```

        private void button1_Click(object sender, EventArgs e)
        {
            listBox1.DataSource = null;
            string x = "";
            if (!textBox1.Text.Trim().Equals(""))
            {
                x = "where " + comboBox1.SelectedValue + " like " + textBox1.Text + "";
                listBox1.DisplayMember = "nomi";
                listBox1.ValueMember = "id";
                listBox1.DataSource = kutname.Select("select * from kutname " + x);
            }
        }
    }

```

Bu hodisa yordamida malumotlar bazasiga so'rov yuboriladi.

Berilgan so'rov bo'yicha bizga malumotlar listbox kontrolida chiqarilib beriladi.

Filter tugmasi bosish bilan Filt oinasi ishga tushadi. Buni korinishi ko'rinishi udagicha (3.1.3-rasm).

3.1.3-rasm.

```

private void btnSearch_Click(object sender, EventArgs e)
{
    string shart = "";
    if (!kitobsoni.Text.Equals(""))
        shart += " kitobsoni between " + kitobsoni.Text + " and " + kitobsoni2.Text;

    if (!address.Text.Equals(""))
        shart = PlusAnd(shart, " address=" + address.Text + "");

    if (!tel.Text.Equals(""))
        shart = PlusAnd(shart, " tel=" + tel.Text + "");

    if (!price.Text.Equals(""))
        shart = PlusAnd(shart, " price between " + price.Text + " and " + price2.Text);

    if (!square.Text.Equals(""))
        shart = PlusAnd(shart, " square between " + square.Text + " and " + square2.Text);
}

```

```

if (!shart.Equals(""))
{
    shart = " where " + shart;
    listBox1.DisplayMember = "nomi";
    listBox1.ValueMember = "id";
    listBox1.DataSource = DBmanager.GetData("select * from kutname " + shart);
}
}

```

OK tugmasi yuoridagi hodisani chairadi. Bunda so'rov yordamida ma'lumotlar oraliq'i tanlab olinadi. Dasturning asosiy ismi bu Administrator ismi bo'lim ushbu oinada uyidagi kodlar bilan ifodalanadi. uydagi ko'rinishga ega (3.1.4-rasm).



3.1.4-rasm.

Administrator bo'limiga kirish uchun Login va Password orali kiriladi. Bunga ishlatiladigan kodlar uydagicha.

```

public partial class authen : Form
{
    public authen()
    {
        InitializeComponent();
    }
    public string reuest = "";
    private void ok_Click(object sender, EventArgs e)
    {
        OdbcCommand cmd = new OdbcCommand("select ism from admin where login=" + log.Text
+ " and pwd=" + pas.Text + "");
        string ism = DBmanager.executeCommand(cmd, "ism")[0];
        if (!ism.Equals("Error"))
        {
            reuest = ism;
            Close();
        }
    }
}

```

Parol hamda loginni tekshirish uchun yuoridagi class yaratilgan.

Bazani to'ldiradigan moydonlarni hosil ilish uchun ushbu kodlardan foydalanamiz. Bu interfesni ko'rinishi 3.1.5-rasmda ko'satilgan.

Nomi	
Director	
Fond	
Maydoni	
Umumiy narxi	
Ishchilar soni	
Xonalar soni	
Address	
Telefon	

File Saqlash

3.1.5-rasm.

Salash tugmasi bo'silganda uyidagi kiritilgan ma'lumotlarni bazaga yozadi.

```
private void Save_Click(object sender, EventArgs e)
{
    try
    {
        richTextBox1.Clear();
        kutname k = new kutname();
        k.nomi = "" + dataGridView1[1, 0].Value;
        k.director = "" + dataGridView1[1, 1].Value;
        k.kitobsoni = "" + dataGridView1[1, 2].Value;
        k.square = "" + dataGridView1[1, 3].Value;
        k.price = "" + dataGridView1[1, 4].Value;
        k.ishchisoni = "" + dataGridView1[1, 5].Value;
        k.xonasoni = "" + dataGridView1[1, 6].Value;
        k.address = "" + dataGridView1[1, 7].Value;
        k.tel = "" + dataGridView1[1, 8].Value;
        string x = k.Save();
        if (x.Equals("1"))
        {
            richTextBox1.Text = "Ma'lumot yozildi!";
            if (!label1.Text.Equals(""))
            {
                FileX.CreateDirectory(FileX.GetCurrentDirectory + "\\data\\" + papka(k));
                FileX.Save(label1.Text, nom, FileX.GetCurrentDirectory + "\\data\\" + papka(k) + "\\");
            }
        }
        else
        {
            richTextBox1.Text = "Ma'lumot yozilmadi!\nXatolik:\n" + x;
        }
    }
    catch (Exception exp) {
        richTextBox1.Text += "Ma'lumot yozilmadi!\nXatolik:\n" + exp.Message;
    }
}
```

}

Dasturni asosiy kodlari shulardan iborat bo'lib hosil bo'lgan dasturni islatishimiz mumkin.

3.2. ARM dizayni uchun bo'lgan dasturni ishlatish qo'llanmasi.

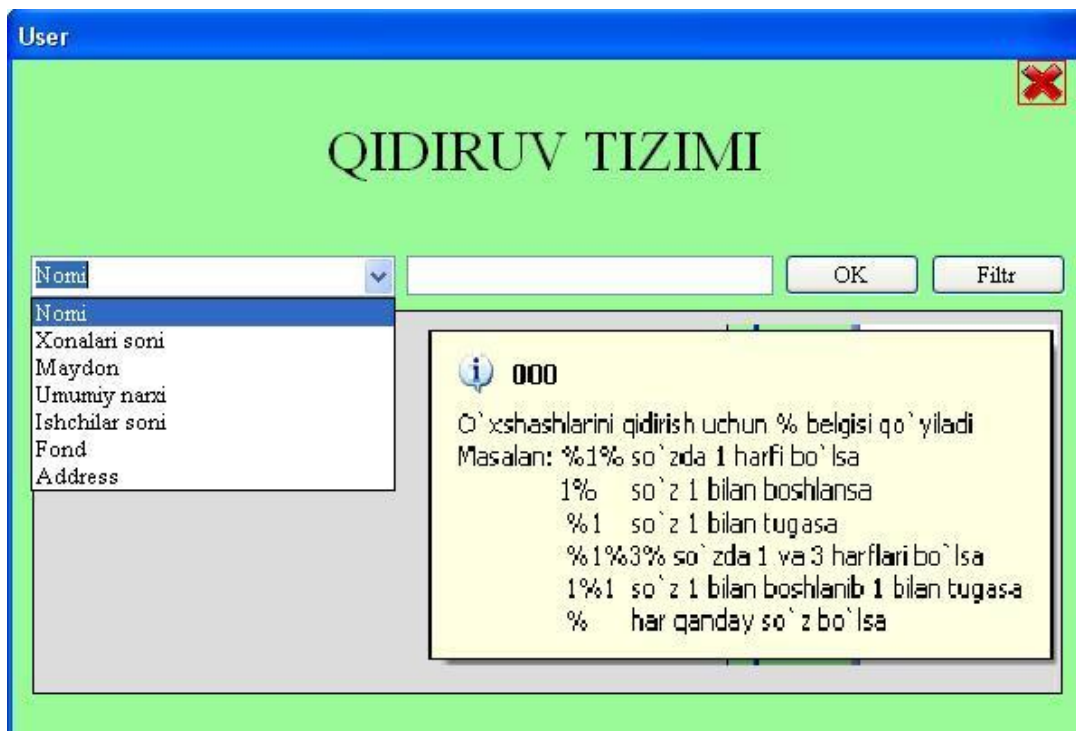
Dastur interfeysi

Eng asosiy oyna Kutubhona loyhasi deb nomlanadi. Bu interfeysda **User** va **Administrator** oynalaridan iborat. User oynasi hamma foydalanuvchilar uchun mo'ljallangan bo'lib u orali oynalarni ochish imkoniga ega bo'lishingiz mumkin. Administrator, bosharuvchi uchun mo'ljallangan bo'lib u albatda Login va Parol so'larini bilishi kerak. Buni uydagi rasmda ham ko'rishingiz mumkin (3.2.1-rasm).



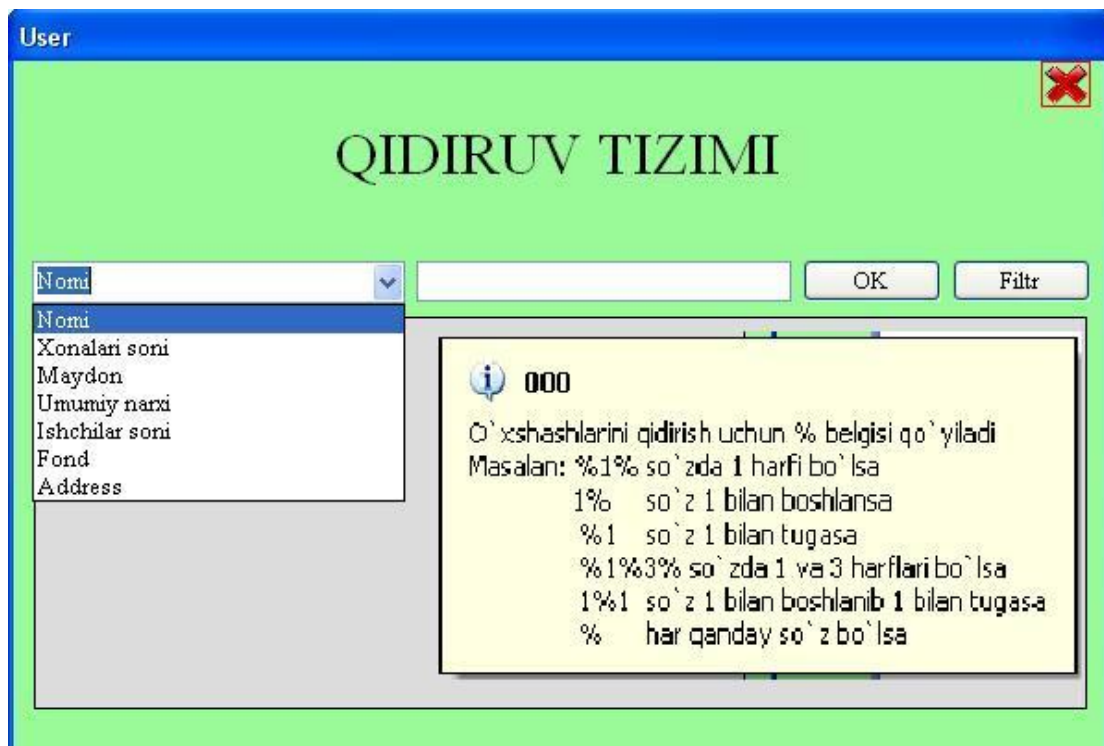
3.2.1-rasm.

User oynasi. Bu ona yordamida foydalanuvchilar o'ziga kerakli dizaynga ega bo'lgan kutubhona rasmlarini yoki kutubhona manzillari, nomini, fondini, telefon raamini vahakoza ma'lumotlarini ko'rishi va olishi mumkin. Buning uchun albatda dasturdan idirish nomini tanlab idirishga beramiz. Buni udagi rasmda ham kurishimiz mumkin (3.2.2-rasm).



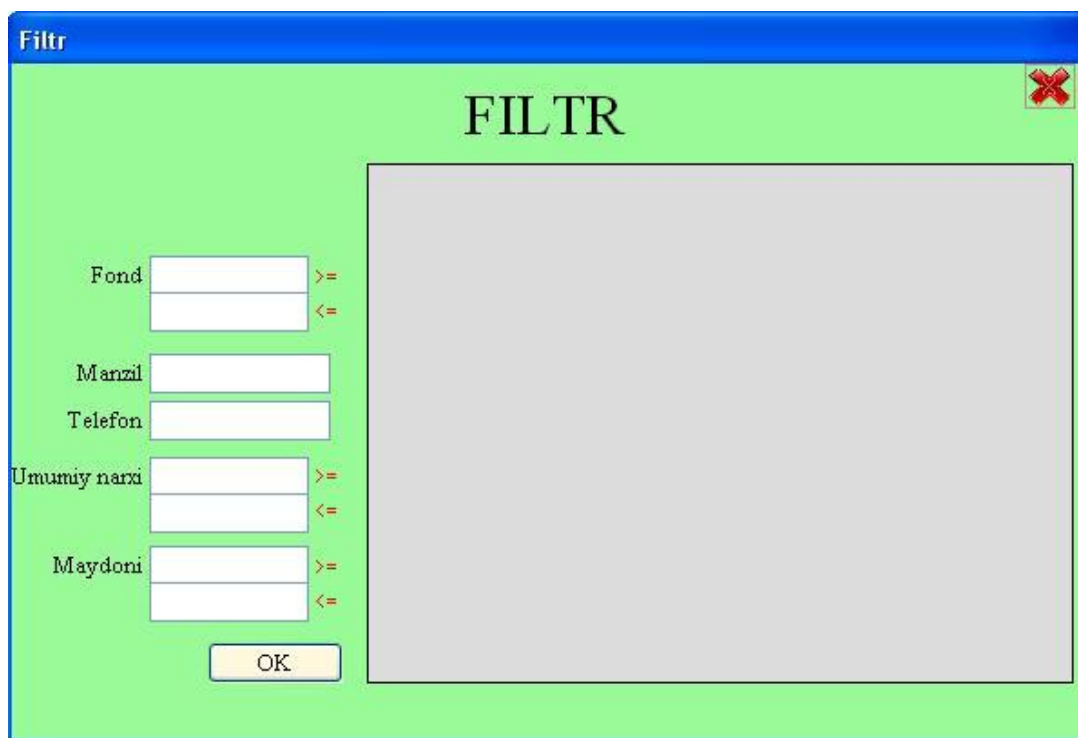
3.2.2-rasm.

iduruv tizimi Nomi, Honalar soni, Maydoni, Umumiy narhi, Ishchilar soni, Fondi va Manzili bo'yicha idiradi. Misol ushun Fond bo'yicha idirish kerak bo'lsa, Fondni belgilab belgilangan joyga kutubhonani fondini kiritamiz. Bu oynadagi idirish tizimida avzallik tomoni o'xshashlarini ham chiarib berishi mumkin. O'shashlarini idirishi uchun biz foyiz (%) bilgisini ishlatishimiz kerak bo'ladi. O'shashlar kerak bo'lmasa ani nomlar yoki iymatlarni ham kiritib topishimiz mumkin. O'shashini topishi uchun ham ikki hil usuli oldi ismi va tugagan ismiga foyiz (%) o'yib idirish mavjud. Masalan: Kutubhona ellik ming fontga ega bo'lsin uni uydagi ko'rinishda yozamiz 50000% bu bizga ellik ming fontga ega bo'lgan va shunga o'shash bo'lgan iymatni chiarib beradi. uydagi rasmda ko'satilgan (3.2.3-rasm).



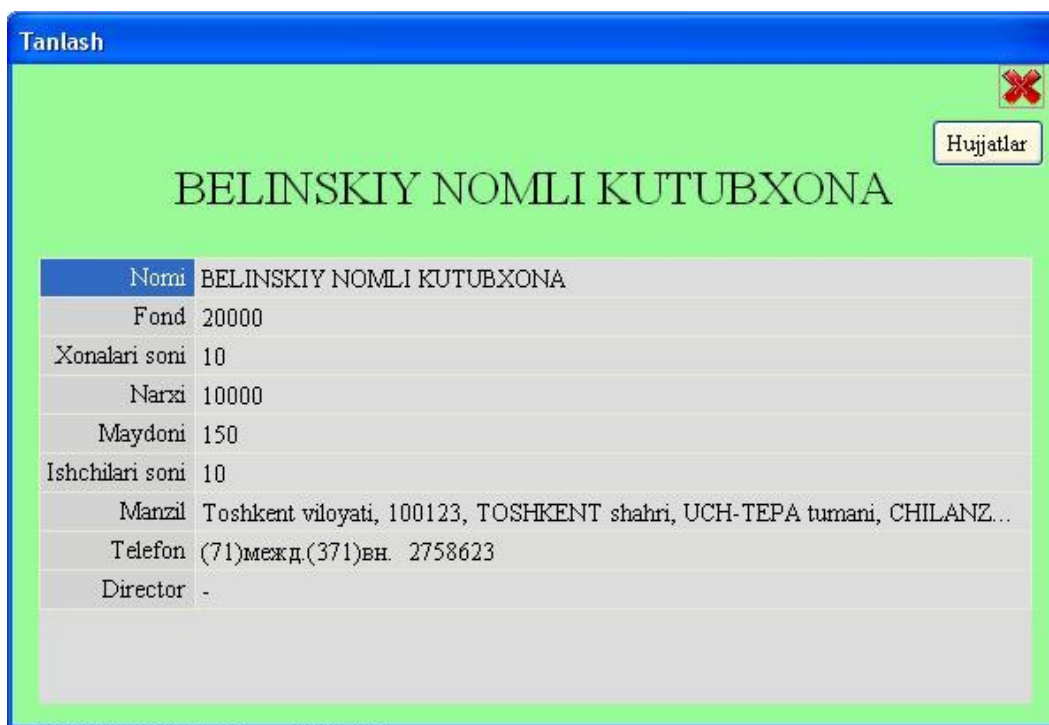
3.2.3-rasm.

Ikkinchi idirish tizimi Filtr deb nomlanadi. Filtr idirish tizimi ani bo'gan nomlar, iymatlar yozishni talab iladi. Bunda o'xshashlarni idirish mumkin emas. Filtrda idirish usuli chegara o'yib ham yoki o'masdan ham idirish mumkin. Misol uchun kutubhonaning katta-kichikligi yani maydoni ancha kattalika ega bo'yicha idiraylik. Kutubhonang maydoni ming kvadrat metrdan ikki ming kvadrat metr gacha bo'lsin. Kiritganimizdan so'ng ok knopkasi eziladi. Ushbu rasmda ko'rsatilgan (3.2.4-rasm).



3.2.4-rasm.

Ok knopkasi ezilgandan so'ng sizga kutubhona haqida ma'lumotlar rasmlarni ko'rishish mumkin. idirib topilgan ma'lumot ushbu ko'rinishda buladi (3.2.5-rasm).



The screenshot shows a window titled 'Tanlash' with a green background. At the top right, there is a red 'X' icon and a button labeled 'Hujjatlar'. The main content is a table with the following data:

Nomi	BELINSKIY NOMLI KUTUBXONA
Fond	20000
Xonalari soni	10
Narxi	10000
Maydoni	150
Ishchilari soni	10
Manzil	Toshkent viloyati, 100123, TOSHKENT shahri, UCH-TEPA tumani, CHILANZ...
Telefon	(71)межд.(371)ВН. 2758623
Director	-

3.2.5-rasm.

Agar siz kutubhona rasmlarini kurmochi bo'sangiz unda siz yuoridagi Hujjatlar knopkasin ezasiz. Rasmlar arxiv farmatda bo'ladi.

Dasturning asosiy oynasida **Administrator** bo'limi bo'lib u bo'lim bosharuvchi uchun mo'ljallangan bo'lib u albatda Login va Parol so'larini bilishi kerak bo'ladi (3.2.6 rasm). Bu bo'limda kutubhona haqidagi ma'lumotlarni kiritib ma'lumotlar bazasini to'ldirib boradi.



The screenshot shows a window titled 'Authentication' with a green background. It contains two input fields: 'Login' and 'Password'. To the right of the 'Password' field is an 'OK' button.

3.2.6-rasm.

Ma'lumotlar bazasini to'ldirish jarayoni. To'ldirish jarayoniga uydagi ma'lumotlar kerak bo'ladi. Nomi, Direktor, Fondi, Maydoni, Ummumiy narxi, Ishchilar soni, Xonalar soni, Manzili, Telefon va Rasmi kerak bo'ladi. Rasmi salash uchun avval rasmi arxiv farmatiga keltirib olamiz so'ng File knopkasi bosamiz va rasmi tanlaymiz va salash knopkasini ezamiz. Ma'lumot yozildi (3.7-rasm).

The image shows a screenshot of a software application window titled "Admin". The window has a blue title bar with a standard Windows icon on the left and a close button (X) on the right. The main content area has a light green background. On the left side, there is a vertical list of labels: "Nomi", "Director", "Fond", "Maydoni", "Umumiy narxi", "Ishchilar soni", "Xonalar soni", "Address", and "Telefon". Each label is in a grey rectangular box. To the right of each label is a large, empty green rectangular input field. At the bottom right of the form, there are two yellow buttons: "File" and "Saqlash". Below the input fields, there is a large, empty light blue rectangular area.

3.2.7-rasm.

Hulosa

Bu bob bitiruv malaka ishining asosiy maqsadlaridan bo'lgan qismi hisoblanib, unda ARM dizayni ko'rinishlarini topib beradigan dasturning ayni o'zi haqida gairiladi.

Dastur yaratilib, uning interfeysi haqida so'z yuritilib, bob so'ngida ARM dizaynini ko'rsatib beruvchi, uch o'lchovli grafik dasturning qo'llanilishi haqida batafsil yoritilib beriladi.

4 BOB. MEHNAT MUXOFAZASI VA TEXNIKA XAVFSIZLIGI

4.1. Shovin va uning inson tanasiga ta'siri

Shovin, ultratovushlar ajralib chiishga arab bir xil bo'ladilar: ular ammasi jismlarning tebranishdan tashkil topib, bizning eshitish organizmi tomonidan abul ilinadi. Ular bir-birlaridan faat tebranish chastotasi bilan va odamlar ularni ar xil abul ilishi bilan far iladilar. 20dan 20000 gts gacha tebranishlarni tovush deb ataladi va ularni biz tovushdek eshitamiz. Shunday bir ancha tovushlarni tartibsiz o'shilishi shovin deb ataladi. 20 gts dan past bo'lgan tebranishlarni infratovush deb ataladi. 20000 gts dan yuori bo'lgan tebranishlarni esa ultratovush deyiladi. Ultratovushlarning biz eshita olmaymiz, ularni faat ba'zi bir uy xayvonlarigina eshita oladi. atti ismlarning tebranishiga va shu tebranishlarni jismlarning o'zlari yoki bosha atti jismlar orali o'zatilishiga silkinish deyiladi.

Silkinishni biz chayalishdek abul ilamiz va ularni tebranish chastotasi 1 gts dan 100 gts gacha bo'ladi. Yuorida aytib o'tilganidek turli chastotadagi ar xil tovushlarning tartibsiz o'shilishib eshitilishi shovin deb ataladi.

Ritimlarga rioya ilingan olda muntazam ravishda kelib chiadigan oangrabo tovushlarga muzikali tovushlar deb ataladi. Muzika va shuningdek, shovin, bir vatning o'zida tovush chiaradigan ator oddiy yoeki sof tonlar, ya'ni tovush chiaradigan jismlarning mayin tebranishidan kelib chiadigan tovushlardan iborat. Shuning uchun ar anday ovoz aloida komponentlarga bo'linishi mumkin.

Muzika bizga estetik zav beradi, shovin esa ashimizni keltiradigan darajada ta'sir iladi. Tovush chiaradigan jismlarning tebranishdan osil bo'ladigan tovushlar yoki shovinli to'linlar avo bo'shliiga taralib, avoni go uyultiradi, go siyraklashtiradi va bu bilan boli bo'lgan avo bosimini o'zgartiradi. Bu bosim esa kishilarning tashi eshitish yo'llari orali o'tib, uloning eshitish yo'llari orali o'tib, uloning noora pardasini, undan keyin ichki ulodagi eshitish suyakchalari sitemasi orali ichki uloning abul iluvchi apparatini yoki chiano organlarini ko'p yoki oz tebrantirib, arakatga keltiradi.

Tovush kuchi attiligiga arab sub'ektiv baolanadi. Tovush tarkibi asosiy tonlarni kuzatib boruvchi o'shimcha tonlarning midori va sifati bilan xarakterlanadi. Tebranish chastotasi ovoz balandligini anilaydi va ovozning asosiy sub'ektiv xarakteristikalaridan biri isoblanadi. Tovush tebranishlarining absolyut uvvati yoki fizik birligi sifatidagi tovush kuchi va fiziologik sezgi sifatidagi uning attiligi o'rtasida to'ridan-to'ri proporsional bolanish yo'. Bu bolanish murakkab bo'lib, odamning eshitish apparatlari funktsiyalarining asosiy xususiyatlari bilan bolidir. Normal eshitishda tovush tebranishlarining 20 dan-20000 gts gacha chastotasi abul ilinadi, shunga am eng yuori chegara faat bolalar yoshiga xosdir.

Ular baloatga etgan sari eshitish organlari tomonidan abul ilinadigan tovushlarning chastotasi kamaya boradi va yosh o'tib olganida 15000 gts dan oshmaydi. Ana shu chegaralarda ar bir tovush uchun tovush uvvatining yoki tovush kuchining oxirgi ta'siri bor. uvvatning minimal oxirgi ta'siri

uning bilinar bilinmas sezgisini osil iladigan tovush kuchiga mos keladigan tovush kuchiga mos keladi, ya'ni tovush eshitilishi bo'saasida turadi. uvvatning maksimal oxirgi ta'sirni «ori bo'saasiga» mos keladi-tovush uvvati keyinchalik zo'rayganda tovushning kuchayishi eshitilmay, balki ikkala ulo am zirirab oriy boshlaydi.

Tovush uvvati, ya'ni tovush kuchi anchalik kuchli bo'lsa, tovush balandligini bir meyorda kuchaytirish uchun absolyut o'sish, shunchalik ko'p talab ilinadi. Ma'lum bo'lishicha, eshitish organi tomnidan, abul ilinadigan tovushning balandligi tovush tebranishning mutlo o'sishiga parallel ravishda kuchayibgina bormay, uning kuchayish logarfmiga taxminan proportsional am ekan. Shuning uchun am tovush kuchini o'lchash uchun logarifmli sistemasi birligidan foydalaniladi.

Tovush attilik darajasini anilash katta praktik aamiyatga egadir. Shovini tekshirish uchun maxsus asboblar ishlatiladi. Bu boblarining birida shovin attiligini eshitib ko'rib, uni zummerida osil etilgan va kuch bo'yicha regulyatsiya ilinadigan standart shovin attiligiga taoslash yo'li bilan o'lchanadi. Bosha asboblar bilan shovinni o'lchash uchun odamning eshitib ko'rishi shart emas. Bu ob'ektiv deb ataladigan shovin o'lchovchi asboblar to'ridan-to'ri detsibellar bilan shovin kuchining darajasini o'lchashga imkon beradi. Bundan tashari nioyatda murakkab tuzilgan tovush analizatorlari am mavjud bo'lib, ular mavjud, shovinning anday chastotalardan iboratligini va umumiy tovush uvvatining aysi chastota bo'laklariga to'ri kelishini juda am anilik bilan belgilab beradi.

atti shovin eshitish organlariga yomon ta'sir ilishi natijasida ishchilarning eshitish obiliyati pasayib ketadi. Bunda, avvalo eng kuchli darajada yuori tebranish chastotasiga ega bo'lgan tovushlarni abul ilish buziladi. Bu ko'pincha yuori chastotaga ega bo'lgan pichirlab gapirishni yaxshi eshitmaslik bilan ifodalanadi. Juda ko'p tebranishlarga ega bo'lgan tovushlar kar bo'lib olishda asosiy rol o'ynaydi. Past tovushlar yoki oz sonli tebranishlarga ega bo'lgan tovushlar, garchi ularning kuchi yoki tebranish amplitudasi katta bo'lganda am deyarli zararsiz xisoblanadi.

Bu ozonchalarda va ozonchalarda pnevmatik asboblar bilan ishlovchi amda bosha odamlardan ajralgan olda yakka ishlovchi bosha kasb egalarida eshitish obiliyatining tezda pasayib ketishi bilan ifodalanadi. Kasb orasida kart bo'lib olishiga tibbiy nutai nazaridan araganda. Uning asosida ichki uloning zararlanishi yotadi. Zaraning kuchi va uning chiano shkalasi bo'yicha taralish faat tovushning balandligiga boli bo'libgina olmasdan, shu bilan birga shovin ta'sirining bevosita intensivligi va uzoligiga am bolidir.

Professional kar bo'lib olishda avoning o'tkazuvchanligi asosiy rol o'ynaydi. Lekin ishlab chiarishda shovin ichki uloa faat avo orali o'tibgina olmasdan, balki bosh suyagi orali am o'tadi. uloning eshitishiga katta talab o'ymaydigan ko'pgina kasblarda yaxshi eshitmaslik ishlashga to'silik ilmaydi. Lekin ishda ulo yaxshi eshitishi lozim bo'lgan olda atti shovinda uloning oirlanib olishi ishlash obiliyatining pasayishiga olib keladi. Ko'pincha kasbda ishlay olmay olishiga am sababchi bo'ladi.

Professional shovin boshni aylantirib, miyada ori turgazadi va ulo shanib nerv sistemasiga am yomon ta'sir iladi. Aynisa fikrni to'plab, aliy ish bilan shuullanishga imkon bermaydi. Uzo vat mobaynida shovinning odamga sezilmas darajada ta'sir ilishi nerv sistemasini ishdan chiishiga olib keladi. Shovin ta'sirida turli organ va sistemalarning, masalan; xazm ilish, on aylanish va shunga o'xshashlarning normal faoliyati buziladi.

Shovinni kamaytirishga aratilgan chora-tadbirlar Yangi sanoat korxonalarini va sexlarini loyialashda shu sanoat korxonasi va sexlarida kelib chiishi mumkin bo'lgan shovin bosimi darajalarini anilash muim vazifa isoblanadi.

Ma'lumki shovin chiaruvchi mashina va mexanizmlar sanoat korxonasining biror sexida joylashganligini isobga olib, ana shu shovinni tevarak-atrofdagi ishlab-chiarish korxonalariga, aoli yashash joylariga shovin ta'sirini kamaytirishga aratilgan chora-tadbirlar korxonani loyixalash davrida isobga olinadi. Shovinni isoblash asosan, uyidagi vazifalarni o'z ichiga oladi:

1) Ma'lum nutada shovin chiarishi mumkin bo'lgan va shovin tavsiflari ani bo'lgan shovin manbaining shovin bosimi darajasini anilash.

2) Shovinniig kamaytirilishi lozim bo'lgan midori.

3) Shovinni ruxsat etiladigan midor darajasiga keltirish chora-tadbirlari. isoblash nutasi ochi maydonda yoki berk xona ichida joylashgan bo'lsa, bularning ar biri uchun isoblash formulalari ar xil bo'ladi. Hisoblash sakkizta oktava orali bo'yicha ayrim-ayrim o'tkaziladi.

Hisoblab topilgan shovin darajalari iymati yo'l o'yilishi mumkin bo'lgan iymatlar bilan solishtiriladi va kamaytirilishi zarur bo'lgan shovin darajasi midori anilanadi (dB):

$$\Delta L_x = L - L_{o'sh}$$

Agar shovin taratuvchi manba xona ichida joylashgan bo'lsa, shovin devor, pol, shift va bosha joylarga tegib ko'p marta aytishi natijasida xonadagi shovin xuddi shunday tasharidagi shovinga nisbatan 10-15 dB ortib ketishi mumkin.

Shuning uchun am bino ichida ishlayotgan mashina, bino tasharisida ishlayotganga nisbatan attiro tovush chiarayotgandek tuyuladi.

Shuning uchun am bino ichidagi shovin intensivligini anilaganda to'g'ri kelayotgan shovin intensivligi bilan birga aytgan tovush intensivligi I ni am e'tiborga olish kerak. Agar shovin taratuvchi manba xona ichida joylashgan bo'lsa, shovin devor, pol, shift va bosha joylarga tegib ko'p marta aytishi natijasida xonadagi

shovin xuddi shunday tasharidagi shovinga nisbatan 10-15 DB ortib ketishi mumkin. Shuning uchun am bino ichida ishlayotgan mashina, bino tasharisida ishlayotganga nisbatan attiro tovush chiarayotgandek tuyuladi.

Shuning uchun am bino ichidagi shovin intensivligini anilaganda to'g'ri kelayotgan shovin intensivligi bilan birga aytgan tovush intensivligi I ni am e'tiborga olish kerak.

$$I = I_T + I_K = (P\Phi/S) + (4P/B)$$

Bunda V - xona uchun o'zgarmas miqdor bo'lib $V=A (1 - \alpha_{O'rt})$ ga teng. A - tovush yutish ekvivalent maydoni, $A=\alpha_{O'rt} \cdot S_{yuza}$ yuza, bunda $\alpha_{O'rt}$ - S yuzaga ega bo'lgan tovush yutish o'rtacha koeffitsienti. Shovqin yutish koeffitsienti $\alpha=I_{IoT}/I_{Tysh}$ bunda: I_{IoT} -yutilgan tovushning intensivligi; I_{Tysh} - tushayotgan tovushning intensivligi, $\alpha \leq 1$.

Shovqin manbai yaqinida shovqin darajasi to'g'ridan-to'g'ri manbadan chiqayotgan tovush sifatida, undan uzoqlashganda esa, qaytayotgan tovushlar qo'shilgan qolda ifodalanadi. Sanoat korxonalarida I ning miqdori 0,3-0,4 atrofida bo'ladi. Shuning uchun agar $A=V$ deb qabul qilsak, xatolik uncha katta bo'lmaydi.

Ba'zi bir amallarni bajarib, dB bilan belgilanadigan shovqin qisobini quyidagicha ifodalash mumkin.

$$L = L_p + 10 \lg (\Phi/S + 4/B)$$

4.2. Yonqin xavfsizligi

Yonqin sanoat korxonalari, xalq xujaligining barcha tarmokdarida yuz berib, etkazadigan zarari jixatidan tabiiy ofatlarga tenglashishi mumkin bo'lgan xodisa qisoblanadi. Ular katta moddiy zarar keltirishi bilan birga oqir baxtsiz xodisalarga zaxarlanish, kuyish qamda kishilar xalokatiga sabab bo'lishi mumkin.

Yonqinga qarshi kurash ishlari davlat miqiyosida amalga oshiriladi. Yonqin xavfsizligini ta'minlash, uning rivojlanib, tarqalib ketmasligi chora-tadbirlarini oldindan ko'rish, unga qarshi samarali kurash olib borish yonqinni o'chirishda qo'llaniladigan birlamchi vositalardan tuqri foydalanishga qaratilgan.

Murakkab oksidlanish jarayonida yonuvchi moddalardagi bir moddaning ikkinchi moddaga aylanishi oqibatida katta miqdorda issiqlik va nurlanish ajralishi bilan kechadigan xolatga yonish deb tushuniladi. Yonqinga asosan uch omil: yonuvchi modda, yondiruvchi muxit, qizdirish jarayoni sabab bo'ladi.

Yonish jarayoni, asosan, ikki xil bo'lishi mumkin. Birinchidan, qattiq jismlar yonish jarayonida xavo muxitidan ajralgan xolda buladi. Bunday yonish xavo qarorati natijasida yonish zonasini kislorod bilan ta'minlaganligi uchun diffuziyali yonish deyiladi, uni yoqoch, ko'mir va boshqa moddalar yonganda kuzatish mumkin. Yonishning ikkinchi xili yonuvchi gazlar va suyuqliklarning parlari, yonuvchi moddalarning changlari xavo bilan aralashgan xolatdagi yonishi bo'lib, u kinetik yonish deb yuritiladi. Bunday yonish xajmli yonish jarayonida o'tadi.

Yonish tezligi modda tuyinganligiga, xarakatiga boqliq bo'ladi. Agar bunday yonish yopiq xajmlarda yoki idishlarda bo'lsa, portlash xodisasi ruy beradi.

Yonish quyidagi turlarga bo'linadi:

- yonuvchi aralashmaning bir laxzada yonib, o'chishi;
- qizdirish natijasida yonishning vujudga kelishi;
- uchqunlanish natijasida alangaga aylanish;
- organik moddalar ichida ruy beradigan ekzotermik reaksiyalar natijasida, yonuvchi aralashmaning tashqaridan qizishsiz o'z-o'zidan yonib ketishi;
- o'z-o'zidan alangalanish, o'z-o'zidan yonishning alanga bilan davom etishi;
- portlash-kimyoviy jarayonning bosim va quvvat xosil qilish bilan o'tishi.

Yonuvchi modda ma'lum xaroratda uzidan yonuvchi bug ajratishi natijasida alangalanishni ta'minlasa, bu xarorat alangalanish qarorati deb yuritiladi. Ba'zi bir organik moddalar (torf, qipiq, paxta, ko'mir maxsulotlari) o'z-o'zidan yonib ketish xususiyatiga ega. Chunki bular qovak asosga ega bo'lganligi va oksidlanishi mumkin bo'lgan yuza juda kattaligi tufayli, ochiq joylarda ma'lum miqdorda yio'ilib, ob-qavo o'zgarishi, kislorod ta'sirida qizib, yonib ketadi. Buning asosiy sababi, organik moddalar namlanganda uning ichki qismida mikroorganizmlar rivojlanadi va natijada issiqlik ajralib chiqadi. Bu xodisa organik moddalarning o'z-o'zidan qizish jarayoni deb ataladi.

Yonish jarayoni yonuvchi modda molekularining kislorod molekulari bilan birikish xodisasi qisoblanadi. Uni akademik N.N.Semyonovning zanjirli reaksiya nazariyasi asosida tushuniladi. Oksidlanish reaksiyasi natijasida odatda issiqlik ajralishi ma'lum sharoitda tezlashib ketishi mumkin. Mana shu tezlanish davri yonishga o'tgan davrga to'qri kelib, buni o'z-o'zidan alangalanish xodisasi deb yuritiladi.

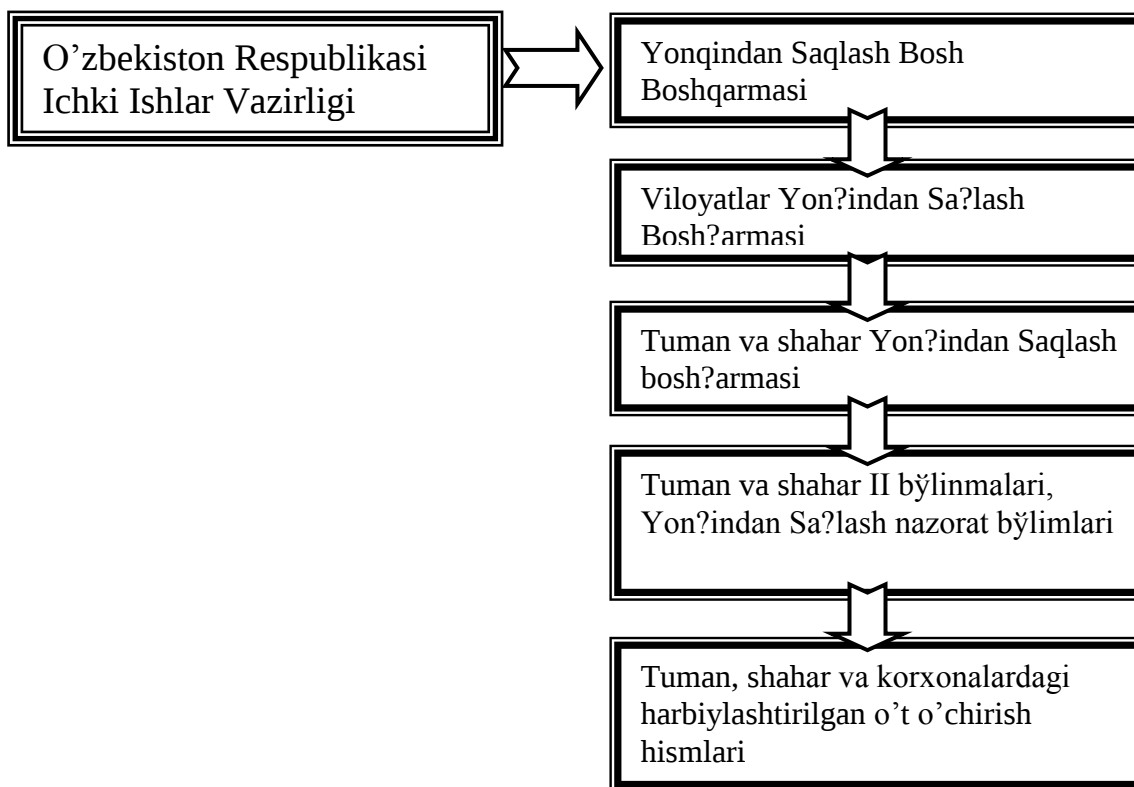
O'z-o'zidan alangalanish issiqlik ta'sirida yoki zanjirli reaksiya asosida yuz berishi mumkin. Issiqlik ta'sirida yonishda reaksiya natijasida ajralib chiqayotgan issiqlik tashqi muxitga tarqalayotgan issiqlikdan katta bo'lgan taqdirdagina vujudga keladi. Buni qo'yidagi misolda ko'rib chiqamiz.

Faraz qilaylik, idishda V xajmida yonuvchi gaz yoki buqlanib yonuvchi gaz xolatiga kelgan suyuqlik xavo bilan birga to'ldirilgan bo'lsin. Shu xonadagi qarorat va atmosfera bosimi bilan idishdagi aralashma o'rtasida xech qanday reaksiya bulmaydi. Ma'lumki, reaksiya jarayoni faqatgina qarorat ko'tarilishi bilan ruyobga chiqadi. Agar biz idish qaroratini asta-sekin ko'tara borsak, ya'ni idishni qizdirsak, undagi aralashma qaporati xam ko'tarilib, reaksiya tezligi xam ortib boradi va o'z navbatida ajralib chiqayottan issiqlik xam ko'tariladi.

Yonqindan qimoyalash tashkilotlari tizimi

O'zbekiston davlatchiligi tizimida yonqinga qarshi kurashishda asosiy ma'suliyat Ichki Ishlar Vazirligi qarashli bo'lgan Yonqindan saqlash bosh boshqarmasiga yuklatilgan. U shaqar va

qishloqlardagi xalq xo'jaligining bino va inshootlarida yonqin xavfsizligini ta'minlashda markaziy boshqaruv organi vazifasini o'taydi.



1-rasm. Respublika yonqindan saqlash xizmatining strukturaviy tarkibi

Bino va inshootlarni yonqin va portlash xavfi bo'yicha guruqlanishi

Bino va inshootlarni yonish va portlashda moyillik darajasini aniqlashdan maqsad ularda sodir bo'ladigan yonqin va portlashlar oqibatida yuzaga keluvchi buzilishlarni va odamlarga xavfli va daqshatli ta'sirini oldini olishdan iborat. Bino va inshootlarni yonish va portlashga moyilligi, ularning qanday ashyolardan qurilganligiga va ularda mavjud ishlab chiqarish jarayonida ishlatilgan yoki saqlanadigan xomashyolarning yonuvchanlik xususiyatlari bilan belgilanadi.

Texnologik loyiqalash me'yori va qurilish qoidalarini va me'yorlariga binoan sanoat korxonalari va omborlari yonish va portlash xavfi bo'yicha 5-ta toifalarga bo'linadi, jumladan A, V, V, G va D. Bularning A va V toifalari yonish va portlashga moyil. V va G toifalari bo'lsa faqat yonishga xavfli deb qisoblanadi. D toifa esa portlash va yonish xavfi mavjud emas.

Bino va inshootlarni bunday guruqlanishi, ularda ishlatiladigan yoki saqlanadigan engil yonuvchi gazsimon va suyuq moddalarning buqlari qavo bilan aralashganda, portlovchi gazli muxitni qosil qiluvchi agregat qolati va ularning alanganlanish qarorati (Ta)ga binoan amalga oshirilgan.

A-toifaga yonish va portlash xavfi mavjud bo'lgan, chaqnab yonish qarorati 280S dan past bo'lgan, yonuvchi gaz va engil alanganlanuvchi suyuqlik buqlari qavodagi kislorod bilan yoki suv bilan birikish natijasida portlashga moyil xavfli bosimi 5 kPa dan oshiq bo'lgan, gazsimon aralashmalar qosil bo'ladigan, korxonalar kiradi. Bu guruqga kiruvchi kimyo sanoatining atseton, oltingugurt,

karbon, efir, superfosfat va boshqa moddalarni ishlab chiqaruvchi korxonalarni misol qilib ko'rsatish mumkin. V-toifaga qam yonish va portlash xavfi bo'lgan, chaq nab yonish qarorati 280S dang yuqori bo'lgan, engil alanganuvchisuyuqlik buqlari, yonuvchi chang va gazlar qavodagi kislorod bilan, suv bilan qorishganda xavfli, portlovchi aralashma qosil qiluvchi miqdorda bo'lib, ular yonayotganda xonadagi xavfli bosim 5 kPa dan yuqori bo'ladi. Bunga ammiyak ishlab chiqarish sanoatini misol qilib ko'rsatishimiz mumkin.

V-toifaga faqat yonuvchi, ya'ni A va V toifalarga kirmaydigan sanoat korxonalari, jumladan chaq nab yonish qarorati 1200S dan yuqori bo'lgan, yonuvchi qattiq jismlarni ishlab chiqarish va qayta ishlov berish qamda qar xil yoqilqi moddalarni ishlatadigan sanoat korxonalari kiradi. Bunga misol qilib, yoqochni qayta ishlovchi mebelsozlik sanoati, qoqoz, karton, to'l qoqoz ishlab chiqaruvchi korxonalarni ko'rsatish mumkin.

G-toifaga yonmaydigan modda va ashyolarning qaynoq, cho'qlangan yoni eritilgan qolatda ishlatadigan korxonalar kiradi. Bunga metallurgiya sanoati korxonalari, issiqlik ishlab chiqarish markazlari va buqxonalar misol bo'la oladi.

D-toifaga yonmaydigan modda va ashyolarning sovuq qolatda ishlatadigan va saqlaydagn sanoat va qishloq xo'jaligi korxonalari kiradi. Masalan, toshni maydalash, keramika va tsement zavodlari shular jumlasidandir.

Bino va inshootlarni yonqin va portlash xavfi bo'yicha guruqlanishi, ulardagi barcha xonalarning yonish va portlash moillik toifasi aniqlangandan so'ng belgilanadi. Agar binoda A toifaga taaluqli bo'lsa, uning maydoni binodagi barcha xonalarning umumiy maydonidan 5%dan kam bo'lmasa yoki satqi 200m² dan ko'p bo'lsa, vu qolda bino A toifaga kiradi. Binoda qar xil toifaga taaluqli xonalar mavjud bo'lsa, A va V toifadagi xonalarning yiqindi maydoni, qolgan barcha xonalar umumiy maydonining 5%dan kam bo'lmasa yoki satqi 200 m² dane ziyod bo'lsa, bu bino V toifaga mansub bo'ladi.

Yonqinga qarshi umumiy talab va qoidalar

qurilish maydonlarida yonqin xavfsizligini ta'minlashda iqtisodiy samarador va texnik jiqatdan asoslangan ilqor ishlab chiqarish usullarini qamda yonqinni oldini olish va o'chirishning zamonaviy vositalarini qo'llash eng zarur omillar jumlasiga kiradi.

Yonqin xavfsizligi bo'yicha tadbirlar qurilishni tashkillashtirish va ishni bajarish loyiqalarida ishlab chiqiladi va ular yonqinni oldini olish va o'chirishni ta'minlashga qaratilgan bo'ladi. Bu tadbirlar "qurilishni tashkil qilish", "Yong'in xavfsizligi", "Sanoat korxonalarini bosh tarxi", "Yong'in xavfsizligi", qamda O'zbekiston Respublikasi IIV YoMBB tomonidan tasdiqlangan umumiy qurilish ishlarida "Yonqin xavfsizligi qoidalari" asosida ishlab chiqiladi.

Qurilish maydonining mutasadi raqbarlari (uchastka boshliqi, prorab, usta) yonqin xavfsizligi bo'yicha mas'ul qisoblanishadi va quyidagi talablarni bajarishlari shartdir:

-Qurilishni tashkillashtirish va ishni bajarish loyiqalarida aks ettirilgan yonqinga qarshi tadbirlarni va ularga xos yonqin xavfsizligi qoidalarini talablarini bilishlari qamda barcha ishchi va xizmatchilar tomonidan ularni to'liq va so'zsiz bajarilishini ta'minlashlari va nazorat qilishlari;

-me'yoriy qujjatlarda ko'rsatilgan o't o'chirish vositalari bilan ta'minlash va ularni qamma vaqt ishlatishga tayyor qolda saqlanishini ta'minlashlari;

-qurilayotgan bino va yordamchi ijtimoiy va omborxonalarni yonqin xavfsizligi qolatini doimiy nazorat qilib turishlari;

-elektr va issiqlik tarmoqlarini qarovsiz qolmasligini ta'minlash;

-qurilish maydonida o't o'chirish uchun zarur bo'lgan suv ta'minoti manbalarini aniqlamasdan va o't o'chiruvchi mashinlarni to'siqsiz qarakatlanishini ta'minlay oladigan yo'llarni qamda telefon-aloqa tarmoqlarini qurmasdan turib qurilish ishlarini boshlamaslikni ta'minlash;

-yonqin sodir bo'lganda zudlik bilan yonqinga qarshi qududiy xizmat bo'limiga xabar berish va birlamchi o't o'chirish vositalari yordamida yonqinni bartaraf etishni ta'minlash.

Muxandis va texnik xodimlarni yonqindan muqofaza qilish tadbirlarini bajarishga va shu yo'l bilan xalq mulkini asrab qolishga jalb qilish maqsadida, maqalliy Davlat yonqin nazorati tashkilotlari bilan doimiy aloqada bo'lib turadigan, xavfsizlik tadbirlarini o'z vaqtida bajarilishini nazorat qilishda asosiy omil qisoblanadigan Yonqin texnik komisiyaasini (YoTK) tuzish zarur bo'ladi. qurilish maydonida ishlayotgan qar bir ishchi va xizmatchi ish boshlashdan oldin Yonqin texnik minimumi (YoTM) bo'yicha xavfsizlik qoidalariga asosan maxsus o'qitilishi va tushuntirishlar olib borilishi shart. Bunday o'quv kursini o'tagan ishchi va xizmatchilar o'qish yakunida imtiqon topshirishlari lozim.

Tushuntirish jarayonida ishchi va xizmatchilarni yonqin xavfsizligi me'yorlari qamda qurilish maydoniga xos yonqinga qarshi tartib va qoidalar bilan batafsil tanishtirilishi shart.

Xulosa

- ARM dizayn ko'rinishlarini tanlab olish dasturi bugungi kunning eng yangi ahborot texnologiyasi bo'lib, u hech bir avtomatlashtirilgan kutubxonaladra shu kungacha hali bajarilmagan.
- ARM dizayn ko'rinishlarini tanlab olish dasturida ishlovchi ahborot resurs markazi va kutubxonalar rahbar hodimlari, zamonaviy grafik dasturdan habardor bo'lib, ular orqali dizayn tushunchasiga ega bo'ladilar.
- ARM dizayn ko'rinishlarini tanlab olish dasturi orqali kelajakda yangidan tashkil topadigan ARM va kutubhanalar dizayni uchun hech qanday tashqi dizaynerlarning yordamiga muhtoj bo'linmaydi.

Foydalanilgan adabiyotlar:

1. G.Lebedov, A.Kushnirenko, R.Svoren. «Informatika va hisoblash texnikasi asoslari». Toshkent, «O'ituvchi» 1991 yil.
2. T.A.Nurmuxammedov. «IBM PC va MS DOS bilan tanishuv» Toshkent. «Mehnat» -1994 yil.
3. A.Axmedov, N. Taylaov «Informatika» Toshkent. «Mehnat» -2001 yil
4. V.L.Broydo «Ofis texnikasi: bosharish va ish yuritish uchun» Toshkent. «Mehnat» -2001 yil
5. U. YU. Yuldashev. R. R. Boiev. “Informatika” Toshkent 2002 yil.
6. “Баркамол авлод йили” Давлат дастури тўғрисида: Ўзбекистон Республикаси Президентининг 2010 йил 27 январь Қарори //Маърифат. – 2010. - №8. – Б. 1-6.
- 7.Республика аҳолисини ахборот-кутубхона билан таъминлашни ташкил этиш тўғрисида: Ўзбекистон Республикаси Президентининг 2006 йил 20 июнь Қарори //Маърифат. – 2006. – 21 июнь.
http://www.elbib.ru/indtx.php3?env_page=journal/journal.ru.html_.
<http://www.iis.ru/el-lib/1999/199903/rusbridge/rusbridge.en.html>
6. www.Autodeck.com
7. www.Adobe.com
8. <http://www.3dnews.ru>
9. www.OSzone.net

ILOVA

Admin

Kod

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace kutubxona
{
    public partial class Admin : Form
    {
        public Admin()
        {
            InitializeComponent();

            private void Admin_Load(object sender, EventArgs e)
            {
                dataGridView1.ColumnCount = 2;
                dataGridView1.RowCount = 9;
                dataGridView1.Columns[0].AutoSizeMode =
DataGridviewAutoSizeMode.AllCells;
                dataGridView1.Columns[1].AutoSizeMode = DataGridviewAutoSizeMode.Fill;
                dataGridView1.Columns[0].DefaultCellStyle.BackColor =
dataGridView1.BackgroundColor;
                dataGridView1.Columns[0].ReadOnly = true;
                dataGridView1.GridColor = dataGridView1.BackgroundColor;
                dataGridView1[0, 0].Value = "Nomi";
                dataGridView1[0, 1].Value = "Director";
                dataGridView1[0, 2].Value = "Fond";
                dataGridView1[0, 3].Value = "Maydoni";
                dataGridView1[0, 4].Value = "Umumiy narxi";
                dataGridView1[0, 5].Value = "Ishchilar soni";
                dataGridView1[0, 6].Value = "Xonalar soni";
                dataGridView1[0, 7].Value = "Address";
                dataGridView1[0, 8].Value = "Telefon";
                dataGridView1.Columns[0].SortMode = DataGridViewColumnSortMode.Programmatic;
                dataGridView1.Columns[1].SortMode = DataGridViewColumnSortMode.Programmatic;
                dataGridView1.Font = new Font("Times New Roman", 10);
                dataGridView1.Columns[0].DefaultCellStyle.BackColor = Color.LightGray;
            }

            private void button1_Click(object sender, EventArgs e)
            {
                OpenFileDialog op = new OpenFileDialog();
                if (op.ShowDialog() == DialogResult.OK)
```

```

        {
            label1.Text = op.FileName;
        }
    }
    string nom
    {
        get
        {
            return DateTime.Now.ToString("yyMMddHHmmss");
        }
    }
    string papka(kutname k)
    {
        return k.CreateDirectory();
    }
    private void Save_Click(object sender, EventArgs e)
    {
        try
        {
            richTextBox1.Clear();
            kutname k = new kutname();
            k.nomi = "" + dataGridView1[1, 0].Value;
            k.director = "" + dataGridView1[1, 1].Value;
            k.kitobsoni = "" + dataGridView1[1, 2].Value;
            k.square = "" + dataGridView1[1, 3].Value;
            k.price = "" + dataGridView1[1, 4].Value;
            k.ishchisoni = "" + dataGridView1[1, 5].Value;
            k.xonasoni = "" + dataGridView1[1, 6].Value;
            k.address = "" + dataGridView1[1, 7].Value;
            k.tel = "" + dataGridView1[1, 8].Value;
            string x = k.Save();
            if (x.Equals("1"))
            {
                richTextBox1.Text = "Ma`lumot yozildi!";
                if (!label1.Text.Equals(""))
                {
                    FileX.CreateDirectory(FileX.GetCurrentDirectory + "\\data\\" + papka(k));
                    FileX.Save(label1.Text, nom, FileX.GetCurrentDirectory + "\\data\\" + papka(k)
+ "\\");
                }
            }
            else
            {
                richTextBox1.Text = "Ma`lumot yozilmadi!\nXatolik:\n" + x;
            }
        }
        catch (Exception exp) {
            richTextBox1.Text += "Ma`lumot yozilmadi!\nXatolik:\n" + exp.Message;
        }
    }
}

```

Autentication

Kod

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Data.Odbc;
namespace kutubxona
{
    public partial class authen : Form
    {
        public authen()
        {
            InitializeComponent();
        }
        public string reuest = "";
        private void ok_Click(object sender, EventArgs e)
        {
            OdbcCommand cmd = new OdbcCommand("select ism from admin where login=" +
log.Text + " and pwd=" + pas.Text + "");
            string ism = DBmanager.executeCommand(cmd, "ism")[0];
            if (!ism.Equals("Error"))
            {
                reuest = ism;
                Close();
            }
        }
    }
}
```

Hujjatlar

Kod

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace kutubxona
{
    public partial class Data : Form
    {
        public Data(string id)
```

```

    {
        InitializeComponent();
        FileX.CreateDirectory(FileX.GetCurrentDirectory + "\\data\\" + id + "\\");
        webBrowser1.Url = new Uri(FileX.GetCurrentDirectory + "\\data\\" + id + "\\");
    }

    private void button1_Click(object sender, EventArgs e)
    {
        Close();
    }
}

```

Filtr

Kod

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace kutubxona
{
    public partial class Filtr : Form
    {
        public Filtr()
        {
            InitializeComponent();
        }

        string PlusAnd(string shart1, string shart2)
        {
            if (!shart1.Equals(""))
            {
                if (!shart2.Equals(""))
                    shart1 = shart1 + " and " + shart2;
            }
            else
            {
                if (!shart2.Equals(""))
                {
                    shart1 = shart2;
                }
            }
            return shart1;
        }

        private void btnSearch_Click(object sender, EventArgs e)
        {
            string shart = "";
            if (!kitobsoni.Text.Equals(""))
                shart += " kitobsoni between " + kitobsoni.Text + " and " + kitobsoni2.Text;
        }
    }
}

```

```

if (!address.Text.Equals(""))
    shart = PlusAnd(shart, " address=" + address.Text + "");

if (!tel.Text.Equals(""))
    shart = PlusAnd(shart, " tel=" + tel.Text + "");

if (!price.Text.Equals(""))
    shart = PlusAnd(shart, " price between " + price.Text + " and " + price2.Text);

if (!square.Text.Equals(""))
    shart = PlusAnd(shart, " square between " + square.Text + " and " + square2.Text);

if (!shart.Equals(""))
{
    shart = " where " + shart;
    listBox1.DisplayMember = "nomi";
    listBox1.ValueMember = "id";
    listBox1.DataSource = DBmanager.GetData("select * from kutname " + shart);
}

private void listBox1_MouseDoubleClick(object sender, MouseEventArgs e)
{
    new Tanlash(Convert.ToInt32(listBox1.SelectedValue)).ShowDialog();
}

private void button3_Click(object sender, EventArgs e)
{
    Close();
}

private void kitobsoni_TextChanged(object sender, EventArgs e)
{
    kitobsoni2.Text = kitobsoni.Text;
}

private void price_TextChanged(object sender, EventArgs e)
{
    price2.Text = price.Text;
}

private void square_TextChanged(object sender, EventArgs e)
{
    square2.Text = square.Text;
}
}
}

```

Main (Asosiy forma)

Kod

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace kutubxona
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            new User().ShowDialog();
        }

        private void button3_Click(object sender, EventArgs e)
        {
            Close();
        }

        private void button2_Click(object sender, EventArgs e)
        {
            authen auth= new authen();
            auth.ShowDialog();
            if (!auth.reuest.Euals(""))
                new Admin().ShowDialog();
        }
    }
}
```

Tanlash

Kod

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Data.Odbc;
namespace kutubxona
```

```

{
    public partial class Tanlash : Form
    {
        public Tanlash(int ID)
        {
            InitializeComponent();
            id = ID;
            dataGridView1.Font = new Font("Times New Roman", 12);
            dataGridView1.Columns[0].DefaultCellStyle.BackColor =
dataGridView1.BackgroundColor;
            dataGridView1.Columns[1].DefaultCellStyle.BackColor =
dataGridView1.BackgroundColor;
            dataGridView1.Columns[0].AutoSizeMode =
DataGridViewAutoSizeColumnMode.AllCells;
            dataGridView1.Columns[0].DefaultCellStyle.Alignment =
DataGridViewContentAlignment.TopRight;
        }
        int id = -1;
        private void Tanlash_Load(object sender, EventArgs e)
        {
            kutname lib= kutname.Library("select * from kutname where id=" + id);
            dataGridView1.RowCount = 9;
            label1.Text = lib.nomi;
            dataGridView1[0, 0].Value = "Nomi";
            dataGridView1[1, 0].Value = lib.nomi;
            dataGridView1[0, 1].Value = "Fond";
            dataGridView1[1, 1].Value = lib.kitobsoni;
            dataGridView1[0, 2].Value = "Xonalari soni";
            dataGridView1[1, 2].Value = lib.xonasoni;
            dataGridView1[0, 3].Value = "Narxi";
            dataGridView1[1, 3].Value = lib.price;
            dataGridView1[0, 4].Value = "Maydoni";
            dataGridView1[1, 4].Value = lib.scuare;
            dataGridView1[0, 5].Value = "Ishchilari soni";
            dataGridView1[1, 5].Value = lib.ishchisoni;
            dataGridView1[0, 6].Value = "Manzil";
            dataGridView1[1, 6].Value = lib.address;
            dataGridView1[0, 7].Value = "Telefon";
            dataGridView1[1, 7].Value = lib.tel;
            dataGridView1[0, 8].Value = "Director";
            dataGridView1[1, 8].Value = lib.director;
            dataGridView1.Columns[0].SortMode = DataGridViewColumnSortMode.Programmatic;
            dataGridView1.Columns[1].SortMode = DataGridViewColumnSortMode.Programmatic;
            dataGridView1.Font = new Font("Times New Roman", 10);
            dataGridView1.Columns[0].DefaultCellStyle.BackColor = Color.LightGray;
        }
        private void button1_Click(object sender, EventArgs e)
        {
            Close();
        }
    }
}

```

```

private void Rasmlar_Click(object sender, EventArgs e)
{
    new Data("'" + id).ShowDialog();
}
}
}

```

User (idiruv tizimi)

Kod

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Data.Odbc;
namespace kutubxona
{
    public partial class User : Form
    {
        public User()
        {
            InitializeComponent();

private void button1_Click(object sender, EventArgs e)
{
    listBox1.DataSource = null;
    string x = "";
    if (!textBox1.Text.Trim().Equals(""))
    {
        x = "where " + comboBox1.SelectedValue + " like '" + textBox1.Text + "'";
        listBox1.DisplayMember = "nomi";
        listBox1.ValueMember = "id";
        listBox1.DataSource = kutname.Select("select * from kutname " + x);
    }
}

private void User_Load(object sender, EventArgs e)
{
    comboBox1.DisplayMember = "descrip";
    comboBox1.ValueMember = "nomi";
    comboBox1.DataSource = DBmanager.GetData("select * from nom");
}

private void button2_Click(object sender, EventArgs e)
{
    new Filtr().ShowDialog();
}
}

```

```

    }

    private void listBox1_MouseDoubleClick(object sender, MouseEventArgs e)
    {
        new Tanlash(Convert.ToInt32(listBox1.SelectedValue)).ShowDialog();
    }

    private void button3_Click(object sender, EventArgs e)
    {
        Close();
    }
}

```