

**O'ZBEKISTON RESPUBLİKASI ALOQA, AXBOROTLASHTIRISH
VA TELEKOMMUNİKATSIYA TEXNOLOGİYALARI DAVLAT
KO`MITASI**

**TOSHKENT AXBOROT TEXNOLOGİYALARI UNIVERSITETI
NUKUS FİLIALI**

Axborot texnologiyalari kafedrası

**Kompyuter injiniringi fakultetining
Informatika va axborot texnologiyalari yo`nalishining
talabasi Qutlmuratov Azamatning**

BITIRUV MALAKAVIY ISHI

***Mavzusi: “Fuqarolar nafaqa ta’minot bo’limining ma’lumotlar bazasini
SQL tili yordamida ishlab chiqish”***

Ilmiy rahbar:_____ t.f.n. Arzimbetov T.Z.

Kafedra mudiri:_____ t.f.n. Arzimbetov T.Z.

MUNDARIJA

Kirish.	3
1. TIZIM TAHLILI VA MASALANING QO'YILISHI	6
1.1. Avtomatlashtirilgan tizimlarda ma'lumotlarni tashkil etish . . .	6
1.2. Ish jarayonini avtomatlashtirishda masalaga yondoshuv.	9
1.3. Masalaning qo'yilishi.	12
2. TIZIMINI YARATISHDA FOYDALANILGAN DASTURIY VOSITALAR.	14
2.1. Ma'lumotlar bazasini yaratish bosqichlari.	14
2.2. Ma'lumotlar bazasini boshqarish tizimi arxitekturasini	26
2.3. MySQL dasturlash tili va uning imkoniyatlari.	29
2.4. PHP dasturlash tili haqida	31
3. "FUQAROLAR NAFAQA TA'MINOT BO'LIMINING MA'LUMOTLAR BAZASINI YARATISH.	37
3.1. Dasturiy vositalarga qo'yilgan talablar.	37
3.2. Fuqarolar nafaqa ta'minot bo'limi tiziminig strukturasi.	38
3.3. Foydalanuvchiga qo'llanma	39
XULOSA.	45
ADABIYOTLAR RO'YHATI.	46
ILOVA	

KIRISH

Mamlakatimiz inson manfaatlari, huquq va erkinliklari yuksak qadriyat bo'lgan ijtimoiy yo'naltirilgan bozor iqtisodiyotiga asoslangan huquqiy demokratik davlat va fuqarolik jamiyat barpo etish yo'lidan izchil rivojlanib bormoqda. Iqtisodiyotimizning turli soha va tarmoqlari o'rtasidagi mutanosiblikning kuchayishi hamda barqaror o'sish sur'atlarining ta'minlanishi natijasida aholi daromadlari, turmush darajasining sezilarli ravishda oshishi ertangi kunga bo'lgan ishonchimizning tobora mustahkamlanib borishiga zamin yaratmoqda.

Davlatimiz rahbarining ma'ruzasida jahonda e'tirof etilgan taraqqiyotning o'zbek modeli hamda Mamlakatimizda demokratik islohotlarni yanada chuqurlashtirish va fuqarolik jamiyatini rivojlantirish kontseptsiyasining ustuvor yo'nalishlarini izchil amalga oshirish natijasida, dunyo iqtisodiyotida yuz berayotgan inqiroz holatlariga qaramasdan, yildan yilga mamlakatimiz iqtisodiyotining yuqori barqaror o'sish sur'atlari va makroiqtisodiy mutanosibligi taminlangani qayd etildi.

Bu ma'ruzaning ilmiy-nazariy ahamiyati avvalo unda bugungi kunda jahon miqyosida ko'zga tashlanayotgan iqtisodiy rivojlanish tendentsiyalari, mavjud muammolar, global inqiroz holatini bartaraf etish bo'yicha amalda qo'llanilayotgan chora-tadbirlar chuqur tahlil etib berilganida ko'rinadi. Aynan ana shunday yondashuv asosida mamlakatimizda amalga oshirilayotgan islohotlar, ularning samaradorligini ta'minlayotgan omillarning mazmu-mohiyati ochib beriladi.

Axborot tizimlari va texnologiyalari yildan-yilga inson faoliyatining turli sohalarida keng qo'llanilmoqda. Ularni yaratish, ishga tushirish va qayotga keng tadbiq etishdan maqsad – jamiyat va inson butunhayot faoliyatini axborotlashtirish borasidagi muammolarini xal etishdir.

Jamiyatni axborotlashtirish deganda inson faoliyatining barcha ijtimoiy ahamiyatga ega bo'lgan sohalarda boyitilgan bilimlar, ishonchli axborotlar bilan to'liq va oz vaqtida foydalanishni ta'minlashga qaratilgan kompleks chora-tadbirlarni hamma joylarda tatbiq etish tushuniladi. Bundan shu narsa nazarda

tutilmoqdaki, zamonaviy axborot tizimlari va texnologiyalarini hamma joyga tatbiq etish qabul qilinajak qarorlar samarasini oshiradi. Bu faqat milliy iqtisod rivojlanishining iqtisodiy ko'rsatkichlari o'sishinigina emas, balki ayni paytda ishlab chiqarishni rivojlantirish, yangi ish joylarini tashkil etish, axolining turmush darajasini oshirish, atrof-muxitni muxofaza qilishga yo'naltirilgan fundamental va amaliy fanlarda sifatli ilmiy yutuqlarga erishishni ham ta'minlaydi.

Hozirgi kunda inson hayotida ma'lumotlar bazasida (MB) kerakli axborotlarni saqlash va undan oqilona foydalanish juda muhim rol uynaydi. Sababi jamiyat taraqqiyotining qaysi jabhasiga nazar solmaylik o'zimizga kerakli ma'lumotlarni olish uchun, albatta, ma'lumotlar bazasiga murojaat qilishga majbur bo'lamiz. Demak, ma'lumotlar bazasini tashkil qilish axborot almashuv texnologiyasining eng dolzarb xal qilinadigan muammolaridan biriga aylanib borayotgani davr taqozosidir.

Ushbu bitiruv malakaviy ishning maqsadi «Fuqarolar nafaqa ta'minot bo'limining ma'lumotlar bazasini SQL tili yordamida ishlab chiqish» bag'ishlangan. Korxonanung ish jarayonini avtomatlashtirishda ma'lumot bazasini boshqarish tizimini yaratish demakdur. Bundan asosiy maqsad nafaqa taminot bo'limning fuqarolar haqida barcha ma'lumotlarni markazlashtirib, ularni boshqarib turadigan avtomatlashtirilgan tizim yaratish dir.

Bu bitiruv malakaviy ish uch bo'lim, xulosa, foydalanilgan adabitotlar va ilova qismlaridan iborat.

Kirishda tanlangan mavzu dolzarbligi, korxonada avtomatlashtirilgan tizimini yaratish zarurligi, maqsadga muvofiqligi asoslanib qisqacha bayon berildi.

Birinchi bo'limda -soha tahlili va masalaning qo'yilishi keltirildi. Echilayotgan masalaga to'g'ri yondosha bilish yo'llari, usullari yoritildi. Va shularga asoslanib masalani qo'yilishi belgilandi.

Ikkinchi bo'limda- dasturlash muxitini tanlashga bag'ishlanib, unda axborot-ma'lumot bazasini yaratishda dasturlash tili tanlandi, apparat va dastur ta'minotiga qo'yilgan talablar keltirildi.

Uchinchi bo`limda - tizimni loyhalash jarayonlari yoritilib, ma'lumotlar bazasini boshqarish tizimiga tavsif va strukturasi berildi, foylanuvchiga va dasturchiga batafsil yo`riqnoma keltirildi.

Xulosa - Malakaviy bitiruv ishining asosiy xulosalari va natijalari keltirildi.

1. TIZIM TAHLILI VA MASALANING QO'YILISHI.

1.1. Avtomatlashtirilgan tizimlarda ma'lumotlarni tashkil etish

Ixtiyoriy korxona, tashkilotni o'zaro bog'langan elementlardan tashkil topgan informatsion tizim deb qarash mumkin. Bu tizimda ma'lum ko'rinishda tasvirlangan qayta uzatiluvchi va qayta ishlanuvchan axborot harakterlanadi deb qarash mumkin. Axborot tizim ma'lum axborot texnologiyalarga asoslanib ishlaydi. Bunda axborot texnologiyasi tushunchasiga, axborotlarni almashish printsiplari, ularni saqlash, uzatish, qayta ishlash usullari va qurilmalari majmuasi deb qarash mumkin. Dastlab Avtomatlashtirilgan tizimlar masalama masala yechishga asoslangan. Bunda oddiy, yengil, yechish algoritm mavjud bo'lgan masalalarni yechish tanlab olingan, chunki bunday masalalarni yechish tez o'z samarasini bergan. Unda asosan operativ bosqich masalalarini yechishni avtomatlashtirish xal qilgan, Bu masalalarga oylik maoshni hisoblash, ta'minlash, xizmatchilar masalalari va boshqalar kirgan .

Masalama masala echish usuli juda soda, ochiq va uning mazmuni quydagicha. har bir masala uchun o'zining ma'lumotlar bloki M1 va amaliy dasturlar paketi P1 tuzilgan. Bu quyilgan masalani yuqori samaradorlik bilan echish imkoniyatini bergan. Lekin murakkab avtomatlashtirilgan tizimlarni yaratishda masalama masala echish usulida muammolar paydo bo'ladi.

1. Ma'lumotlar ortiqchaligi va takrorlanishi paydo bo'ldi. Masalan, maoshni hisoblash masalasi va hodimlar masalasini bir xil ma'lumotlar bilan saqlaydi. (hodimlar haqida, ularni maoshi va mansabi va boshqalar). Murakkablik shundan iborat ediki, biror masalada ma'lumotlarni o'zgartirish, o'z-o'zidan boshqa masalalarda ham tuzatishni kiritish talab etadi [12].

2. Ma'lumotlar bilan amaliy dasturlar (Beysik, Paskal, Fortran kabi tillarda) orasida boqlanish borligi. Amaliy dasturlar ma'lumotlarni tavsiflashni ham, ular bilan ishlash algoritmlarini ham uz ichiga olgan (tanlash, qo'shish, almashtirish va boshqalar). Buning natijasida ma'lumotlarni tashkil qilishni ixtiyoriy uzgarishi, dasturni o'zgartirishni talab qilar edi. Bundan tashqari

ma'lumotlar bilan ishlaydigan bir xil algoritmlar barcha amaliy dasturlar tarkibida bo'lishlari talab etilardi.

Bu va boshqa kamchiliklar dasturchilarni avtomatlashtirilgan tizimlar yaratishda ma'lumotlar yangicha tashkil qilishni talab kilardi. Buning natijasida avtomatlashtirish tizimda ma'lumotlarni tashkil qilishiga standart talablar ko'yishni taqazo etardi.

Ularni asosiylari quydagilar:

- Ma'lumotlarni integratsiyasi, bunda barcha ma'lumotlar markazlashgan holda saqlanadi va yig'iladi, bunda predmet sohani dinamik modeli yaratiladi.
- Ma'lumotlarni amaliy dasturlardan maksimal mumkin bo'lgan mustaqilligini (bog'lik bo'lmasligini) ta'minlash, ya'ni predmet sohani logik modelini EHM xotirasida fizik mustaqilligini ta'minlash.

Bu talablarni bajarish barcha masalalr uchun umumiy bo'lgan ma'lumotlar blokini (ma'lumotlar bazasi - MB) yaratishga olib keldi va fizik bosqichdagi ma'lumotlar bilan manipulyatsiya kiluvchi bita boshkaruvchi dastur – ma'lumotlar bazasini boshkarish tizimini yaratishga olib keldi .

Masalama masala echish usuli amaliy dasturlardan, fizik bosqichda ma'lumotlar bilan ishlash (manipulyatsiya) qilish vlsitalari ma'lumotlar bazasini boshkarish tizimlari (MBBT) chikarildi MBBT gina ma'lumotlar mustaqilligi (amaliy dasturlar boglik emasligini) ta'minlaydi, amaliy dasturlar har biri alohida masalani mantiqiy ishlashini boshqaradi.

Fizik tashkil qilishning uzgarishi MBBT tomonidan qabul qilinadi va amaliy dasturga ta'sir qilmaydi. Amaliy dasturni mantiqiy ishlashi, fizik ma'lumotlarni murojaat mexanizatsiyali va qayta tashkil qilishni talab etmaydi.

Shunday qilib MBBT kiritish ma'lumotlarni mantiqiy tizimini (ya'ni foydalanuvchi nuktai nazardan ma'lumotlarni tashkil qilinishi haqidagi tasavvurni) EHM xotirasidagi ma'lumotlarni fizik strukturasidan (tuzilmasi) ajratadi [4].

Ma'lumotlar bazasi tushunchasi birinchi marta 1962 yilda paydo bo'ldi. Ma'lumotlar bazasi avtomatlashtirilgan axborot tizimlarini eng asosiy elementlari

hisoblanadi. Ma'lumotlar bazasi quydagi hossalalarga ega bo'lgan ma'lumotlar majmuidir:

- umumiy masalalar echishga muljallangan ma'lumotlarni integratsiyasi;
- ma'lumotlarni strukturalashgani;
- ma'lumotlarni o'zaro bog'liqligi;
- ma'lumotlar tavsifini amaliy dasturlarda mustaqilligi, chunki ma'lumotlar va ularni tavsifi MB bilan birga saklanadi.

MBBT – ma'lumotlar paketi bo'lib, u quydagi imkoniyatlarini beradi:

- foydalanuvchilarni ma'lumot tavsiflash va ishlash (manipulyatsiya) til vositalari bilan ta'minlash;

- ma'lumotlarni mantiqiy modelini himoya bilan ta'minlash;

- mantiqiy ma'lumotlarni yaratish va manipulyatsiyalash amallarini (tanlash, qo'shish, tiklash, olib tashlash va boshqalar) va bu amallarni fizik ma'lumotlar ustida bajarish.

- ma'lumotlarni himoyasi va bir butunligini ta'minlash, chunki kollektiv rejimda ko'p foydalanuvchilar ishlaganda ular tomonidan umumiy fizik ma'lumotlarga ishlatilishi mumkin. Bu foydalanuvchilar tomonidan noo'rin (nekorrektno'y) ma'lumotlarni himoyalash, kurilmalarni nosoz ishlashi natijasida bo'ladigan ma'lumotlarni o'chib ketishi (yemirilishi)dan himoya qilishni ta'minlash. MBBT ni qurishni ikkita varianti mavjud.

Birinchi variantda MBBT da ma'lumotlarni tavsiflash va manipulyatsiya qilish vositalari amaliy dastur yozilayotgan universal til tarkibiga kiritiladi.

Bu holda MBBT ma'lumotlarga maxsus so'rovlar tilining interpretori (yoki kompilyatori)dan iborat bo'lib, uning yordamida ma'lumotlar bazasini boshqarish muloqot yoki dastur rejimida komandalar yordamida boshqariladi.

Shunday qilib ma'lumotlar bazasini avtomatlashtirilgan axborot tizimlarda qo'llanishni quydagi afzalliklari bor:

1. Axborot resurslarini markazlashgan boshqarish, barcha ilovlarni ma'lumotlar bilan sinxron ishlashini ta'minlash;

2. Ma'lumotlar integrallashganligi uchun ortiqchalikni nazorat qilish muammosini yo'qligi;
3. Ma'lumotlarni takrorlashninishini yo'qligi uchun ular bir marta kiritiladi va ko'p marta ishlatiladi;
4. Ma'lumotlarni tashkil qilish vositalari unifikatsiyalashganligi va amaliy dasturlarni ma'lumot tashkil qilishga bog'liqmasligi.
5. Amaliyotda ma'lumotlar bazasiga asoslanib ishlaydigan avtomatlashtirilgan axborot tizimlari har xil sohada keng tatbiq qilinmoqda.

1.2. Ish jarayonini avtomatlashtirishda masalaga yondoshuv

O'zbekiston davlati va xukumatining raxbari axoli ijtimoiy himoyasi masalasini doimo diqqat markazida turadi.

Shunga muvofiq 1996 yil 1 apreldan boshlab O'zbekiston Respublikasining hududida mehnat pensiyasining shu mudaddatdan boshlab barcha turdagi pensiyalar 1,4 barobar oshiriladi.

Pensiya ta'minotiga bo'lgan huquq-insonning muhim huquqlaridan biridir.

O'zbekiston Respublikasida 1992 yil 8 dekabrda qabul qilingan Konstitutsiya bilan huquqiy mustahkamlab qo'yilgan.

O'zbekiston Respublikasi Konstitutsiyasining 39-moddasida shunday deyilgan «Har kim qariganda, mehnat layoqatini yo'qotganda, shuningdek boquvchisidan mahrum bo'lganda va qonunda nazarda tutilgan boshqa hollarda ijmoiy ta'minot olish huquqiga ega».

Fuqarolarning bu huquqi davlat tomonidan kafolatlangan va 1993 yil 3 sentyabrda maxsus qabul qilingan «Fuqarolarning davlat pensiya ta'minoti to'g'risida»gi O'zbekiston Respublikasining qonuni bilan ta'minlab qo'yilgan.

«Har kim qariganda, mehnat layoqatini yo'qotganda, shuningdek boquvchisidan mahrum bo'lganda va qonunda nazarda tutilgan boshqa hollarda ijtimoiy ta'minot olish huquqiga ega» O'zbekiston Respublikasi Konstitutsiyasining 39-moddasi.

O'zbekiston Respublikasida iqtisodiy isloxot o'tkazilmoqda, Baxolarning to'xtovsiz o'sishi va shu bilan bog'liq holda axoli manfaatlari va turmush darajasini to'xtovsiz himoya qilish zarurligi sababli ish haqini, talaba va o'quvchilar stipendiyasi, mehnatga layoqatsiz fuqarolarning pensiyalarini oshirishga doir davlat tadbirlari birinchi o'ringa chiqadi. Aytilgan tadbirlar O'zbekiston Respublikasi Prezidentining Farmonlari, Vazirlar Maxkamasining qarorlari bilan amalga oshiriladi.

Davlat pensiyalarining turlari

Ushbu qonun bilan davlat pensiyalarining quyidagi turlari belgilanadi:

- yoshga doir pensiya;
- nogironlik pensiyasi;
- boquvchisini yo'qotganlik pensiyasi.

Yoshga doir pensiya shunday pensiyaki, u fuqaroning ma'lum yoshga etganda, tegishli ish stajga ega bo'lganida tayinlanadi.

Shunday qilib, yosh va staj, ya'ni ikkita shart bu tur pensiya ta'minotiga bo'lgan huquqini aniqlaydi.

Boshqa holatlar muhim ahamiyatga ega emas.

Yoshga doir pensiya umumiy asoslarda – erkaklar 60 yoshga etganlarida va kamida 25 yil ish stajiga ega bo'lganlarida, ayollar 55 yoshga etganlarida va kamida 20 yillik ish stajiga ega bo'lganlarida tayinlanadi.

Bu eng kam pensiya yoshi bo'lib, dastlab 1928 yilda belgilangan va o'sha paytdan beri o'zgarmay kelmoqda.

Nogironlik pensiya mehnat qilish qobiliyati yo'qotilishi munosabati bilan belgilanadi, uning miqdori mehnat qobiliyatini yo'qotish darajasiga bog'liq.

Nogironlik pensiyani nogironlik qachon boshlanganligidan: ish davridami, ishga kirgunga qadarmi, uni tugatganidan keyinmi qat'i nazar tayinlanadi.

Agar nogironlik mehnatda mayiblanish yoki kasb kasalligi sababli boshlangan bo'lsa, u holda pensiya ishining davomiyligi sababli boshlangan bo'lsa, u holda pensiya ishining davomiyligidan qat'iy nazar tayinlanadi. Nogironlik

insonning mehnat faoliyati bilan bog'liq bo'lmagan hollarda pensiya tayinlash uchun ish staji talab qilinadi, bundan ishining davomiyligi yoshga bog'liq bo'ladi.

Fuqarolarning pensiya ta'minot bo'limlarining topshiriqnomalariga asosan pensiyalarni bevosita to'lashni Aloqa vazirligi korxonalari, jamg'arma bank muassasalari amalga oshiradilar.

Aloqa korxonalari tomonidan pensiyalarni to'lash bilan bog'liq barcha masalalar O'zbekiston Respublikasi Ijtimoiy ta'minot va Aloqa vazirliklari tomonidan tasdiqlangan maxsus qo'llanma tomonidan boshqarib turiladi. Bu qo'llanma Moliya vazirligi, O'zbekiston Respublikasi Vazirlar Maxkamasi qoshidagi Ijtimoiy sug'urta jamg'armasi va O'zbekiston Respublikasi Markaziy Bank boshqarmasi bilan kelishilgan holda amalga oshiriladi.

Mazkur qo'llanma bilan Aloqa vazirligi korxonalari («O'zbekiston pochta» konserni pochta aloqasining davlat korxonalari, aloqa bog'lamalari, pochta tarmoqlari, aloqa bo'limlari), Ijtimoiy ta'minot vazirligi idoralari (hududiy boshqarma, tuman, shahar ijtimoiy ta'minot bo'limlari) va Vazirlar Maxkamasi qoshidagi ijtimoiy sug'urta jamg'armasining bo'limlari o'rtasidagi pensiya va nafaqani eltib berish va to'lash, shuningdek mablag' bilan ta'minlashga doir o'zaro munosabatlar boshqarib turiladi va pochta jo'natmalari orqali amalga oshiriladi.

To'lov hujjatlari, topshiriqnoma ish qog'ozlari markazlashgan tartibda Ijtimoiy ta'minot vazirligi tomonidan tayyorlanadi.

Pensiya va nafaqalarni to'lash aloqa bog'lamasi tomonidan ijtimoiy ta'minot idoralari bilan hamkorlikda tuzilgan jadval bo'yicha amalga oshiriladi.

Pensiya axborotlarini avtomatlashtirilgan shartlarda pensiya va nafaqa to'lashni tashkil etish, to'langan pullarni nazorat qilish va hisoblash Respublika Ijtimoiy ta'minot, Moliya, Aloqa vazirliklari va O'zbekiston Respublikasi Vazirlar Maxkamasi xuzuridagi ijtimoiy sug'urta jamg'armasi ko'rsatmalari bilan aniqlanadi.

1. 3. Masalaning qo'yilishi

Mazkur bitiruv ishi Fuqarolarning pensiya ta'minot bo'limi ish jarayonini avtomatlashtirishga bag'ishlangan.

Ma'lumotlar bazasida Fuqarolarning pensiya ta'minot idorasining ish faoliyati haqidagi axborotlar saqlanadi.

Ishdan maqsad Fuqarolarning pensiya ta'minot bo'limi ish jarayonini avtomatlashtirishga bag'ishlanib, maqsadga erishish uchun quydagi masalalarni echish talab etiladi:

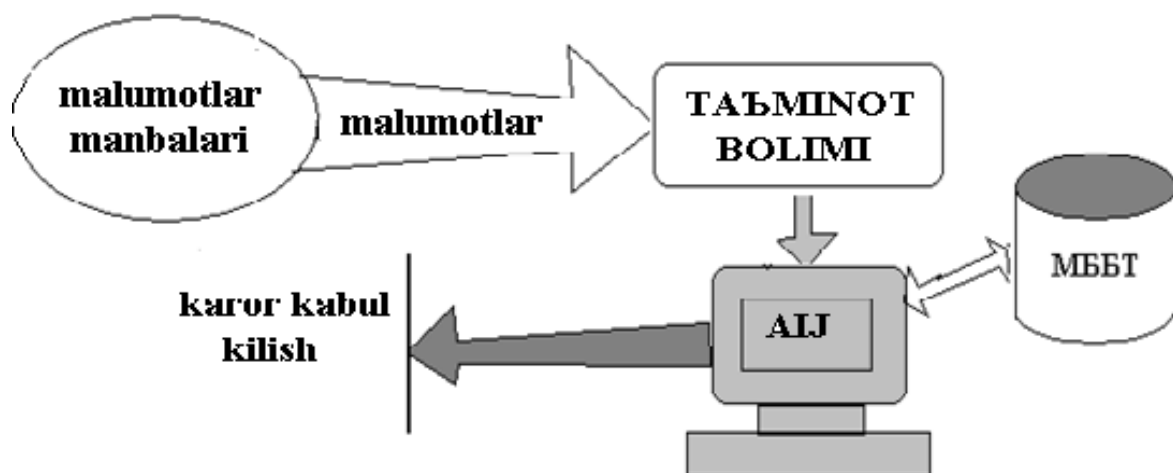
- eng zarur ma'lumotlarni onson qidirish uchun optimal ma'lumotlar bazasini loyhalash;
- avtomatlashtirilgan tizimni loyhalashda axborotlarni ishonchli va xavfsiz tashkil qilish;
- har oylik ro'yhat va topshiriqnomalarni tuzish va aloqa bog'lamalarga yangi to'lov hujjatlarini jo'natish;
- fuqarolarning pensiya ta'minot idoralari, ijtimoiy sug'urta idoralari, aloqa bog'lamalari, bank va boshqa mutassadi idoralar o'rtasidagi hisob-kitoblar hisobotlarini tayyorlash;
- yaratilgan dastur vositalari soda va foydalanishga qulay do'stona interfeysga ega bo'lishi lozim.

Loyhalanayotgan ma'lumotlar bazasi jadvallariga quydagi ma'lumotlar kiritiladi: pensiya va nafaqa oluvchining shaxsiy hisob varaqasidagi ma'lumotlar: shaxsiy varaqasi raqami, aloqa bo'limining va ijtimoiy ta'minot bo'limining nomi, shuningdek pensiya oluvchining ismi, nasabi, otasining ismi va uning manzili, pensiya yoki nafaqaning to'lov sanasi, uning turi.

Yaratilgan ma'lumotlar bazasi asosida zarur axborotlarni so'rov va tez qidiruv jarayonini tashkil etish, kerak bo'lsa bosmaga chiqarish, hamda optimal qaror qabul qilish imkoniyatlarini beradi.

Keltirilgan avtomatlashtirilgan ish joyi sxemasining ko'rinishi uncha murakkab emas ma'lumotlar manbalari ta'minot bo'limiga berilib, kompyuterga

kiritilib, qayta ishlanish natijasida kerakli qaror qabul qilinadi va ma'lumotlar bazasida saqlanib turiladi.



1.4.1.-rasm. Ta'minot bo'limi hodimining avtomatlashtirilgan ish joyi sxemasi.

2. TIZIMINI YARATISHDA FOYDALANILGAN DASTURIY VOSITALAR

2.1.Ma'lumotlar bazasini yaratish bosqichlari.

Ma'lumotlar bazasini yaratish predmet sohasini o'rganishdan boshlanadi. Bu paytda to'planadigan va foydalanuvchiga taqdim etiladigan axborotning quyidagi xususiyatlariga alohida e'tibor beriladi:

1. **To'laqlonlik.** Axborot to'laqlonligi ob'ekt faoliyatining u yoki bu tomonlarining miqdoriy va sifat parametrlarini aniq belgilash hamda mos qarorlarni ishlab chiqarishda ifodalaniladi. Axborotning noto'laqlonligi qarorlar qabul qilishda xatolarga olib kelishi mumkin.

2. **Ishonchlilik** qabul qilinadigan qarorlar samaradorligi saqlanadigan etib kelgan va natijaviy axborotlarda muayyan darajada buzilishlarga yo'l qo'yadi.

3. Axborotni **qabul qilishning bemalolliigi** vaqt birligida ma'lumotlarni qabul qilish tezligi bilan belgiladi. Shu bois ham ma'lumotlar ko'proq jadval shaklida beriladi, u nafaqat axborot mazmunini ochib beradi, balki yengil qabul qilinadi ham.

4. Ma'lumotlarning **dolzarbliigi** muayyan vaqt mobaynida aniq vazifani amalga oshirish uchun yaroqligini ifodalaydi. Shu bois ham dolzarblik, hozirjavoblik va tezkorlik axborotga xos xususiyatlardir.

5. **Kechikmaslik** axborotning qulay yoki belgilangan vaqtda kelib tushishini anglatadi. Bu talabni buzish axborotni qadrsizlantiradi.

6. **Aniqlik** uning to'g'riligini anglatadi. Axborotning aniqligi uning barcha iste'molchilar tomonidan bir xil qabul qilinishini ta'minlaydi.

7. **Tezkorlik** vaqt o'tgach axborot eskirishi va dolzarbligini yo'qo-tishini aks ettiradi.

Axborotning o'z vaqtida qabul qilinmasligi qaror qabul qilishni kechiktiradi, oqibatda qabul qilinayotgan qarorlar o'zgaruvchan sharoitda talabga javob bermaydi. Axborot qanchalik tezkor bo'lsa, u shunchalik qimmatli bo'ladi.

Axborotning qimmati aniqlik darajasi oshgani sayin yoki xabar qilinayotgan va aniq natijalar o'rtasidagi farq kamayganda tez ko'tariladi. To'liqroq va ishonchli axborot to'g'ri qarorni qabul qilishni ta'minlaydi.

Ma'lumotlar bazasiga kiritiladigan ma'lumotlarni aniqlangandan keyin bu ma'lumotlar bazasini tashkil etishda foydalaniladigan ma'lumotlar bazasini boshqarish tizimini (MBBT) tanlash lozim

MBBT o'z tasniflanishining muhim belgilaridan biri bo'lgan ma'lumotlar modeli turlaridan birini (tarmoqli, ierarxik yoki relyatsion) ta'minlaydi. MBBT ma'lumotlar bazalarining ko'p maqsadli tavsifini, ma'lumotlarni himoyalash va qayta tiklashni amalga oshiradi. Rivojlangan muloqot vositalari va yuqori darajali talablar tilining mavjudligi MBBTni oxirgi foydalanuvchi uchun oson vositaga aylantiradi.

Axborot tizimlari murakkab ma'lumotlar strukturasini barpo etishni talab qiladi. Murakkab strukturlashgan ma'lumotlarni boshqarish ma'suliyatini o'ziga oluvchi axborot tizimlarini umumiy qismini ajratishga va umumlashtirishga intilishlar "Ma'lumotlar bazalarini boshqarish tizimlarini"- (MBBT) yaratishga birinchi asosli sabab bo'ladi.

To'liq variantda MBBT quyidagi komponentlardan tuzilishi mumkin:

- Klaviatura orqali ma'lumotlarni to'g'ridan-to'g'ri boshqarish imkonini beruvchi foydalanuvchining muhiti;
- Interpretator sifatida ish yurituvchi, ma'lumotlarga ishlov berish amaliy tizimini dasturlashning algoritmik tili. Interpretator dasturlarni tez tuzish va maromiga etkazish imkonini beradi;
- Mustaqil EXE-fayl shaklidagi tayyor tijorat maxsulotiga tugallangan dastur ko'rinishini beruvchi kompilyator;
- Ko'p mehnat talab qiluvchi amallarni tez dasturlovchi utilit-dasturlar (hisobotlar, shakllar, jadvallar, ekranlar, menyu va boshqa ilovalar generatorlari).

Xususan MBBT – bu foydalanuvchining aslahaviy qobig'i hisoblanadi. MBBT tarkibida dasturlash tilining mavjudligi aniq masalalarni va aniq foydalanuvchiga mo'ljallangan ma'lumotlarga ishlov berishning murakkab tizimlarini yaratish imkonini beradi.

Ma'lumotlar bazalarini loyhashtirishda quyidagi tavsiflarini solishtirish va tahlil qilishga asoslangan MBBTni asoslab tanlab olish muhim vazifa hisoblanadi:

- dasturiy texnik bazasi (ShK turi va modeli, hisoblash vositalari konfiguratsiyasiga qo'yiladigan talablar, OT versiyasi);
- ma'lumotlar bazalarining turlari (amaliy, predmet, lokal, integrellashgan, taqsimlangan);
- foydalanuvchilarning malakalari (MBBT bilan ishlash uchun maxsus tayyorgarlikka ega bo'lmagan foydalanuvchi, mutaxassisligi dasturchi bo'lmagan darajadagi MB bilan ishlashga tayyorgarligi bo'lgan predmet sohasining mutaxassisi – foydalanuvchi, amaliy dasturchi, MB larning administratori);
- ma'lumotlar bazalari bilan foydalanuvchilarning muloqot qilish vositalari (dasturlash tillarini o'z ichiga oluvchi ma'lumotlar ustida ish olib borish va tasvirlash tili);
- ma'lumotlarni qayta ishlash rejimi (paketli, interaktiv, tarmoqli);
- ma'lumotlarni mantiqiy va fizik mustaqilligi;
- ma'lumotlar bazalari axborot strukturalarining asosiy hossalari (mantiqiy strukturalashtirish-MBBT vositalari orqali amal qiladigan va tashkiliy tuzilmasini o'zgartirmasdan uni modifikatsiya qilish imkoniyati, ma'lumotlar turini kengaytirgan holda qarshiliksiz ishlov berishda);
- xavfsizlik darajasini ta'minlash va ma'lumotlarning to'laqligi;
- xizmat ko'rsatishning standart vositalari mavjudligi (ma'lumotlar bazalarini kuzatishni doimiy dasturiy modullarini va ma'lumotlar lug'atini, ma'lumotlar bazalarini yengillashtiradigan (yukini tushiradigan), qayta tashkil etadigan va qayta strukturalashtiradigan, tiklaydigan jurnalni yuritish, kiritish va hisobotlar generatorlari va h.k.);
- ishlatish tavsiflari (loyhachilar haqida, sarmoya egalari haqida, moddiy-texnik ta'minotga bo'lgan talablar, tarqatish shakli).

Tanlab olingan MBBT bir qator talablarni qondira olishi kerak. Bularga predmet sohasi turli funksiyalarining samarali bajarilishi; xotira resurslaridan

samarali foydalanish uchun saqlanayotgan ma'lumotlar hajmini minimallashtirish; moxiyatli axborot qarorlarini qabul qilishga imkon yaratish; xavfsizlikni ta'minlash jarayonini boshqarish; hodimlarga nisbatan ma'lumotlar bazalarini ishlatish bilan bog'liq yuqori talablarning yo'qligi, EHMni ishlatish muomalasini soddalashtirish.

MBBTning asosiy vositalari quyidagilar:

- ma'lumotlar bazalari tuzilmalariga topshiriq berish (tasvirlash) vositalari;
- ma'lumotlarni kiritish, ko'rish va muloqotlar rejimida ishlashga mo'ljallangan ekran shakllarini loyhalash vositalari;
- berilgan sharoitlarda ma'lumotlarni tanlash uchun talablar yaratish, shuningdek, ularni ishlash bo'yicha operatsiyalar bajarish vositalari;
- foydalanuvchiga qulay ko'rinishda ishlov natijalarini bosmaga chiqarish uchun ma'lumotlar bazasidan hisobot yaratish vositalari;
- til vositalari – makroslar, qurilgan algoritmik til (Dbase, Visual Basic, Microsoft Access yoki boshalar), talablar tili (QBE- Query Ehample, SQL) va h.k. Ular ma'lumotlarni ishlashning nostandart algoritmlarini, shuningdek foydalanuvchi topshiriqlaridagi voqealarni ishlash protseduralarini bajarish uchun qo'llaniladi.

Ushbu malakaviy bitiruv ishini ishlab chiqishda, avvalam bor mavjud bo'lgan ma'lumotlar bazasini yaratish usullarini ko'rib chiqamiz va kerakli amallarni belgilab olamiz.

Ma'lumotlar modeli – o'zaro bog'langan ma'lumotlar tuzilishi va ular ustida bajariladigan operatsiyalar majmuidir. Model shakli va unda foydalaniladigan ma'lumotlar tuzilishining turi MBBT (ma'lumotlar bazalarini boshqarish tizimi) yordamchi model yoki ma'lumotlarni ishlashning amaliy dasturi yaratiladigan dasturlash tilida qo'llanuvchi ma'lumotlarni tashkil etish va ishlash kontseptsiyasini aks ettiradi.

Fayllar modeli ma'lumotlarining asosiy tuzilmalari (strukturalari) – maydon, yozuv, fayl hisoblanadi. Yozuv ma'lumotlarini ishlashning asosiy tuzilma birligi operativ va tashqi xotira o'rtasidagi almashuv birligi hisoblanadi.

Maydon – ma'lumotlarni tashkil etishning oddiy birligi bo'lib, axborotning alohida, bo'linmas birligi rekvizitga mos keladi.

Yozuv – mantiqan bog'langan rekvizitlarga mos keluvchi maydonlar yig'indisidir. Yozuvning tuzilishi o'z tarkibiga kiruvchi har bir oddiy ma'lumotga ega maydonlar tarkibi va ketma-ketligi bilan belgilanadi.

Fayl – alohida maydonlarda mazmunga ega bo'lgan bir xil tuzilishdagi ko'plab yozuv nusxalaridir. Yozuv nusxasi maydonlarning konkret mazmunini o'z ichiga olgan yozuvlarni aks ettiradi. Fayl yozuvi tuzilishida maydon yagona mazmunga ega va guruhli ma'lumotlar mavjud emas (quyida qarang: ierarxik va tarmoqli modellar haqidagi bo'lim) bo'ladi. Har bir yozuv nusxasida yagona yozuv kaliti bir xil bo'ladi. Umumiy hollarda yozuv kalitlari ikki xil ko'rinishda bo'ladi: dastlabki (birlamchi) va ikkilamchi kalit.

Dastlabki kalit (DK) – yozuvni ma'no jixatidan bir xillashtiruvchi bir yoki bir necha maydonlar. Agar dastlabki kalit bir maydondan iborat bo'lsa u oddiy deyiladi, agar bir necha maydonli bo'lsa – turli tarkibli kalit hisoblanadi.

Ikkilamchi kalit (IK) – dastlabkididan farqli o'laroq, shunday maydonki, uning mazmuni faylning bir necha yozuvlaridan takrorlanadi, ya'ni u yagona bo'lmaydi. Agar dastlabki kalitning mazmuni bo'yicha faqat bitta yozuv nusxasi topilsa, ikkinchi kalit bo'yicha bir necha nusxa topilishi mumkin.

Ma'lumotlarning sanab o'tilgan tuzilishi bir qator MBBTda qo'llaniladi. Bu esa ushbu tushunchani ma'lum ma'noda universallashtiradi.

Indeksatsiyalash. Kalit bilan fayl yozuvlariga kirishning samarali vositasi indeksatsiyalashdir. Indeksatsiyalashda indeksli qo'shimcha fayl yaratiladi. U ma'lumotlar fayli kalitining barcha mazmunini tartiblashtirib o'zida saqlaydi. Indeksli faylda har bir kalit mazmuni uchun ma'lumotlar faylining tegishli yozuviga mo'ljallangan ko'rsatkich bo'ladi. Hajmi asosiy faylda kichik indeksli fayl mavjud bo'lganda berilgan kalit bo'yicha qidirilayotgan yozuv tez topiladi. Ma'lumotlar faylida yozuv ko'rsatkichi yordamida ushbu yozuvga bevosita yo'l ochiladi. Indeksatsiyalash faqat dastlabki emas, balki ikkilamchi kalit bo'yicha ham amalga oshirilishi mumkin.

Fayllar modeli ma'lumotlarini mantiqiy tashkil etishni tasvirlash.

Ma'lumotlarni mantiqiy tashkil etishni tasvirlashda har bir faylga yagona nom beriladi va uning yozuvlar tuzilmasi tasvirlanadi. Yozuvlar tuzilmalarini tasvirlash undagi maydonlar va ularning yozuv ichidagi joylashuv tartibini o'z ichiga oladi. Har bir maydon uchun qisqartma ko'rsatkich fayl nomi (yozuv ichidagi maydon identifikatori), maydon hajmi saqlanayotgan ma'lumot turi, maydon uzunligi va raqamli ma'lumotlarning aniqligi belgilab olinadi. Yozuvning yagona dastlabki kaliti vazifasini o'tovchi maydonlar uchun kalit belgisi ko'rsatiladi. 1 – jadvalda «Etkazib berish» fayli yozuvining tasvirlanishi misol qilib ko'rsatilgan. Bu erda yozuvning dastlabki kaliti turli tarkiblidir, chunki rekvizit belgilar NPS, KTOV, DATP yig'indisi muayyan yetkazib berishni aniqlaydi. Tovarlarni etkazib berishning miqdoriy tavsifi rekvizit asoslar KOLT, ST bilan tasvirlanadi .

Ma'lumotlar bilan amal bajarishning ilk tizimlari axborot ishlashning an'anaviy usullariga asoslanib tuzilgan edi. Har bir muayyan holat uchun tashqi foydalanuvchining o'z mantiqi ishlab chiqiladi. U axborot tuzilmasi, tanlash operatsiyasi, axborotni qo'shish va yo'q qilish kabi tushunchalarni o'z ichiga oladi. Ma'lumotlar va dastur o'rtasidagi o'zaro bog'liqlik yuzaga keladi: ma'lumotlarni o'zgartirishda, yo dasturni almashtirish yoki ma'lumotlarni qaytadan tuzish zaruriyati yuzaga keladi.

Murakkab axborotlarni ishlab chiquvchilar duch kelgan bu va boshqa qiyinchiliklar ma'lumotlar bilan amal bajarish uchun tizimlarga nisbatan standart talablarning shakllanishiga olib keldi. Asosiy talablardan biri – ma'lumotlarning iloji boricha mustaqil yoki axborot tuzilmasini fizik tushunchalardan alohida qilish kerak. Bunda hamma ma'lumotlar ko'p foydalanuvchilar kirishi mumkin bo'lgan holda ba'zi standart ichki tuzilishli qilib saqlanadi.[7]

Ma'lumotlar bazasi – axborotlar tizimlarining eng muhim tarkibiy qismi hisoblanadi. Oxirgi foydalanuvchi va ma'lumotlar bazasi administratorining ishini yengillashtirish uchun MBBT yaratilgan edi. Bu tizimlar ma'lumotlar bazasini amaliy dasturlardan ajratadi. MBBT dastur va apparat vositalarining murakkab kompleksi bo'lib, foydalanuvchi shu tufayli faqat ma'lumotlar bazasini mantiqiy

tashkil etishnigina tasavvur qiladi. Ma'lumotlar bazasini mantiqiy tashkil etish uni fizik amalga oshirishdan (ya'ni tashkil etish va fayllarni ishlashdan) sezilarli farq qilishi mumkin. Foydalanuvchilarning ixtiyorida talablar tili bo'lib ular yordamida foydalanuvchilar ma'lumotlarni tanlashi va o'zgartirishi mumkin.

Yuqorida ishonch hosil qilganimizdek, bu xotiralarning qo'shimcha sarflanish, tanlash va xizmat ko'rsatish vaqtining ko'payishini talab qiladi. Bundan tashqari, tuzilmalarni qayta tashkil etish xato qilish xususiyatiga ega bo'lgan dasturlar bilan bajariladi. Tizimli dasturlar foydalanuvchilarga oson bo'lmaganligi sababli, bunday xatolarni, ma'lumotlarni tasdiqlash operatsiyalari orqaligina aniqlash mumkin. Bu turli xil axborot tuzilmalarini samarali ta'minlaydigan tizimlarni loyhalashni qiyinlashtiradi. Bu qiyinchiliklarni yengish uchun zamonaviy MBBTlar quyidagi talablar javob berishi lozim:

- ma'lumotlarning mustaqilligi;
- talablarning kuchli tili;
- javobning qisqa vaqti;
- ma'lumotlar va kataloglarni qayta tashkil etishni qisqartirish yoki ulardan voz kechish.

Ma'lumotlarning mustaqilligi – MBBTga asosiy talab, talablarning kuchli tili esa foydalanuvchining talablarini qondirishning muhim shartidir. Bu tillar assotsiativ manzillashtirish va ma'lumotlar to'plami bilan amallar bajarish vositalariga ega. Bu esa o'z navbatida EHMLardan samarali foydalanishga sharoit yaratadi.

Ma'lumotlar bazasini boshqarish tizimi. Bu quyidagilarga imkon beruvchi dasturiy vositalarning to'plamidir.

a) foydalanuvchilarni ma'lumotlarni aniqlash va amallar bajarish tili vositalari bilan ta'minlaydi. Bunday vositalarga ma'lumotlarni aniqlash tili (MAT) va ma'lumotlar bilan amallar bajarish(MABT) kiradi. Ma'lumotlar tili atamasi aytib o'tilganlarining har ikkalasini yoki ulardan birini anglatadi. Ma'lumotlar so'zi ma'lumotlar tilini C++, Paskal va h.k. kabi tillar turidan farqlaydi. Lekin ma'lumotlar tili universal tilga, masalan C++, Paskalga kiritilishi mumkin. Bunday

holda dasturlashning universal tili va ma'lumotlar tili tegishli ravishda (kirituvchi) til va ma'lumotlarning til osti deb ataladi.

b) foydalanuvchi ma'lumotlarining modelini qo'llab-quvvatlashni ta'minlaydi. Ma'lumotlar modeli – ba'zi ilovaga tegishli fizik ma'lumotlarning mantiqiy taqdim etilishini aniqlash vositasidir.

v) aniqlash, yaratish va mantiqiy ma'lumotlar bilan amallar bajarish (ya'ni tanlash, yangilash, kiritish va yo'q qilish)ga imkon beruvchi MAT va MABT funktsiyalarini amalga oshiruvchi dasturlarni ta'minlaydi.

g) ma'lumotlarning himoyasi va yaxlitligini ta'minlaydi. Tizimdan foydalanish faqat shunga huquqi bo'lgan foydalanuvchigagina ruxsat etiladi. Foydalanuvchilar ma'lumotlar ustida operatsiya bajarayotganlarida saqlanayotgan ma'lumotlarning muvofiqligi (yaxlitligi) ta'minlanadi. Gap shundaki, MBBT ko'plab foydalanuvchilar ishi jamoa rejimiga asosan tuziladi.

Relyatsion model (inglizcha relation – munosabat) o'tgan asrning 70- yillari boshida Kodd tomonidan yaratilgan. Modelning soddaligi va egiluvchanligi ma'lumotlar bazalari tuzuvchilarini e'tiborini o'ziga tortdi. 80-yillarga kelib keng taraqqiy eta boshladi va relyatsion MBBTlar sanoat standarti darajasiga ko'tarildi.

Model relyatsion algebrasi tushunchalari tizimiga tayanadi. Bu tushunchalarning eng muhimlari jadval, satr, ustun, munosabat va birlamchi kalitlar sanaladi, bajariladigan amallar esa jadvallar bilan ish olib borishda aks etadi.

Relyatsion modellarda axborot to'g'ri to'rtburchakli jadvallar ko'rinishida aks ettiriladi. Har bir jadval ustun va satrlardan tashkil topgan bo'ladi va ma'lumotlar bazalari ichida o'zining takrorlanmas nomiga ega.

Jadval real olam axborotini – mohiyatini aks ettiradi, uning har bir satri (yozuvi) esa ob'ektning aniq bir nusxasini – nusxa mohiyatini aks ettiradi. Jadvalning har bir ustuni ushbu jadvalda o'ziga xos nomga ega. Jadval kamida bir ustunga ega bo'lishi kerak.

Ma'lumotlarning relyatsion modeli yuqorida ko'rib chiqilgan tarmoqli va hierarxiklardan foydalanuvchi uchun qulay bo'lgan jadvalli tasavvurlar va

ma'lumotlarga kirishning oddiy tuzilishi bilan farq qiladi. Ma'lumotlarning relyatsion modeli oddiy ikki o'lchamli jadval – **munosabat (model ob'ektlari)larning** yig'indisidir. Relyatsion modeldagi relyatsion bog'langan ikki jadvallar orasidagi mantiqiy bog'langan aloqalar jadval munosabatlariga tegishli bir xil atributlarning mazmun jixatidan tengligiga ko'ra o'rnatiladi.

Jadval-munosabat relyatsion modellarning universal ob'ekti hisoblanadi. Bu relyatsion modelni ta'minlovchi turli MBBTlardagi ma'lumotlarni birxillashtirish imkonini beradi. Relyatsion modellarni ishlash operatsiyalari munosabatlar algebrasi va relyatsion hisob-kitoblarning universal apparatidan foydalanishga asoslangan.

Jadval relyatsion model ma'lumotlari (ob'ekti)ning asosiy turi hisoblanadi. Jadvalning tuzilishi ustunlarning yig'indisi bilan belgilanadi. Jadvalning har bir satrida tegishli ustunga mos keluvchi bittadan mazmun joylashgan bo'ladi. Jadvalda ikkita bir xil satr bo'lishi mumkin emas. Satrlarning umumiy soni chegaralanmagan.

Ustun ma'lumotlarining ba'zi tarkibiy qismi - **atributga** mos keladi. Atribut ma'lumotlarning eng oddiy tuzilmasidir. Jadvalda yuqorida ko'rib o'tilgan tarmoqli va ierarxik modellardagi kabi ko'p tarkibiy qismlari guruh yoki takrorlanuvchi guruh kabilar belgilanishi mumkin emas. Jadvalning har bir ustuni ma'lumotlar tegishli tarkibiy qismi (atribut)ning nomiga ega bo'lishi kerak. Ma'nosi jadval satriga teng bo'lgan bir yoki bir nechta atributlar jadvalning **kaliti** hisoblanadi.

Ma'lumotlar bazasini tuzishda relyatsion yondashuv munosabatlar nazariyasining terminologiyasi qo'llaniladi. Eng oddiy ikki o'lchamli jadval munosabat sifatida belgilanadi. Tegishli atribut mazmuniga ega bo'lgan jadval ustuni **domen** deyiladi. Turli atributlar mazmuniga ega bo'lgan satrlar esa **kortej** deb ataladi.

Relyatsion jadval-munosabat. R relyatsion jadval munosabatining ko'rinishi berilgan. R munosabat (relyatsion jadval)ning formal ta'rifi uning domenlari D_i (ustunlari), kortejlari K_i (satrlari) haqidagi tushunchaga tayanadi. Ko'plab

domenlar $\{D_i\}$ belgilangan R munosabat deb, $D_1(D_2(D_3..... D_n$ domenlarini **dekart (bevosita) ishlab chiqaruvchi ko'plikka** aytiladi.

Jadval-munosabat ma'lumotlar tarkibiy qismi atributlar $(A_1, A_2,)$ nomiga ega bo'lgan ustunlarni o'z ichiga olgan d atributlarning mazmuni jadvalning asosiy qismida joylashgan bo'lib satrlar va ustunlarni tashkil qiladi. Bir ustunda atributlar mazmunining ko'pligi domen D_i ni hosil qiladi. Bir satrda atributlar mazmunining ko'pligi bir kortej K_j ni hosil qiladi. R munosabat ko'plab tartibga solingan kortejlar orqali hosil bo'ladi: $R = \{ K_j \}, j = 1-m$ $K_j = \{d1_j, d2_j,.....dn_j\}$

n – munosabat domenlarining soni; munosabatlarning ko'lamini belgilaydi.

J – kortej nomeri;

k – munosabatdagi kortejlarning umumiy soni bo'lib, munosabat koordinata soni deyiladi.

Jadval-munosabatning kaliti. Kortejlar jadval-munosabat ichida takrorlanmasligi zarur va ular tegishli yagona identifikator – dastlabki (birlamchi) kalitga ega bo'lishi kerak.

Dastlabki (birlamchi) kalit atributdan tashkil topgan bo'lsa **oddiy**, bir necha atributdan tashkil topganda esa **turli tarkibli** deb ataladi. Munosabatda dastlabki kalitdan tashqari ikkilamchi kalit ham bo'lishi mumkin.

Ikkilamchi kalit – mazmuni turli satr – kortejlarda takrorlanishi mumkin bo'lgan kalitdir. Ular bo'yicha ikkinchi kalitning bir xil mazmuni satrlar guruhi izlab topiladi.

Satrlar ustunlardan farqliroq o'z nomlariga ega emas, ularning jadvalda joylashish tartibi aniqlanmagan va satrlar soni mantiqan chegaralanmagan bo'ladi. Satrni tartib raqamiga ko'ra tanlab olib bo'lmaydi. Faylda har bir satr o'z raqamiga ega bo'lsa ham, bu narsa satrni tavsiflamaydi. Bu raqam satr jadvaldan olib tashlanganda o'zgaradi. Mantiqan satrlar o'rtasida “birinchi” va “oxirgi” degan tushuncha yo'q.

2.2.1. jadval. Relyatsion jadval-munosabat R munosabat = {K1, K2.....}

A1	A2	A3	A4	A5	A6
d11	d21	d31	d41	D51	d61
d12	d22	K2 кортежи			d62
d13	d23	d33	d43	Д о м е н	d63
.....					
.....					
.....					
D1	D2	D3	D4	D5	D6

atributlar (ustunlar)
nomi

Korteji
K2 = {d12, d22, d32, d42, d52, d62}

Domen
D5 = {d51, d52, d53, d54,..}

R – relyatsion jadval munosabatning tasviri

Relyatsion tizimlarning qo'llanilishi murakkab navigatsiya zaruriyatni bartaraf qildi. Chunki ma'lumotlar endi bir fayl ko'rinishida emas, balki mustaqil naborlar asosida tuzilib, ma'lumotlarni tanlab olish uchun amaliy to'plamlar nazariyasi – relyatsion algebra amallari qo'llaniladi.

Jadvalning har bir satrini bir ma'noli nomlovchilari bo'lgan ustun yoki ustunlar to'plami relyatsion model jadvalida mavjud bo'lishi kerak. Ushbu ustun yoki ustunlar to'plami jadvalning birlamchi kaliti deb ataladi.

Agar jadval birlamchi kalitning unikalligi talablarini qanoatlantirsa, u holda munosabat mavjud deyiladi. Relyatsion modellarda barcha jadvallar munosabatlarga o'zgartirilgan bo'lishi kerak. Relyatsion model munosabatlari biri biri bilan bog'langan bo'ladi. Bog'lanishlar tashqi kalitlar bilan ta'minlanadi.

Tashqi kalit – boshqa munosabatning (jadvalni) birlamchi kaliti qiymatini bir ma'noda tavsiflovchi ustun (ustunlar to'plami). Boshqacha aytganda, tashqi kaliti aniqlangan munosabat ustunlari to'plami birlamchi kalit bo'lib xizmat qiluvchi boshqa mos munosabatga murojaat qilishi demakdir.

Relyatsion jadval (munosabat) sxemasi jadval yozuvini tashkil etuvchi maydonlar nomi to'plamini o'zida aks ettiradi:

JADVAL NOMI (Maydon 1, Maydon 2,Maydon n).

Relyatsion modellarning hozirgi zamon MBBTlarida ustunlikka erishishi quyidagi omillar bilan aniqlanadi:

- 1) rivojlangan nazariyaning mavjudligi (relyatsion algebrani);
- 2) ma'lumotlarni boshqa modellarini relyatsion modellarga keltirish apparatining mavjudligi;
- 3) axborotga ruxsatli kirishni tezlashtirishni maxsus vositalarini mavjudligi;
- 4) tashqi xotirada MB aniq fizik tashkil etish bilimiga ega bo'lmay ular bilan ish olib borish imkoniyatini yaratadigan MBga nisbatan standartlashgan yuqori darajadagi so'rovlar tilining mavjudligi.

Yuqoridagilardan quyidagi qisqa xulosalarni chiqarish mumkin:

1. Ma'lumonlarning mantiqiy modelini yaratishdan asosiy maqsad turli-tuman axborotlarni tizimlashtirish va uning xususiyatlarini mazmuniga ko'ra, hajmiga, bog'lanishlariga, barcha foydalanuvchilar toifalarining axborotga bo'lgan talabini qondirish dinamikasiga ko'ra aks ettirilishidir.

2. Foydalanuvchining tashqi modeli ushbu foydalanuvchining mantiqiy modeliga nisbatan kontseptual talablarini aks ettiradi hamda foydalanuvchining mantiqiy modeli asosida predmet sohasini tasavvurlariga mos keladi. Demak, tashqi model qanchalik puxta loyhalashtirilgan bo'lsa, axborot modeli predmet

sohasini shunchalik aniq va to'liq aks ettiradi hamda ushbu predmet sohasini avtomatlashtirilgan tizim boshqaruvi shunchalik aniq va to'liq ishlaydi.

3. Predmet sohasining har qanday axborot tizimini ishlab chiqarishni zaruriy boshlang'ich bosqichi hisoblanadi. Aynan ushbu bosqichda bo'lajak tizim foydalanuvchilari barcha to'plamining axborot talabini aniqlaydi va o'z navbatida ma'lumotlar bazasini moxiyati aniqlab olinadi.

4. Mashina muxitida ma'lumotlarni tashkil etish ikki: mantiqiy va fizik bosqichlar bilan tasvirlanadi. Ma'lumotlarni bevosita mashina (tashuvchisi)(da joylashtirish usulini belgilaydi. Ma'lumotlarni zamonaviy amaliy dasturlar vositalarida tashkil etishning bu bosqichi foydalanuvchining aralashuvisiz avtomatik ravishda ta'minlanadi.

5. Har qanday axborot tizimlarini asosini maxsus tarzda tuzilgan – ma'lumotlar bazalari tashkil etadi. Ma'lumotlar bazalari – ma'lumotlar to'plamini taqdim etishni va tashkil etishni ob'ektiv shakli bo'lib(masalan maqola, hisob-kitoblar) bunda ushbu ma'lumotlar EHM orqali qidirib topiladigan va ishlov beriladigan holatda tizimlashtirilgan bo'lishlari kerak.

6. Ma'lumotlarning mustaqilligi – MBBTga nisbatan asosiy talab asoslanadi, talablarning kuchli tili esa foydalanuvchining talablarini qondirishning muhim shartidir. Bu tillar assotsiativ manzillashtirish va ma'lumotlar to'plami bilan amallar bajarish vositalariga ega. Bu esa o'z navbatida EHMLar yangi qurilmalarining ichida ulardan samarali foydalanishga sharoit yaratadi.

2.2. Ma'lumotlar bazasi boshqarish tizimi arxitekturası

MBBT alohida olingan modullardan tashkil topgan:

- MB boshqarish bloki disklardagi ma'lumotlar bilan foydalanuvchi dasturi va tizimning so'rovi (guery) orasidagi interfeysni aniqlaydi.
- Fayl menedjeri-ma'lumotlar tuzilmasi bilan disklar o'rtasidagi bog'lanishni boshqaradi.
- Guery-protessor ingliz tilida yozilgan guery gaplarini MBni boshqarish bloki tushunadigan tilga o'tkazadi.

- Prekompilyator DML(Data Manipulation Language) ma'lumotlar bilan manipulyatsiya qiladigan til bo'lib, u quydagi operatsiyalarga javob beradi:
 - MBdan ma'lumotlarni ajratib olish,
 - MBga ma'lumotlarni kiritish,
 - MBdan ma'lumotlarni olib tashlash,
 - MBni modifikatsiya (o'zgartirishlar) qilish.

Kompilyator DDL(Data Definition Language)-MB tilini, uning tuzilmasini va tashqi xotiralardagi axborot turini aniqlaydi. MBning tuzilmasi ko'pincha jadval shaklida bo'ladi.

Shuni ta'kidlash lozimki, hozirgi vaqtda deyarli barcha MBBSlar asosan relyatsion modellar asosida tashkil qilinmoqda. Shuni nazarda tutgan Microsoft Office korporatsiyasi ham eng ommalashgan dastur vositalarga ega, bu dastur vositalari ixtiyoriy sohada yuqori darajadagi professional hujjatlar tayyorlash imkonini beradi. Shulardan biri MBlar bilan ishlashga mo'ljallangan Microsoft Access dasturi bo'lib, bu programma Visual Basic for Application dasturlash muxitida makroslar yaratish va boshqa bir qancha imkoniyatlarga egaki, bu foydalanuvchiga har tomonlama mukammal bo'lgan hujjatlar tayyorlashga yordam beradi.[8]

Microsoft Office ning o'zbek tilidagi varianti yo'qligi, uning faqat ingliz va rus tilida yaratilgan versiyalaridangina foydalanish imkoniyatiga ega ekanligimizdan mazkur fikrlar Microsoft Access ning ruscha versiyasiga tayanib yozilgan. Microsoft Access dasturi ham relyatsion modellar asosiga ko'rilgan bo'lib, unda tashkil qilinadigan MBlar jadval ko'rinishida aks etadi. Bunday jadvaldagi ustunlar maydon deb, satrlar esa yozuv deb ataladi.

Maydon – ma'lumotlarni tashkil etishning oddiy birligi bo'lib, ma'lumotning alohida, bo'linmas birligiga egaligi rekvizitga mos keladi.

Yozuv–mantiqiy bog'langan rekvizitlarga mos keluvchi maydonlar yig'indisidir. Yozuvning tuzilishi o'z tarkibiga mos har bir oddiy ma'lumotga ega maydonlar tarkibi va ketma-ketligi bilan belgilanadi.

Demak, maydon MBning asosiy tuzilmali elementi bo'lib quydagi parametrlar bilan ifodalanadi:

- uzunligi (belgi simvollarda ifodalanib baytlarda ulchanadi),
- nomi (maydonning o'ziga xos alohida xususiyati),
- podpis- imzo (ustun sarlavhasi haqida ma'lumot).

Maydonlar xususiyatiga va tarkibiga qarab quyidagi turlarga bo'linadi:

1. Matnli maydon.
2. Sonli maydon.
3. Vaqt va sanani ifodalovchi maydon.
4. Mantiqiy maydon (1(0; Ha yoki yo'q; rost yoki yolg'on kabi mantiqiy birliklar bilan ifodalanadi).
5. Pul birliklarida ifodalangan maydon (raqamlar pul birliklari bilan birgalikda ifodalanadi)
6. OLE maydoni (shakl, tasvir, rasm, musiqiy kliplar va videoyozuvlar shaklida ifodalanadi)

MEMO maydoni- matn uzunligi 256 simvoldan uzun bo'lgan maydonda faqat matnning qaerdaligini ifodalovchi ko'rsatkich turadi. Bu holda har bir maydonda 65 535 simvol saqlanishi mumkin.

Schetchik (sanovchi) maydoni - maydonda turgan ifoda avtomatik ravishda sanalib o'zgaradi.

2.3. MySQL dasturlash tili va uning imkoniyatlari

Har qanday korxona, tashkilot yoki muassasa uchun axborot muhim ahamiyatga ega. Kelayotgan axborotlardan esa kerakli ma'lumotlar olinadi. Korxona, tashkilot yoki muassasalarni axborotlar tizimi deb qarash mumkin, chunki ma'lumotlar ularda to'planadi, qayta ishlanadi va uzatiladi. Bu vazifalarni bajarish uchun esa axborotlar texnologiyasidan foydalaniladi.

Loyihalanayotgan tizim arzon, tezkor va ishonchli bo'lishi, katta hajmdagi ma'lumotlarni saqlash, qayta ishlash imkoniyatiga ega bo'lishi kerak. Eng kam

harajatli texnika va texnologiyalar bilan katta tezlik va ishonchlikka erishish zarur.

Bu dasturimiz bazasining eng asosiy vazifalaridan biri tezkor hisob-kitob tizimidir. Bu hisobotlarni tayyorlash jarayonida ularni aniqlashtirilgan ko'p kriteriyalar -shartlar asosida qidirish va bosmadan chiqarish mumkin bo'lishi kerak. Qidirish jarayoni katta tezlik bilan amalga oshirilishi uchun ma'lumotlar bazasi tugri loyhalashtirilishi, to'g'ri konfiguratsiya qilinishi, jadvallar strukturasi optimal loyhalashtirilishi, jadvallarni kerakli indekslar bilan ta'minlanishi, bazadagi protsedura va funktsiyalar, so'rovlar optimizatorlar yordamida optimizatsiya qilinishi lozim. Hisobotlarni kerakli formatlarda saqlash imkoniyati bo'lishi kerak. Buning uchun hisobot tayyorlash uchun mo'ljallangan dasturiy ta'minotlarni ham to'g'ri tanlash kerak bo'ladi. Aynan shu xususiyati bilan tizim korxonaning keyingi faoliyatini tezkorlik bilan rejalashtirishga imkon beradi.

Yuqoridagi talablarni hisobga olgan holda Elektron omborni loyihalashtirish va amaliyotda qo'llash jarayonida ko'pgina muammolarni xal qilishga to'g'ri keladi. Eng awalo, kerakli hodimlargagina belgilangan ruxsatlarni berish imkoniyati, ish jarayonida bir vaqtda bir maxsulot ustida operatsiya bajarishga harakat qilish, omborda saqlanayotgan ma'lumotlardan rezerv nusxa olish. Bu muammolar ma'lumotlar bazalarini boshqarish tizimlari (MBBT) yordamida hal qilinadi. Hozirga kunda mavjud MBBT-lar o'zining foydalanuvchilari ro'yhati mavjud bo'lib, bu ro'yhatdagi foydalanuvchilarga ma'lum bir ruxsatlar to'plami - rollar belgilanadi. Rezerv nusxa olishni ham turli usullar bilan tashkil qilish mumkin. Masalan, stand by serverlardan foydalangan holda issiq va sovuq nusxa olish mumkin. Issiq nusxa olish usulida MBBT har bir o'zgarishni asosiy bazada saqlab qo'yishdan tashqari rezerv bazaga ham saqlab qo'yadi. Sovuq nusxa olish usulida esa bazadagi o'zgarishlar sodir bo'lishi bilan emas, balki ma'lum bir optimal belgilangan period orasida rezerv bazada saqlab qo'yiladi. Undan tashqari rezerv baza ham shu yagona serverda yoki rezerv serverda bo'lishi mumkin. Operatsion tizimda mavjud dasturiy ta'minotlar (Active Directory kabi), antivirus dasturlari

yordamida nafaqat ombor, balki butun korxona axborot xavfsizligi muammolari xal qilinadi.

Bu dasturimiz bazasining eng asosiy vazifalaridan biri tezkor hisob-kitob tizimidir. U berilgan period uchun istalgan kirim chiqimlar, ichki chiqimlar va yaroqliylik xususiyatini yo'qotgan maxsulotlar haqidagi hisobotlarni berishi kerak. Bu hisobotlarni tayyorlash jarayonida ularni aniqlashtirilgan ko'p kriteriyalar - shartlar asosida qidirish va bosmadan chiqarish mumkin bo'lishi kerak. Qidirish jarayoni katta tezlik bilan amalga oshirilishi uchun ma'lumotlar bazasi to'g'ri loyihalashtirilishi, to'g'ri konfigurasiya qilinishi, tablitsalar strukturasi optimal loyihalashtirilishi, tablitsalar kerakli indekslar bilan ta'minlanishi, bazadagi prosedura va funksiyalar, so'rovlar optimizatorlar yordamida optimizasiya qilinishi lozim. Hisobotlarni kerakli formatlarda saqlash imkoniyati bo'lishi kerak. Buning uchun hisobot tayyorlash uchun mo'ljallangan dasturiy ta'minotlarni ham to'g'ri tanlash kerak bo'ladi. Aynan shu xususiyati bilan tizim korxonaning keyingi faoliyatini tezkorlik bilan rejalashtirishga imkon beradi.

Bizga ma'lumki, respublikamizdagi korxonalarning ko'p qismi ishchi o'rin sifatida Microsoft Windows operatsion tizimidan foydalanadi. Demak, tizim korxonaga biriktirilayotganda ishchi o'rinlari uchun alohida operatsion tizim tanlash muammosidan halos qilish kerak. Buning uchun Microsoft Windows oilasiga mansub operatsion tizim bilan mos MBBT ni tanlash kerak. My SQL Server oilasiga mansub MBBT lar Microsoft Windows operatsion tizimi bilan mos. Bundan tashqari dasturiy ta'minotni ham Microsoft Windows operatsion tizimiga mos PHP dasturiy ta'minoti yordamida yaratish mumkin. U dasturchilarga zarur bo'lgan juda ko'p instrumentariylar bilan jihozlangandir.

Shuni aytib o'tishimiz joiz bo'ladiki, PHP, MySQL dasturiy ta'minotlari birgalikda do'stona ishlaydilar.

Bunda PHP - dasturlash tili, kod yozish dasturi va MySQL - baza hisoblanadi.

2.4. PHP dasturlash tili haqida

PHP 1995 yili, Rasmus Lerdorf Perl/CGI da Internet saytga kiritilganining qanchaligini hisoblovchi kichik bir programmacha yozdi. O'sha vaqtda bunday ishlarni bajaruvchi xech qanday programma bo'lmagan shuning uchun unga judayam ko'p xatlar kelib, uni qanday qilganligini so'rab surishtirishni boshlashgan. Shunda Lerdorf o'zining bu kichik programmasini ya'ni PHP (Personal Home Page yoke Hypertext Processor) ni tekinga tarqatishni boshlagan. Bundan tez tarqalib, juda ko'p qo'llanishni boshlanganini ko'rgan Lerdorf PHP ni ishini ya'ni xususiyatlarini, ko'shimcha komandalarini kengaytirishni boshlaydi. Vaqti kelib Perl uning uchun kamlik qilib u bu programmani C programmalash tilida yozishni boshladi va PHPni ishlab chiqarishda butun Dune programistlari qatnasha boshlashdi.

1997 yili PHP 3.0 versiyasi chiqdi va butun dunyo bo'yicha 50 000 ortiq insonlar shu programmani ishlatishni boshlagan edi. 1999 yilga kelib esa PHP ishlatuvchilar soni 1 000 000dan ortib ketdi.

PHP – bu serverda qayta ishlanuvchi stsenariylar tilidir. ASP kabi PHP kodlar ham bevosita HTML-hujjatni tarkibiga ko'shiladi. Ushbu tilning nomi Personal Home Page Tools so'zlarning qisqartmasidan olingan. PHP da C va Perl tillarida uchragan bir qator muammolar xal etilgan, bundan tashqari, PHP ma'lumotlar bazasi bilan ishlash uchun qulay vositadir. Umuman olganda Perl, PHP – ochiq tizimli tillar hisoblanadi va ularni dasturchilar modernizatsiyalashtira oladi.

PHPni ba'zi komponentlari haqida.O'zgaruvchilar va konstantalar.

PHP da o'zgaruvchilar dollar (\$) ,belgisida boshlanadi. Bu simvoldan tashqari ixtiyoriy sondagi harf, raqam va otiga chiziq simvollari kelishi mumkin, lekin birinchi simvol albatta harf bo'lishi kerak. Shuni esda tutish kerakki, PHPda o'zgaruvchilarning nomlari kalit so'zlardan farqli registrga bog'liqdir.

PHPda o'zgaruvchilarning ta'riflaganda oshkora tipini ko'rsatish shart emas va dastur davomida bita o'zgaruvchi har xil tiplarga ega bo'lishi mumkin.

O'zgaruvchi unga qiymat berilganda initsializatsiya qilinadi va dastur bajarilguncha mavjud bo'ladi. Ya'ni web-saxifa holida to so'rov tugamaguncha.

Tashqi o'zgaruvchilar

Klient so'rovi web-server tomonidan tahlil qilinib, PHP mashinaga uzatilgandan so'ng, u so'rovga tegishli ma'lumotlarni o'z ichiga olgan va bajarish davomida murojat qilish mumkin bo'lgan bir necha o'zgaruvchilarni yaratadi. Oldin PHP sizni tizimingiz atrof-muxit o'zgaruvchilarini oladi va u nomdagi vash u qiymatdagi PHP stsenariysi atrofidagi o'zgaruvchilar yaratadi, toki serverdagi stsenariylarga klient tizimi xususiyatlari bilan ishlash mumkin bo'lsin. Bu o'zgaruvchilar **\$HTTP_ENV_VARS** fassotsiativ massivga joylashtiriladi.

Tabiiiki **\$HTTP_ENV_VARS** massivi o'zgaruvchilari tizimga bog'liqdir (chunki ular aslida atrof muxit o'zgaruvchilaridir). Atrof muxit o'zgaruvchilari qiymatlarini sizni mashinangiz uchun eng (UNIX) yoki set (Windows) komandasi yordamida ko'rishingiz mumkin.

So'ngra PHP GET- o'zgaruvchilarning guruhini yaratadi. Ular so'rov satrini tahlil qilishda yaratiladi. So'rov satri **\$QUERY_STRING** o'zgaruvchida saqlanadi va so'ralgan URL dagi "?" simvoldan keyingi informatsiyadan iborat. PHP so'rov satrini & simvollar bo'yicha alohida elementlarga ajratadi, va har bir elementda "=" belgisini qidiradi. Agar "=" belgisi topilgan bo'lsa, tenglik ap tomonidagi simvollaridan iborat o'zgaruvchi yaratadi.

Konstantalar PHPda define() funktsiyasi yordamida e'lon qilinadi:

define(SONSTANT, value)

Operatorlar

2.4.1-jadval.Arifmetik operatorlar

+	Qo'shish
-	Ayrish
*	Ko'paytirish
/	Bo'lish
%	Modul bo'yicha qoldiqni xisoblash (masalan:5% 2=1)

Razryadli operatorlar: Bu operatorlar guruhi butun qiymatli operatorlarning bitli tasvirlari bilan ishlaydi. Asosan bu operatorlar mantiqiy bayroqlar to'plamini yaratish uchun ishlatiladi. Bizning misollarimizda bu operatorlarni kam ishlatamiz, lekin quyida shu operatorlardan foydalanishga misol keltiramiz.

2.4.1-jadval.Razryadli operatorlar

&	Razryadli kon'yunktsiya(AND)
	Razryadli diz'yunktsiya (OR)
^	Razryadli ajratuvchi YOKI (XOR)
~	Razryadli inkor (NOT)
<<	Chap butun qiymatli operand batli tasvirini o'ng butun qiymatli operand qiymatiga teng razrayadlar soniga chapga siljitish
>>	Chap butun qiymatli operand batli tasvirini o'ng butun qiymatli operand qiymatiga teng razryadlar soniga o'nga siljitish

Misol:

<?

echo(4<<2);//teng 16

*echo"
"*;

?>

sikl operatorlari/ While

Operator **While** oldingi shartli sikl operatori deyiladi, siklga kirishda oldin shartli ifoda hisoblanadi, agar uning qiymati noldan farqli bo'lsa sikl tanasi bajariladi. Shundan so'ng shartli ifodani hisoblash va sikl tanasi operatorlarini bajarish, shartli ifoda qiymati nolga teng bo'lgunga qadar davom etadi. **While** operatoridan har xil ketma-ketliklarni ko'rish qulay, agar ularda oldindan ma'lum so'ngi simvol mavjud bo'lsa. (Bu ayniqsa C++ tilida qulay, chunki C++ tilida satr bu char tipidagi nolinchi simvol bilan tugovchi simvollar ketma-ketligidir).

Sikl operatorlari / Do... while

Bu operator keyingi shartli sikl operatori deyiladi. Ixtiyoriy holda siklga kirilganda sikl tanasi bajariladi (Ya'ni sikl juda bo'lmasa bir marta bajariladi) so'ngra shart hisoblanadi va agar u 0 bo'lsa Yana sikl tanasi bajariladi. Quyidagi misolda nol ro'yhatga ($++\$i \leq \var) shartga bog'lanmagan holda qo'shiladi.

Massivlar

Massivlar initsializatsiyasi

PHPda massivlarni initsializatsiya qilishning 2 usuli mavjud. Birinchisi massiv elementlariga qiymat berishdan iborat:

```
<?
$car[] = "passenger car";
$car[] = "land-rover";
echo($car[1]); // chiqaradi "land-rover"
?>
```

Massiv indeksini ochiq ko'rsatish mumkin:

```
<?
$car[0] = "passenger car";
$car[1] = "land-rover";
echo($car[1]); // chiqaradi "land-rover"
?>
```

PHP da funktsiyalar

Funktsiyalarni ta'riflash va chaqirish

Funktsiyani function kalit so'zi yordamida e'lon qilinadi. Bu kalit so'zdan so'ng figurali qavs ichida funktsiya tanasini hosil qiluvchi har xil operatorlar yoziladi:

```
function My Function()
{ // operatorlar }
```

Agar funktsiya argumentlar qabul qilsa, ular funktsiya ta'rifida o'zgaruvchilarsifatida yoziladi. **Function argumenti** funktsiya tanasida keyingi amallarda qo'llanish uchun uzatiladigan o'zgaruvchidir. Agar funktsiya bittadan argumentga ega bo'lsa, bu argumentlar vergul bilan ajratiladi:

function My Function(\$var, \$var1, \$var2)

Agar funktsiya biror qiymat qaytarsa, funktsiya tanasida albatta **return** operatori mavjud bo'lishi kerak:

Fayllar bilan ishlash

Fayllarni ochish

Fayl biror ma'lumot saqlash fizik qurilmasidagi baytlar ketma-ketligidir. Har bir fayl joylashuvini ko'rsatuvchi absolyut yo'lga ega. Yo'l ajratuvchi sifatida Windows da to'g'ri slesh (/) yoki teskari (\) ishlatilishi mumkin. Boshqa operatsion tizimlarda faqat to'g'ri ishlatiladi.slesh

Fayllarni server fayl tizimida ochish **fopen** funktsiyasi yordamida amalga oshiriladi:

int fopen (string filename, string mode [, int use_include_path])

Birinchi argument **filename** – fayl nomi yoki unga olib boruvchi absolyut yo'l. Agar absolyut yo'l ko'rsatilmasa, fayl kattalogda joylashgan bo'lishi kerak.

Ikkinchi argument **mode** fayl qaysi amallar uchun ochilganligini ko'rsatadi va quydagi qiymatlarga ega bo'lishi mumkin:

- **r** (faylni faqat o'qish uchun ochish; ochilgandan so'ng fayl ko'rsatkichi fayl boshiga o'rnatiladi);

- **r+** (faylni o'qish va yozish uchun ochish; ochilgandan so'ng fayl ko'rsatkichi fayl boshiga o'rnatiladi) ;

- **w** (Yozish uchun yangi bo'sh fayl yaratish; agar shu nomli fayl mavjud bo'lsa, undagi hamma ma'lumot o'chiriladi);

- **w+** (Yozuvlarni ochish uchun yangi bo'sh fayl yaratish; agar shu nomli fayl mavjud bo'lsa, undagi hamma ma'lumot o'chiriladi);

- **b** (ikkilik fayl bilan ishlash usulini (o'qish va yozish))ko'rsatuvchi bayroq; faqat Windows da ko'rsatiladi).

Uchinchi shart bo'lmagan argument **use_include_path** fayllar **include_path** kattalogida ishlash kerakligini belgilaydi. (**include_path** parametr php /'ini faylda o'rnatiladi).

Fayl muvaffaqiyatli ochilganda, fopen funktsiyasi fayl deskriptorini qaytaradi, aks holda – false qaytaradi. Fayl deskriptori ochilganda faylga ko'rsatkich bo'lib, operatsion tizim tomonidan shu fayl bilan amallarni qo'llash uchun ishlatiladi. Funktsiya tomonidan qaytarilgan fayl deskriptorini keyinchalik shu fayl bilan ishlaydigan hamma funktsiyalarda ko'rsatish lozim.

III-BO`LIM. “FUQAROLAR NAFAQA TA’MINOT BO’LIMINING MA’LUMOTLAR BAZASI INTERFEYSINI YARATISH

3.1. Dasturiy vositalarga qo’yilgan talablar.

Dasturiy ta'minot ma'lumotlar bazasi MySQL MBBT da yaratilgan. MySQL kerakli darajadagi operativ xotira va diskdagi bo'sh joyga ega bo'lgan tezkor tizimda ishlatilishi kerak. Quyida apparat ta'minotiga bo'lgan asosiy talablar keltirib o'tiladi:

- Prosessor. MySQL versiyalari Intel x86 yoki shunga mos prosessorlarda, 64-razryadli versiyalari esa Intel Itanium (IA - 64) va x64 oilasiga mansub prosessorlarda ishlaydi. Katta kesh xotiraga ega prosessorlar ishlatilganda sezilarli samaradorlikka erishiladi.

Bundan tashqari moddiy jihatdan hisoblaganimizda bu anchagina sarf - harajat qilishga olib kelishi mumkin. Buning o'rniga biz "Celeron"lar oilasiga mansub bo'lgan chastotasi 3000 Grs(CPU 3.06 Grs yoki CPU 1.8Grs d.c) tavsiya qilishimiz mumkin.

- Diskdagi bo'sh joy. Ma'lumot saqlash uchun bo'sh joyning hajmi server qo'llab quvvatlaydigan ma'lumotlar bazasining soni va hajmi bilan aniqlanadi. Diskdagi bo'sh joyning hajmi barcha ma'lumotlarni, shuningdek, indekslar, tizim fayllari, virtual xotira, tranzaksiya jurnallari va h.k. larni sig'dira oladigan bo'lishi kerak. Xotira hajmidan tashqari jamlagichning o'tkazish qobiliyati ham muhim ahamiyatga ega. Bitta katta hajmli diskdan ko'ra bir nechta kichikroq hajmli disklardan foydalanish tizimning barqarorligini oshirishga yordam beradi. Shuningdek, tranzaksiya ma'lumotlari va jurnallarini alohida disklarga joylashtirish tavsiya etiladi.[12]

- Uzluksiz energiya manbai. MySQL shunday amalga oshirilganki, istalgan vaqtda ma'lumotlarning butunligi kafolatlanadi, informasiya tranzaksiya jurnallari yordamida qayta tiklanadi. Biroq serverning o'zi to'satdan manba o'chib qolishi yoki kuchlanish sakrashidan himoyalanmagan. Har ikkala holda ham server qattiq talofat ko'rishi mumkin. Bunday hollardan saqlanish uchun uzluksiz energiya manбайдan foydalanish tavsiya etiladi.

Dasturiy ta'minot ishlashi uchun 2.4 GGs taktik chastotaga ega markaziy prosessor, 256 MB tezkor xotira, qattiq diskda 100 MB bo'sh joy, 100MB/s tezlikga ega tarmoq kartasi va kabeli, monitor, sichqoncha va klaviatura, Windows oilasiga mansub kamida Windows XP operatsion tizimi, MySQL dasturiy vositalari kerak bo'ladi.

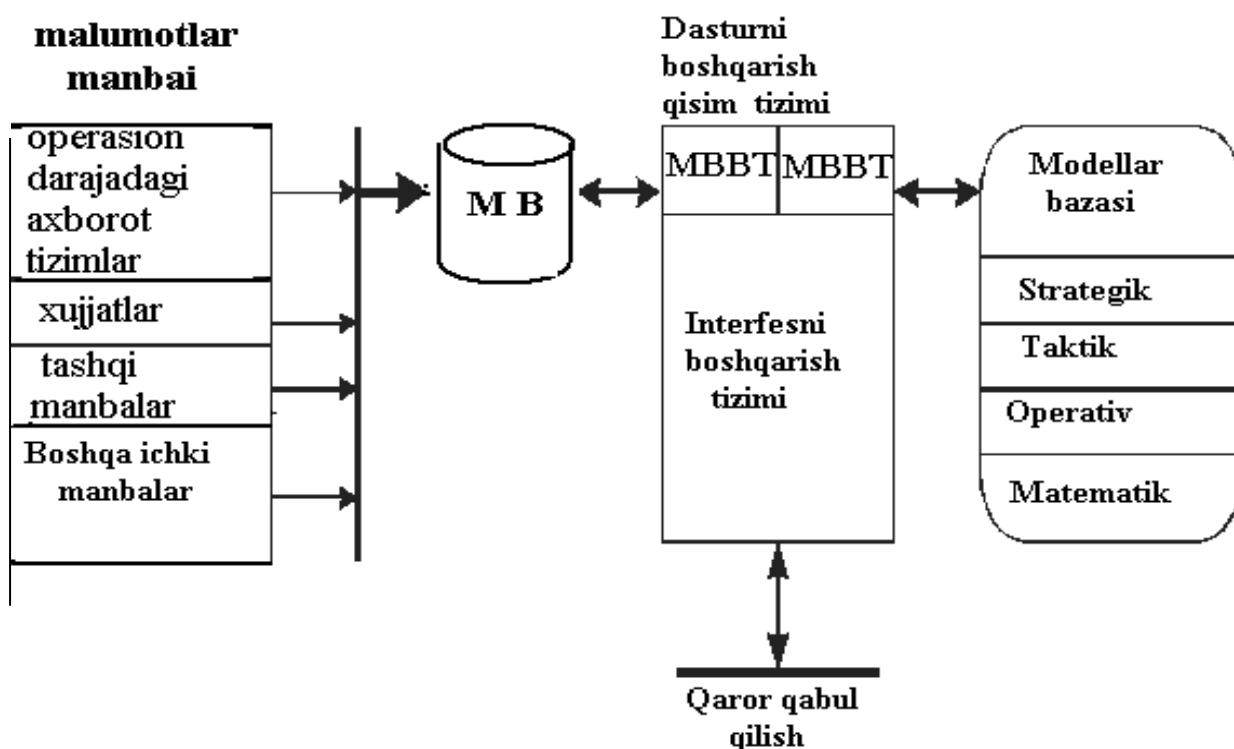
Asosiy talablar.

Yaratilayotgan sistemaning to'g'ri, tezkor va aniq ishlashi unga qo'yilgan talablar asosida ifodalanadi, bu talablar quyidagilardir:

- sistemaning ishlashi uchun Windows operatsion tizimini (LINUX operatsion tizimida ham ishlay oladi) bo'lishi tabiiydir;
- uning barcha funksiyalarini bajarilishi uchun PHP muhiti zarur;
- MySQL ma'lumotlar bazasini boshqarish tizimi bo'lishi kerak.

Barcha qo'yilgan talablarni bajarilishi, ishlab chiqilayotgan sistemani to'g'ri, tezkor, aniq va sifatli ishlashini kafolatlaydi.

3.2. Fuqarolar nafaqa ta'minot bo'limi tizimining strukturasi



3.2.1.-rasm. Avtomatlashtirilgan tizim strukturasi

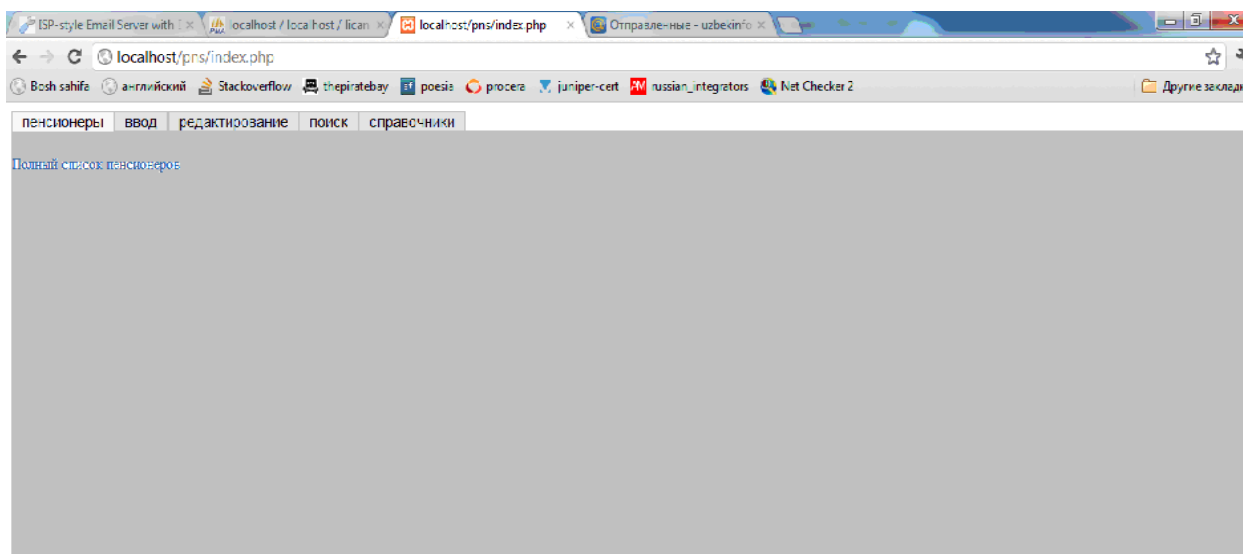
Avtomashtirilgan tizim strukturasi 3ta komponentlardan tashkil topgan:

- ma'lumotlar bazasi
- modellar bazasi
- modellar bazasini boshkarish tizimi (MBBT), ma'lumotlar bazasi boshqarish tizimi (MBBT) hamda foydalanuvchi va kompyuterni boshqaruvchi tizim interfeysidan tashkil topgan qism tizimi.

Interfeys display ekrani orqali kompyuterdagi ma'lumotlar tili hamda foydalanuvchi tili bilan bog'lab beruvchi vosita.

3.3. Foydalanuvchiga yo'riqnoma

Bu dastur nafaqa ta'minoti bo'limini ma'lumot bazasini yaratishga bag'ishlangan. Nafaqaxo'rlar haqida barcha ma'lumotlarni jamlaydi. Quyidagi ko'rinishdagi oyna Bosh sahifa ko'rinishi. Sahifani tepa qismida menyular joylashgan bo'lib: ma'lumotlarni kiritish, taxrirlash, qidirish, ular haqida ma'lumot olish mumkin (3.1 –rasm).



3.1.-rasm. Bosh sahifa ko'rinishi.

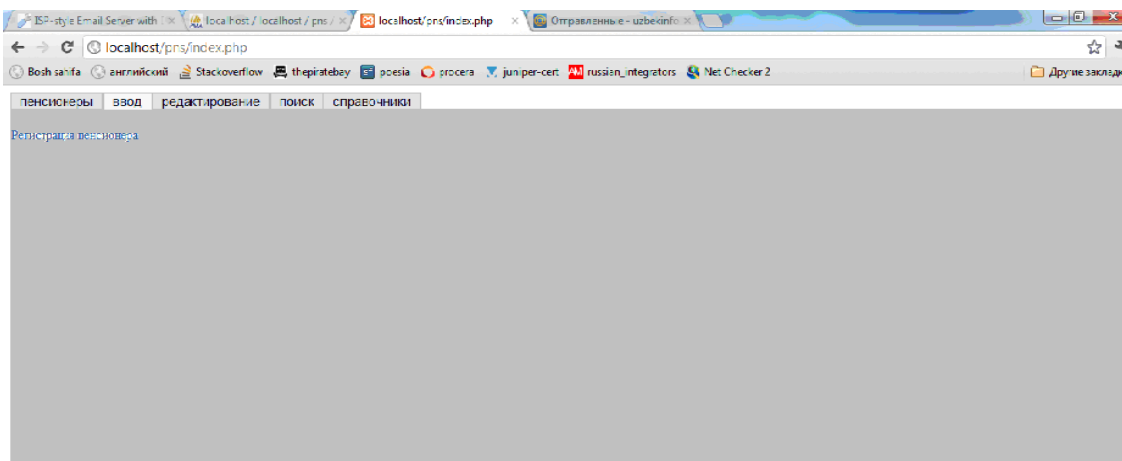
Bosh sahifadagi nafaqaxo'rlar menyusiga kirilsa ular haqida ma'lumot olish, ko'rish mumkin. (Ularning ismi sharifi, manzili, asosi, nafaqa miqdori v.k.)
Bu oynani ko'rinishi qudagicha (3.2 -rasm)



№	ФИО пенсионера	Адрес	Основание	Сумма пенсии
1	Axmedov Maxmud	Deggaron max 8	Yangi tayinlash	76324.3
2	Sunnatov Sattor	Deggaron max 8	Yangi tayinlash	76324.3
3	Qayumova Parda	Deggaron max 12	Yangi tayinlash	76324.3
4	Pozilova Ochil	Deggaron max 9	Yangi tayinlash	218820
5	Abdullaeva Roziya	Deggaron max 1	Yangi tayinlash	189936
6	Xotamova Dilbar	Deggaron max 10	07.07.11 dagi Farmon	549120
7	Yakubov Rashid	Deggaron max 10	Qayta tayinlanmasdan kiritilgan	0
8	Baxronova Sadokat	Deggaron max 1	07.07.11 dagi Farmon	549120
9	Xalimova Nazokat	1 gr.ChAES 34	Yangi tayinlash	116560
10	Murodova Ruzibibi	1 gr.ChAES 12	Yangi tayinlash	549120
11	Dullaeva Mexrimisa	1 gr.ChAES 57	Yangi tayinlash	401122
12	Yoldosheva Soxibyamal	1 gr.ChAES 97	Yangi tayinlash	258499
13	Shodiev Said	1 gr.ChAES 85	Yangi tayinlash	100132
...	Yo'dgorov	1	Yangi	...

3.2 –rasm. Nafaqaxo'rlar haqida ma'lumot oynasi.

Bu oynada yangi hujjat topshirgan nafaqaxo'rlarni registrasiya qilish oynasi (3.3- rasm). Vvod tugamasini bosilgandan so'ng quydagi oyna chiqadi:



3.3- rasm. Yangi hujjat topshirgan nafaqaxo'rlarni registrasiya qilish oynasi.

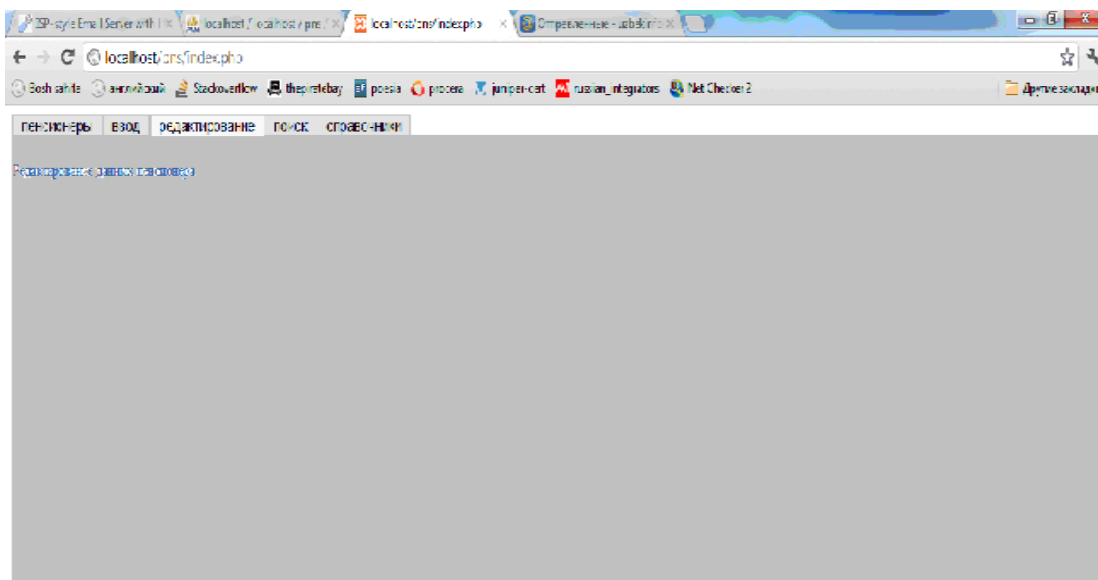
Yangi kirkandan keyin quyidagi oyna ochiladi va yangi kelgan nafaqaxo'r haqida har bir katakka unga tegishli ma'lumot to'latiladi. Bu oynani pastgi qismida kiritilgan ma'lumotni saqlash, tozalash, saxifadan chiqish tugmalari mavjud. (3.4 -rasm)

Ф.И.О.	адрес	серия паспорта	вид пенсии по старости
дата рождения	место выдачи пенсии	сумма пенсии	пол Ж
номер в ведомости №	дата регистрации 01.06.2012	время регистрации 12:12:04	

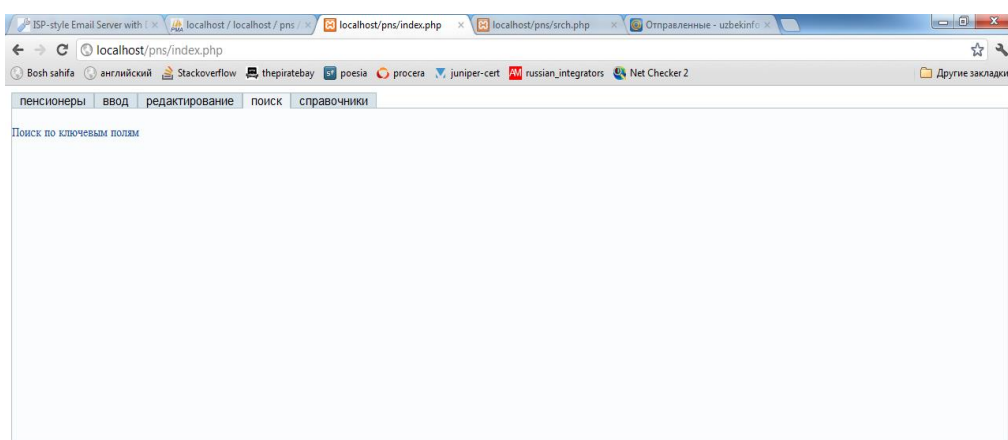
Сохранить Очистить Выйти

3.4 –rasm. Yangi hujjat topshirgan nafaqaxo'rlarni kiritish oynasi.

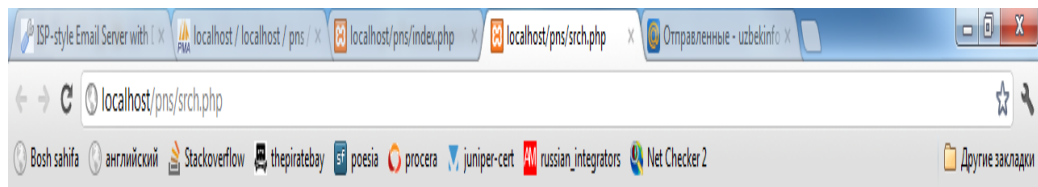
Bu oynada nafaqaxo'rlar haqida kiritilgan malumotlarni taxrirlash, tekshirish mumkin (3.5.-rasm).



Bu oyna yordamida kalitli so'zlar yordamida (Familiyasi, adresi, tug'lgan yili, pasport seriyasi, nafaqa nachisleniya qilingan vedomos bo'yicha) nafaqaxo'rlarni keng doirada qidiruv bajarish mumkin(3.6- rasm), (3.7 -rasm).

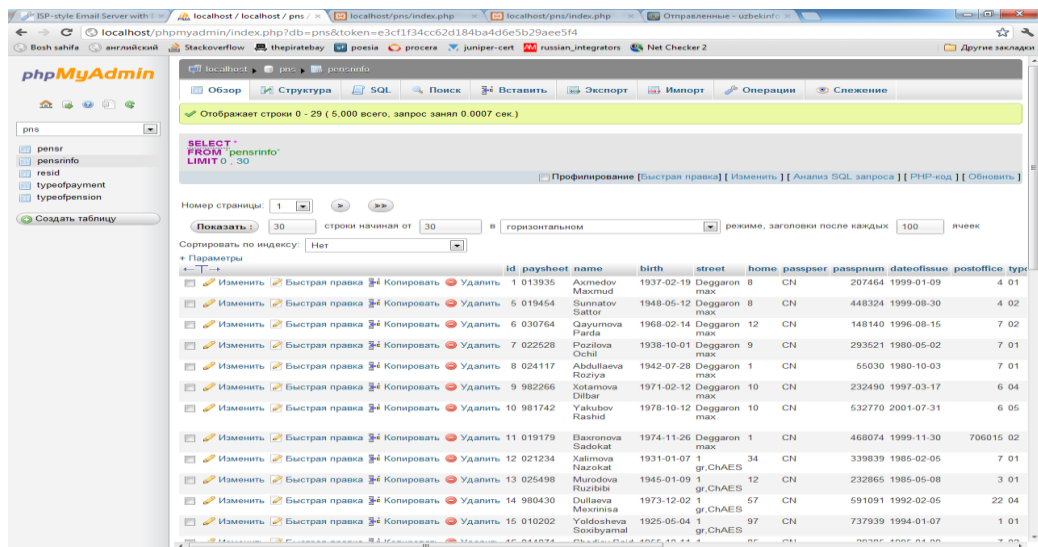


3.6- rasm. Nafaqaxo'rlarni turli atributlar bo'yicha qidiruv tashkil qilish oynasi.

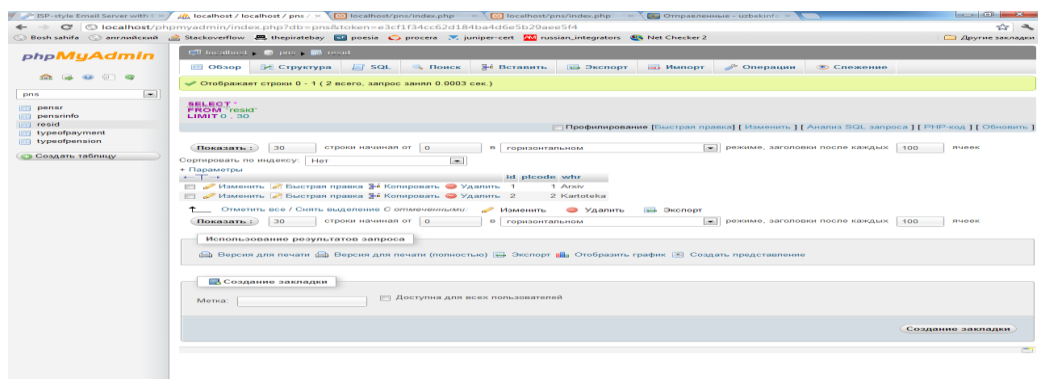


3.7- rasm. Nafaqaxo'rlarni keng doirada qidiruv tashkil qilish oynasi.

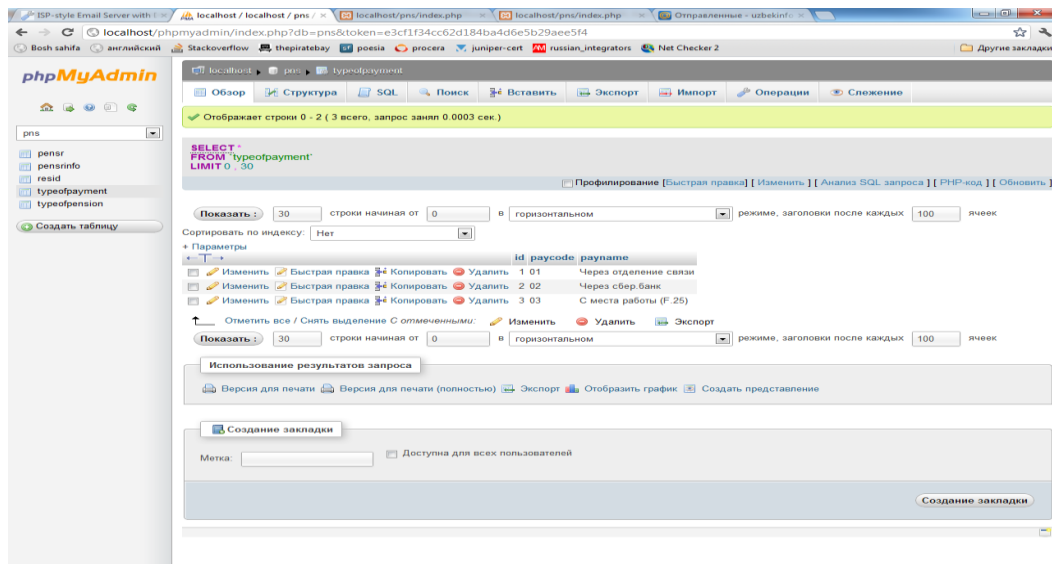
Dasturimizning ma'lumotlar bazalari ko'rinish oynalari



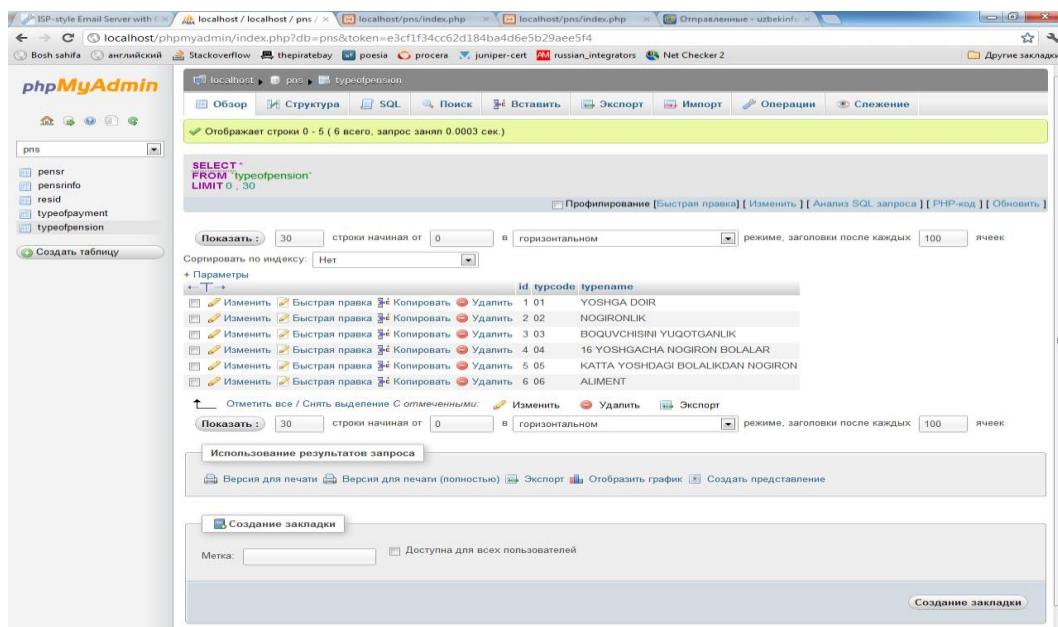
3.8-rasm. Pensinfo jadvali.



3.9.-rasm. Resid jadvali.



3.10.-rasm. Typeofpayment jadvali.



3.11.-rasm. Typeofpension jadvali.

XULOSA

Zamonaviy axborot asrida yashayotgan ekanmiz, hozirgi davrda zamon talabiga mos axborotlarni hayta ishlashga mo'ljallangan tizimlariga ehtiyoj seziladi. Zero, globallashtirilgan asrda, fan texnika rivojlanayotgan ayni bir paytda bunday tizimlarning yaratilishi rivojlanishning omili sifatida qaraladi. Shundan kelib chiqib, Web texnologiyasi asosida, PHP va MySQL dasturlaridan foydalanib yaratishni taqozo qiladi. Zamonaviy texnika vositalarga mos ravishda tezkor, qulay va sifatli axborot tizimlarini yaratish dolzarb vazifa hisoblanadi.

Bitiruv malakaviy ishining quydagi amaliy va nazariy natijalari olindi:

- Ish jarayonini avtomatlashtirishda masalaga to'g'ri yondoshuv orqali avtomatlashtirilgan tizimlarda ma'lumotlarni takshil etildi va Tizimini yaratishda foydalanilgan dasturiy vositalar belgilanib Ma'lumotlar bazasini yaratildi;
- Fuqarolarning nafaqa taminot bo'limining ma'lumotlar bazasi interfesini yaratish orqali hisoblash texnikasi buyicha maxsus tayyorgarlik ko'rmagan foydalanuvchiga ham mo'ljallangan foydalanuvchi uchun qulay bo'lgan interfeysga ega bo'ldik

Xulosa qilib aytish joizki, yuqorida keltirilgan fikrlarni tasdiqi shuki biz yaratgan dastur foydalanuvchi uchun juda ham qulay hamda ishlanishi onson bo'lib, shu bilan birga yuqoridagi fikrimizning dalilidir.

ADABIYOTLAR RO'YHATI

1. Karimov I.A. 2012 yil vatanimiz taraqqiyotini yangi bosqichga ko'tariladigan yil bo'ldi O'zbekiston Respublikasi I.A.Karimovning 2011 yilning asosiy yakunlari va 2012 yilda O'zbekistonning ijtimoiy iqtisodiy rivojlantirishning ustuvor yo'nalishlariga bag'ishlangan Vazirlar Maxkamasining majlisidagi ma'ruzasi G`G` Xalq so'zi, 2012 yil 20 yanvar, № 14 (5434).
2. "Jaxon moliyaviy-iqtisodiy inqirozi, O'zbekiston sharoitida uni bartaraf etish yo'llari va choralari " I.A.Karimov Toshkent-"O'zbekiston"-2009 y.
3. Bajenova I.Yu. Delphi7. Samouchitel programmitsa. – M.: «Kudits-obraz», 2009g.- 448s
4. " Burlak G.N. Bezopasnots rabota na kompyutere: Organizatsiya truda na predpriyatiyax informatsionnogo obslujivaniya./Uch.posobie. M.: Finansı i tsatitsika, 2007.-144 s.
5. Virt N. Algoritmi i tsrukturna dannix: Per. s angl. 2-e izd., ispr. – Spb.: Nevskiy dialekt, 2008. – 352 s.
6. Malumotlar bazalari. B.A. Begalov, E.E.Inogamjonov, Ssh. Nasriddinova. Toshkent – 2007y. – 235s.
7. Xodiev B.Yu., Begalov B.A., Abdullaev A., Pilipenko E.F., Ruzmetova X.N. Vvedenie v bazi dannix i znaniy. Tashkent, 2008. - 134 s.
8. Xodiev B.Yu., Musaliev A.A., Begalov B.A. Vvedenie v informatsionnie sitsemi i texnologii / Pod red. Akad. S.S. Gulyamova. Tashkent: TGEU-127s.
9. Xomonenko A.D., Sıgankov V.M., Maltsev M.G. Bazi dannix: Uchebnik dlya vısshix uchebnix zavedeniy / Pod red. Prof. A.D.Xomonenko. İzd. Vtor., pererab. Spb.: KORONAPrint, 2010. – 672 s.
10. Hayot faoliyati xavfsizligi fanidan ma'ruzalar matni. Q.Yo. Yormatov. Toshkent - 2009 y. -75s.
11. www.Google.com.uz
12. www.learnLisualStudio.com

Ilova

Исходный код:

index.php

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <link rel="stylesheet" type="text/css" href="lib/styles.css">
  <script type="text/javascript">
    function changeTab(sel){
      i=sel.cellIndex;
      num=sel.id.indexOf('_');
      ID=sel.id.substring(0,num);
      sel.className="tabPageSelected"
      tabbs=sel.parentNode;
      for (j=0;j<tabbs.cells.length;j++){
        if(i!=j && j!=tabbs.cells.length-1)
          tabbs.cells[j].className="tabPage";
      }
      //body
      tab_body=document.getElementById(ID);
      tab_body.cells[i].style.display="block";

      for(j=0;j<tab_body.cells.length;j++){
        if(i!=j)
          tab_body.cells[j].style.display="none";
      }
    }
    function selectDefaultTab(ID){
      obj=document.getElementById(ID);
      changeTab(obj);
    }
  </script>
</head>
<body>

<table border="0" class="tabControl" cellspacing="0" cellpadding="0" width="100%">
  <tbody>
    <tr>
      <td>
        <table border="0" cellspacing="0" cellpadding="0" class="tabHeader">
          <tbody>
            <tr>
              <td onclick="changeTab(this);" id="s3475394dfbgf_1"
class="tabPage">
                пенсионеры
              </td>
              <td onclick="changeTab(this);" id="s3475394dfbgf_2"
class="tabPage">
                ввод
              </td>
              <td onclick="changeTab(this);" id="s3475394dfbgf_3"
class="tabPage">
                редактирование
              </td>
              <td>
              </td>
            <tr>
              <td onclick="changeTab(this);" id="s3475394dfbgf_4"
class="tabPage">
                поиск
              </td>
              <td>
              </td>
            <tr>
              <td onclick="changeTab(this);" id="s3475394dfbgf_5"
class="tabPage">
                справочники
              </td>
              <td>
              </td>
            <tr>
              <td width="100%" class="tabHeaderLine"></td>
            </tr>
          </tbody>
        </table>
      </td>
    </tr>
  </tbody>
</table>
```

```

        <tr>
            <td height="400px" valign="top">
                <table width="100%" class="tabControlBody" cellpadding="0"
cellspacing="0" style="height:100%" border="0">
                    <tbody>
                        <tr id="s3475394dfbgf">
                            <td>
                                <br>
                                <a href=list.php target="_self">Полный список пенсионеров</a>
                                <br>
                            </td>
                            <td> <br>
                                <a href=reg.php target="_self">Регистрация пенсионера</a>
                            </td>
                            <td> <br>
                                <a href=# target="_blank">Редактирование данных пенсионера</a>
                            </td>
                        </tr>
                    </tbody>
                </table>
            </td>
            <td> <br>
                <a href="srch.php" target="_blank">Поиск по ключевым
полям</a><br />
            </td>
        </tr>
    </tbody>
</table>
</tr>
</tbody>
</table>
<script
type="text/javascript">document.onload=selectDefaultTab('s3475394dfbgf_1');</script>
</body>
</html>
list.php
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<link rel="stylesheet" type="text/css" href="lib/styles.css">
<?php
require('lib/config.inc');

@$id=$_GET['id'];
@$act=$_GET['do'];
@$rng=$_GET['rng'];
if (isset($rng)) {

} else {
    $rng = 1;
}
$n=100;

    $qr = "SELECT id FROM `pensrinfo`" ;    $result = mysql_query($qr) or die
(mysql_error());
    $m = mysql_num_rows($result);

    for ($i=1; $i<=bcdiv($m,$n,0); $i++) {
        echo "&nbsp;<a href=?rng=$i>".$i."</a>&nbsp;&nbsp;|";
        if (bcmmod($i, 40)==0 ) {
            echo "<br>";
        }
    }
    echo "<br><p align=center><input style=\"background-color:#BAD1E8; border-
style:groove\" type=button value=Назад onclick=\"document.location =
'index.php'\"></p>";
    echo "<table align=center border=0 cellpadding=0 cellspacing=0 id=mtbl>";

    $qr = " SELECT * FROM `pensrinfo` LIMIT ". $n*($rng-1).",".$n;
    $result = mysql_query($qr) or die (mysql_error());
    $i=0;
    $k = mysql_num_rows($result);

```



```

        echo "<tr bgcolor=#BAD1E8 align=center valign=center>";
        echo "<td><img src=img/blank.gif width=10 height=20></td>";
echo "<td><font color=#00008B>№&nbsp;</font></td>";
        echo "<td><b>ФИО пенсионера</b></td>";
        echo "<td><b>Адрес</b></td>";
echo "<td><b>Основание</b></td>";
echo "<td><b>Сумма пенсии</b></td>";
        echo "</tr>";
echo "<tr valign=bottom>
      <td bgcolor=#809CD0 colspan=8><img src=img/blank.gif width=1
height=4></td>
      </tr>";

while( $row=mysql_fetch_array($result) )
{
    $i++;
        echo "<tr valign=bottom>
      <td bgcolor=#ffffff background='img/strichel.gif'
colspan=7><img src=img/blank.gif width=1 height=1></td>
      </tr>";
        echo "<tr valign=center>";
        echo "<td><img src=img/blank.gif width=10 height=20></td>";
echo "<td><font color=#00008B>". $i. "&nbsp;</font></td>";
        echo "<td><b>". $row['name']. "</b></td>";
        echo "<td>". $row['street']. "&nbsp;<td>". $row['home']. "</td>";
        echo "<td>". $row['cause']. "&nbsp;<td>";
        echo "<td>". $row['salaryaccounting']. "&nbsp;</td>";

        echo "</tr>";
    }

    echo "<tr valign=bottom>";
    echo "<td bgcolor=#809CD0 colspan=10><img src=img/blank.gif width=1
height=4></td>";
    echo "</tr>";
?>
</table>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Регистрация контейнеров</title>
</head>
<?php
date_default_timezone_set('Asia/Tashkent') ;
$today = date("Y-m-d");
$today = date("H:i:s") ;
?>

<style>
A:link {
    FONT-SIZE: 12px; COLOR: #235094; FONT-FAMILY: tahoma, sans-serif; TEXT-
DECORATION: none
}

.tit {
    font-size: 18px;
    color: #903;
}

table {
    width:100mm;
    margin:1em auto;
    background-color:#f3f5f8;
    cellpadding:0;
    cellspacing:2;

```

```

border-collapse:collapse;
border-bottom:1px solid #000080;
border-left:1px solid #000080;
border-top:1px solid #000080;
border-right:1px solid #000080;
}

td {
    color:#000080;
    padding:.2em 1em;
    font-family: "Times New Roman", Times, serif;
    font-size: 12pt;
}

th {
    color:#000080;
    padding:.2em 1em;
    font-family: "Times New Roman", Times, serif;
}

</style>
<body>
<script type="text/javascript" src="lib/jquery.js"></script>
<script type="text/javascript" src="lib/jquery.autotab.js"></script>

<script type="text/javascript">
$(document).ready(function() {
    $('#area_code, #contnum, #numfdu').autotab_magic().autotab_filter('numeric');
});
</script>
</body>
</html>
<?php
date_default_timezone_set('Asia/Tashkent') ;
include_once("lib/config.inc");

    mysql_query("SET NAMES utf8");
    $result=mysql_query("SELECT regcont.id,regcont.clientid, regcont.contnum,
regcont.typcont, client.name FROM `regcont`, `client` where
regcont.clientid=client.id");
    echo "<table>
    <tr>
        <td align=\"center\"><span class=\"tit\">Выберите клиента для ввода памятки на
выдачу</span></td>
    </tr>";
    $i=0;
        while( $row=mysql_fetch_array($result) )
        {
            echo "<form method=\"post\" enctype=\"multipart/form-data\" id=\"regmem\"
action=\"memcalc.php\" target=\"_self\">";
            echo "<input type=hidden name=clid value=\".$row['clientid'].\">
                <input type=hidden name=contid value=\".$row['id'].\">
                <input type=hidden name=contnum value=\".$row['contnum'].\">
                <input type=hidden name=clname value=\"\".$row['name'].\">
                <input type=hidden name=typcont value=\"\".$row['typcont'].\">
            </form>";
            echo "<tr><td>&nbsp;<a
href=\"javascript:document.forms[\".$i.\"].submit()\">\".$row['name'].\"</a></td> </tr>";
                @$i++;
            }
        //<a href="javascript:document.forms[0].submit()">
        // id clientid name
        echo "<tr>
        <td align=\"center\">
        &nbsp;<input style=\"background-color:#f3f5f8; border-style:groove\" type=\"button\"
value=\"Выйти\" onclick=\"document.location = 'index.php'\" />
        </td></tr></table>";
        ?>

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

```

```

<title>Регистрация контейнеров</title>
<script type="text/javascript" src="lib/jquery.js"></script>
<script src="lib/jscal2.js"></script>
<script src="lib/ru.js"></script>
<link rel="stylesheet" type="text/css" href="lib/jscal2.css" />
<!--link rel="stylesheet" type="text/css" href="lib/border-radius.css" /-->
<!--link rel="stylesheet" type="text/css" href="lib/styles.css" /-->

<script language="JavaScript" type="text/javascript">
function Interval() {
    var cd = document.forms[0].theDate1.value;
    var dt1 = new Date(Date.UTC(cd.substr(6, 4), cd.substr(3, 2), cd.substr(0,2),
cd.substr(11, 2), cd.substr(14, 2)));
    var ch = "Date1 [" + cd + "] = " + dt1;
    cd = document.forms[0].theDate2.value;
    var dt2 = new Date(Date.UTC(cd.substr(6, 4), cd.substr(3, 2), cd.substr(0,2),
cd.substr(11, 2), cd.substr(14, 2)));
    ch += "<br>Date2 [" + cd + "] = " + dt2;
    var fms = Math.abs(dt1.getTime() - dt2.getTime());
    document.forms[0].timdiff.value = Math.round(fms / 3600000);
//    document.getElementById("view").innerHTML = ch + "<br>Interval: " + fms + "
millisecound";
}

$(function(){
    $("table input:checkbox").click(function(){
        if ( $(this).is(":checked") ){
            $(this).parents("tbody").find("input:radio").attr("disabled",null);
        }
        else{
            $(this).parents("tbody").find("input:radio").attr("disabled",true);
        }
    });
});

/*<![CDATA[*/
/*]]>*/
</script>

</head>
<style>

.tit {
    font-size: 18px;
    color: #903;
}

</style>

<body>
<!--body onLoad="InitForm();" onreset="InitForm();"-->
<?php
@$cname    = $_POST['cname'];
@$tpcnt    = $_POST['typcnt'];
@$contid   = $_POST['contid'];
@$contnum  = $_POST['contnum'];

date_default_timezone_set('Asia/Tashkent') ;
//$tod = date("d.m.Y H:i");
//$tom = date("d.m.Y H:i",mktime(date("H"), date("i"), 0, date("m") , date("d")+1,
date("Y")));
//echo $tom->format('Y-m-d') . "\n";
//int mktime ( [ int $hour = date("H") [ , int $minute = date("i") [ , int $second =
date("s") [ , int $month = date("n") [ , int $day = date("j") [ , int $year = date("Y")
[ , int $is_dst = -1 ]]]]] )

include_once("lib/config.inc");
mysql_query("SET NAMES utf8");
echo "<form method=POST name=memform><table align=center>

```

```

        <tr><td colspan=3 align=\"center\"><span
class=\"tit\">Расчет&nbsp;памятки&nbsp;для&nbsp;.</b>&nbsp;с&nbsp;контейн
ером:<b>&nbsp;№$contnum&nbsp;$typcont&nbsp;фут</b>&nbsp;на&nbsp;выдачу</span></td></tr>
?>
    <tr>
        <td align=center>Начало:<br><input size=14 readonly name=theDate1 id="begDat"
onblur="Interval()" /><button id="begBut">...</button></td>
        <td align=center>Конец:<br><input size=14 readonly name=theDate2 id="endDat"
onblur="Interval()" /><button id="endBut">...</button></td>
        <td>Время:<input type=text name=timdiff readonly size=2 maxlength=3
onclick="Interval()"></td>
    </tr>
<?php
$result=mysql_query("SELECT * FROM `services` WHERE (`show`=
1)AND(`contorvag`!=2)AND(`typcont`!=$tpcnt)");
while( $row=mysql_fetch_array($result) )
{
    $pid=$row['id'];
    // echo " <tr><td><input type=checkbox name=servname
value=\".$row['id'].\"><a>\".$row['servname'].\"</a></td><td><a>Сумма:&nbsp;\".$row['servva
l'].\"</a></td><td><a>НДС:&nbsp;\",0.2*$row['servval'].\"</a></td></tr>";

    if ($row['child']==1)
    {
        echo " <tbody><tr><td colspan=3><input type=radio name=servch\".$row['id'].\"
value=0><b>\".$row['servname'].\"</b></td></tr>";
        $res=mysql_query("SELECT * FROM `servchild` WHERE (`parentid` =
$pid)AND(`typcont`!=$tpcnt)");
        while( $rw=mysql_fetch_array($res) )
        {
            echo "<tr><td><input type=radio name=servch\".$rw['parentid'].\"
value=\".$rw['servval'].\"><a>\".$rw['servname'].\"</a></td>";
            echo
            "<td><a>Сумма:&nbsp;\".$rw['servval'].\"&nbsp;НДС:&nbsp;\",0.2*$rw['servval'].\"</a></td><
td>&nbsp;\"</td></tr>";
        }
        echo "</tbody>";
    }
    else {
        echo " <tr><td><input type=checkbox name=serv\".$row['id'].\" value=\".$row['servval'].\"
><a>\".$row['servname'].\"</a></td>

        <td><a>Сумма:&nbsp;\".$row['servval'].\"&nbsp;НДС:&nbsp;\",0.2*$row['servval'].\"</a></td>
        <td>&nbsp;\"</td></tr>";
    }
    // echo "<tr><td>&nbsp;<a
href=\"javascript:document.forms[\".$i.\"].submit()\">\".$row['name'].\"</a></td> </tr>";
}
$result=mysql_query("SELECT * FROM `coeff` WHERE `show` = 1"); // id NAME VALUE
while( $row=mysql_fetch_array($result) )
{
    echo " <tr><td><a>\".$row['name'].\"&nbsp;\"</a><input type=text maxlength=3 size=2
name=coef\".$row['id'].\"
value=\".$row['coef'].\"><a><td>&nbsp;Стоим. (ед) :&nbsp;\".$row['price'].\"&nbsp;НДС:&nbsp;
\",0.2*$row['price'].\"</a></td>
        <td><input type=hidden name=val\".$row['id'].\" value=\".$row['price'].\"</td></tr>";
}
//<a href="javascript:document.forms[0].submit()">

    $rslt=mysql_query("SELECT * FROM `penalty` WHERE `contorvag`!=2");
    while( $row=mysql_fetch_array($rslt) )
    {
        $pid=$row['id'];
        // echo " <tr><td><input type=checkbox name=servname
value=\".$row['id'].\"><a>\".$row['servname'].\"</a></td><td><a>Сумма:&nbsp;\".$row['servva
l'].\"</a></td><td><a>НДС:&nbsp;\",0.2*$row['servval'].\"</a></td></tr>";

        if ($row['child']==1)

```

```

{
    echo " <tbody><tr><td colspan=3><input type=radio name=pench".$row['id']."
    value=0><b>".$row['penname']."</b></td></tr>";
    $res=mysql_query("SELECT * FROM `penchild` WHERE (`parentid` =
    $pid)AND(`typcont`!= $tpcnt)");
    while( $rw=mysql_fetch_array($res) )
    {
        echo "<tr><td><input type=radio name=pench".$rw['parentid']."
        value=".$rw['penval']."'><a>".$rw['chname']."'</a></td>";
        echo
        "<td><a>Cymma:&nbsp;".$rw['penval']."'&nbsp;HДC:&nbsp;",$0.2*$rw['penval']."'</a></td><td
        >&nbsp;</td></tr>";
    }
    echo "</tbody>";
}
else {
    echo " <tr><td><input type=checkbox name=pen".$row['id']." value=".$row['penval']."'
    ><a>".$row['penname']."'</a></td>

    <td><a>Cymma:&nbsp;".$row['penval']."'&nbsp;HДC:&nbsp;",$0.2*$row['penval']."'</a></td><t
    d>&nbsp;</td></tr>";
}
// echo "<tr><td>&nbsp;<a
href=\"javascript:document.forms[\".$i.\"].submit()\">".$row['name']."'</a></td> </tr>";
}

echo "<tr><td colspan=3 align=\"center\"><a>Cymma:</a>&nbsp;<input style=\"background-
color:#f3f5f8; border-style:groove\" name=totalSum id=totalSum maxlength=8 size=6
type=\"text\" value=\"\" readonly onchange=\"nds()\">&nbsp;
<a>HДC:</a>&nbsp;<input style=\"background-color:#f3f5f8; border-style:groove\"
name=nds id=nds maxlength=8 size=6 type=\"text\" value=\"\" readonly>
<a>Иторо:</a>&nbsp;<input style=\"background-color:#f3f5f8; border-style:groove\"
name=allsum id=allsum maxlength=8 size=6 type=\"text\" value=\"\" readonly>
<input type=hidden name=actn value=\"\"></td></tr>";
echo "<tr>
<td colspan=3 align=\"center\">
&nbsp;<input style=\"background-color:#f3f5f8; border-style:groove\" type=\"submit\"
value=\"Сохранить\" onclick=\"document.forms[0].actn.value='save'\" />
&nbsp;<input style=\"background-color:#f3f5f8; border-style:groove\" type=\"button\"
value=\"Выйти\" onclick=\"document.location = 'mem.php'\" /></td></tr></table>";
print_r($_POST);
?>
</form>
</body>

<script type="text/javascript">
    var cal = Calendar.setup({
        onSelect: function(cal) { cal.hide() },
        showTime: true
    });
    cal.manageFields("begBut", "begDat", "%d.%m.%Y %H:%M");
    cal.manageFields("endBut", "endDat", "%d.%m.%Y %H:%M");

    $("table input:radio, table input:checkbox").click(function(){
        var total=0;
        total =
        parseInt($("input[name='val1']").val())*parseInt($("input[name='coef1']").val());
        total +=
        parseInt($("input[name='val2']").val())*parseInt($("input[name='coef2']").val());

        $('table input:checked:enabled').each(function(index) {
            total += parseInt($(this).val());
        });
        $("input[name='totalSum']").val(total)
        $("input[name='nds']").val(Math.round(0.2*total))
        $("input[name='allsum']").val(Math.round(total+0.2*total))
    });
</script>
</html>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

```



```

while( $row=mysql_fetch_array($result) )
{
    $i++;
    echo "<tr valign=bottom>
        <td bgcolor=#ffffff background='img/strichel.gif'
colspan=7><img src=img/blank.gif width=1 height=1></td>
        </tr>";
    echo "<tr valign=center>";
    echo "<td><img src=img/blank.gif width=10 height=20></td>";
    echo "<td><font color=#00008B>". $i. "&nbsp;</font></td>";
    echo "<td><b>". $row['servname']. "</b></td>";
    echo "<td>". $row['servval']. "&nbsp;</td>";
    echo "<td title=\"Редактировать\"><a
href=?do=ed&id=\". $row['id']. "><img src=img/edit.gif border=no width=16
height=16></a></td>";
    echo "<td>&nbsp;&nbsp;</td>";
    echo "<td title=\"Удалить\"><a href=?do=del&id=\". $row['id']. "><img
src=img/del.gif border=no width=16 height=16></a></td>";
    echo "<td></td>";
    echo "</tr>";
}
echo "<tr valign=bottom>";
echo "<td bgcolor=#809CD0 colspan=10><img src=img/blank.gif width=1
height=4></td>";
echo "</tr>";
?>
</table>
</body>
</html>

<?php
/**
 * Сумма прописью
 * @author runcore
 */
function num2str($inn, $stripkop=false) {
    $nol = 'ноль';
    $str[100]= array('','сто','двести','триста','четыреста','пятьсот','шестьсот',
'семьсот', 'восемьсот','девятьсот');
    $str[11] = array('','десять','одиннадцать','двенадцать','тринадцать',
'четырнадцать','пятнадцать','шестнадцать','семнадцать',
'восемнадцать','девятнадцать','двадцать');
    $str[10] = array('','десять','двадцать','тридцать','сорок','пятьдесят',
'шестьдесят','семьдесят','восемьдесят','девяносто');
    $sex = array(
        array('','один','два','три','четыре','пять','шесть','семь',
'восемь','девять'), // m
        array('','одна','две','три','четыре','пять','шесть','семь',
'восемь','девять') // f
    );
    $forms = array(
        array('тийин', 'тийин', 'тийин', 1), // 10^-2
        array('рубль', 'рубля', 'рублей', 0), // 10^ 0
        array('тысяча', 'тысячи', 'тысяч', 1), // 10^ 3
        array('миллион', 'миллиона', 'миллионов', 0), // 10^ 6
        array('миллиард', 'миллиарда', 'миллиардов', 0), // 10^ 9
        array('триллион', 'триллиона', 'триллионов', 0), // 10^12
    );
    $out = $tmp = array();
    // Поехали!
    $tmp = explode('.', str_replace(',','.', $inn));
    $rub = number_format($tmp[0],0,'','');
    if ($rub==0) $out[] = $nol;
    // нормализация копеек
    $kop = isset($tmp[1]) ? substr(str_pad($tmp[1], 2, '0', STR_PAD_RIGHT),0,2) :
'00';

    $segments = explode('-', $rub);
    $offset = sizeof($segments);
    if ((int)$rub==0) { // если 0 рублей
        $o[] = $nol;
        $o[] = morph(0, $forms[1][0],$forms[1][1],$forms[1][2]);
    }
}

```

```

    }
    else {
        foreach ($segments as $k=>$lev) {
            $sexi= (int) $forms[$offset][3]; // определяем род
            $ri = (int) $lev; // текущий сегмент
            if ($ri==0 && $offset>1) { // если сегмент==0 & не последний
уровень (там Units)
                $offset--;
                continue;
            }
            // нормализация
            $ri = str_pad($ri, 3, '0', STR_PAD_LEFT);
            // получаем циферки для анализа
            $r1 = (int) substr($ri,0,1); // первая цифра
            $r2 = (int) substr($ri,1,1); // вторая
            $r3 = (int) substr($ri,2,1); // третья
            $r22= (int) $r2.$r3; // вторая и третья
            // разгребаем порядки
            if ($ri>99) $o[] = $str[100][$r1]; // Сотни
            if ($r22>20) { // >20
                $o[] = $str[10][$r2];
                $o[] = $sex[ $sexi ][$r3];
            }
            else { // <=20
                if ($r22>9) $o[] = $str[11][$r22-9]; // 10-20
                elseif($r22>0) $o[] = $sex[ $sexi ][$r3]; // 1-9
            }
            // Рубли
            $o[] = morph($ri,
$forms[$offset][0],$forms[$offset][1],$forms[$offset][2]);
            $offset--;
        }
    }
    // Копейки
    if (!$stripkop) {
        $o[] = $kop;
        $o[] = morph($kop,$forms[0][0],$forms[0][1],$forms[0][2]);
    }
    return preg_replace("/\s{2,}/"," ",implode(' ', $o));
}

/**
 * Склоняем словоформу
 */
function morph($n, $f1, $f2, $f5) {
    $n = abs($n) % 100;
    $n1= $n % 10;
    if ($n>10 && $n<20) return $f5;
    if ($n1>1 && $n1<5) return $f2;
    if ($n1==1) return $f1;
    return $f5;
}

echo num2str(878867.15);
?>

```