

**O'ZBEKISTON RESPUBLIKASI OLIY VA O'RTA TA'LIM
VAZIRLIGI**

BUXORO DAVLAT UNIVERSITETI

Fizika-matematika fakulteti

“Axborot texnologiyalari” kafedrası

Fayziyeva O'lmasoy Istam qizining

**“UNIVERSITET INFORMATSION TIZIMIDA IQTIDORLI TALABALAR
BILAN ISHLASH MONITORINGI MODULINI YARATISH”**

“Amaliy matematika va informatika” ta'lim yo'nalishi bo'yicha bakalavr
darajasini olish uchun

BITIRUV MALAKAVIY ISHI

“Ish ko'rildi va himoyaga ruxsat
berildi”

Kafedra mudiri

_____ dots.T.B.Boltayev

«___» _____ 2016 y.

Ilmiy rahbar _____ **I.O.Jalolov**

«___» _____ 2016 y.

Taqrizchi _____

«___» _____ 2016 y.

«Himoya qilishga ruxsat berildi»

Fakultet dekani _____ prof.Sh.M.Mirzayev

«___» _____ 2016 y.

MUNDARIJA

Kirish	3
I BOB.Foydalanilgan dasturiy ta'minot va texnologiyalar.	
1.1.SQL server 2008 dasturi va uning imkoniyatlari	6
1.2.SQL tili	12
1.3.HTML tili va PHP dasturlash tili	18
1.4.Universitet informatsion tizimining funksional imkoniyatlari	28
II BOB. Universitet informatsion tizimida iqtidorli talabalar bilan ishlash monitoringi modulini yaratish	
2.1.Modul spetsifikatsiyasi	33
2.2.ER modeli loyihasi	48
2.3.Foydalanuvchi grafik interfeysi	55
2.4.Qo'yilgan masalaning dasturini amalga oshirish	60
Xotima	62
Adabiyotlar	63

Yuksak marralarni ko'zlagan
har qanday millat, xalq uchun
ta'limni rivojlantirish hayot va
mamot masalasidir.

I.A.Karimov.

Kirish

O'zbekiston Respublikasi mustaqillik odimlarini dadil qo'yayotgan hozirgi davrda axborotlashgan jamiyat qurish masalasi mamlakatimiz uchun naqadar katta ahamiyat kasb etayotgani hech kimga sir emas. Internet hayotimizning bir bo'lagiga aylandi, biz uning xizmatlaridan har kuni foydalanishga odatlandik. Respublikamizda o'qitish texnologiyalarini zamonaviylashtirishni jadllashtirish rivojlangan iqtisodiyotli mamlakatlarga qaraganda yanada dolzarb ahamiyatga ega. Chunki hozirgi kunda milliy ta'lim tizimining salohiyati iqtisodiy rivojlanishning yanada yuqori pog'onasiga ko'tarilishga amaliy imkoniyat ta'minlovchi asosiy ijtimoiy resurs sifatida gavdalanadi.

Respublikamiz ta'lim tizimidagi asosiy vazifa jahon talablariga mos keluvchi axborot texnologiyalarini o'qitish jarayoniga qo'llashdan iborat. O'zbekistonda ta'lim tizimining axborotlashtirilishi xalqaro hamjamiyatda ham tan olindi. Masofaviy ta'limni rivojlantirish bo'yicha bir qator dasturlar ishlab chiqilmoqda. Iqtisodiyot va jamiyatda islohotlarning o'tkazilishi natijasida o'quv jarayonining zahira hajmini keskin oshirish bo'yicha yangi talablar qo'yildi. Hozirgi kunda axborot eng asosiy ishlab chiqaruvchi resurslardan biriga, iqtisodiyot va umuman jamiyatning rivojlanish poydevoriga aylanmoqda.

Yangi axborot - kommunikatsion texnologiyalar hozirgi kunda eng dolzarb mavzulardan biri bo'lib kelmoqda, sababi har bir sohani o'rganish, izlanish va tajriba orttirish uchun turli usullardan foydalanish kerak bo'ladi. Shuning uchun yangi axborot kommunikatsion texnologiyalardan foydalanish maqsadga muvofiqdir.

Respublikamizda olib borilayotgan islohotlarning tarkibida yuqori malakali mutaxasislarning roli benihoya kattadir. Prezidentimiz ta'kidlaganidek, "Ertangi

kun yangicha fikrlay oladigan zamonaviy bilimga ega bo'lgan yuksak malakali mutaxasislarni talab etadi". Shu sababli xalqimizning boy intellektual merosi va umumbashariy qadriyatlari, zamonaviy madaniyat, iqtisodiyot, fan, texnika va texnologiyalar asosida yuksak mutaxasislar tayyorlash tizimi ishlab chiqildi va jadal sur'atlar bilan hayotga tadbiiq etilmoqda.

O'zbekistonda o'qitish texnologiyalarini zamonaviylashtirish, jadallashtirish rivojlangan iqtisodiyotli mamlakatlarga qaraganda yanada dolzarb ahamiyatga ega. Chunki hozirgi kunda milliy ta'lim tizimining salohiyatli tizimli rivojlanishi yanada yuqori pog'onaga ko'tarildi.

1.Mavzuning dolzarbligi: Universitet informatsion tizimida iqtidorli talabalar bilan ishlash monitoringi modulini yaratish

2.Bitiruv malakaviy ishimizning maqsadi: Iqtidorli talabalar bilan ishlashni qo'llab-quvvatlash, vaqtni tejash va kerakli hisobotlarni ortiqcha qiyinchiliklar va vaqt sarflamasdan olish. Iqtidorli talabalar ilmiy faoliyatini nazorat qilib borish.

3.Bitiruv malakaviy ishimizning vazifasi: Universitetdagi mavjud tizim bilan bog'langan holda iqtidorli talabalarning ilmiy faoliyatining monitoringini olib borish.

4.Bitiruv malakaviy ishimizning o'rganganlik darajasi: Oliy ta'lim muassasalari uchun iqtidorli talabalar faoliyatini olib borish va nazorat qilish hamda birqancha tez zarur bo'lgan hisobotlarni tayyorlash muammolari amaliy hamda nazariy jihatdan o'rganildi.

5.Bitiruv malakaviy ishimizning predmeti: SQL server 2008 berilganlar bazasini boshqarish tizimi, universitetda mavjud informatsion tizim, HTML tili, php Designer 7 dasturi, php dasturlash tili.

6.Bitiruv malakaviy ishimizning obykti: SQL server 2008 berilganlar bazasini boshqarish tizimi, universitetda mavjud informatsion tizim, HTML tili, php Designer 7 dasturi, php dasturlash tili.

7.Bitiruv malakaviy ishimizning ilmiy farazi: Iqtidorli talabalar ishlarini nazorat qilish, o'z ustida ishlashni kuchaytirish, faol talabalarni rag'batlantirish, olimpiada va tanlovlarda qatnashishni ta'minlash.

8.Bitiruv malakaviy ishimizning yangiligi: Yaratilgan ER model, foydalanuvchi interfeysi dizayni va qayta ishlash algoritmi.

9.Bitiruv malakaviy ishimizning amaliy ahamiyati: Oliy ta'lim muassasalarida iqtidorli talabalar faoliyatini nazorat qilish uchun va kerakli hisobotlarni tezkor amalga oshirish uchun qulay tizim yaratish.

10.Bitiruv malakaviy ishimizning metodologik asosi: Bitiruv malakaviy ishi dasturiy mahsulot tayyorlash metodologiyasiga asoslanib quyidagi etaplarda amalga oshiriladi: tadbqiqiy sohani tizimli tahlil qilish ,tizimli web loyihalash va tayyorlangan web loyihani amalga oshirish .

11.Bitiruv malakaviy ishimizning metodlari: Web arxitekturaga asoslangan MS SQL ma'lumotlar bazasida saqlanib, axborot xavfsizligi, ma'lumotlarning butunligi va tranzaksion funksiyalar ushbu ma'lumotlar bazasini boshqarish tizimi metodlariga asoslanadi.

12.Bitiruv malakaviy ishimizning tarkibi va hajmi: Kirish, 2 ta bob, har bir bobning qisqacha xulosasi, adabiyotlar ro'yxati va xotimadan iborat bo'lib, jami 64 betda bayon qilingan.

Universitetning mavjud informatsion tizimini o'rganish, Universitetning ER modeli SQL server berilganlar bazasiga joylashtirish. SQLga joylashtirilgan mavjud informatsion tizim bazasini, bazada joylashgan jadvallarni o'rganish. Universitet informatsion tizimining funksiyalarini o'rganish. Bu tizimda talabalar haqida to'liq ma'lumot berilgan. Talaba bemalol internetga bog'langan bu tizimdan o'zlarining o'qituvchilar tomonidan olgan ballarini ko'rishlari mumkin.

Yaratilgan ma'lumotlar bazamiz bo'sh, ya'ni hech qanday jadvallari yo'q. Shuning uchun keyingi vazifamiz jadvallarni yaratish bo'ladi. Jadvallarning tuzilishi xuddi Access da tuzilgan jadvallar kabi bo'ladi.[3]

Jadvallarni yaratish uchun <<Tables>> novdasining kontekstli menyusidan <<New Table>> ni tanlaymiz.

Jadvallar orasidagi bog'lamni aniqlash uchun birlamchi kalitni berish zarur. Buning uchun zarur bo'lgan maydonning kontekstli menyusidan <<Set primary key>> ni tanlash kerak.

Jadvallar orasidagi bog'lamni va ma'lumotlar sxemasini yaratish uchun yangi ma'lumotlar bazasining diagrammasini yaratish zarur. Buning uchun <<Database Diagrams>> novdasining kontekstli menyusidan kerakli punktni tanlaymiz. Hosil bo'lgan oynada, diagrammaga kerakli bo'lgan jadvallarni qo'shamiz.

Management Studio muhitida so'rov yaratish quyidagicha amalga oshiriladi. <<Views>> novdasining kontekstli menyusidan <<New view>> punktni tanlaymiz. Shundan so'ng kerakli ustun va jadvallarni tanlaymiz.

1.1.2 Ma'lumotlar bazasini boshqarish tizimlari.

Ma'lumotlar bazasini loyihalashtirish jarayoni ikki bosqichga bo'linadi: MB mantiqiy tuzilishini tashkil qilish va tashuvchilarda MB ni hosil qilish.

MB ning mantiqiy tuzilishi-obyektga tegishli bo'lgan axborotlarning MB da joylanishini ifodalaydi. Hosil bo'lgan MB ning mantiqiy bog'lanish modeli birinchi bosqichning natijasi hisoblanadi. Bu modelda uch turdagi axborot ifodalanadi: obyekt to'g'risidagi xabarlar, ularning xususiyati va o'zaro munosabatlari. Har bir obyekt modelda yozuv turlari orqali ko'rsatiladi. Ularning xususiyatlari yozuv maydonlari orqali ifodalanadi, munosabatlar esa – yozuv va maydon turlari o'rtasidagi aloqalar yordamida tasvirlanadi. Bunday model EHM, operatsion tizim, MBBTning mohiyatiga bog'liq bo'lmaydi, ya'ni axborotning ma'nosiga bog'liq bo'lmagan holda ularni ifodalash usuli va aloqasini ta'minlaydi.

Mantiqiy modelni chizmalı va jadvalli usullar yordamida ifodalash mkin. Chizmalı usullarda ma'lumotlar o'rtasidagi bog'lanish graflar yordamida

tasvirlanadi. Bunda grafning uchlari yozuvlarni ifodalaydi. Graflarning qirralari yozuvlar o'rtasidagi aloqalarni ko'rsatadi. Jadvalli usulda esa obyekt to'g'risidagi ma'lumotlar bir yoki bir necha ustundan iborat bo'lgan jadvallar orqali ifodalanadi.

Hozirgi kunda mantiqiy modellarning pog'onali, tarmoqli va relyatsion turlaridan foydalanilmoqda. Shaxsiy EHMLarning paydo bo'lishi relyatsion modellarning keng tarqalishiga sababchi bo'ldi.

Pog'onali model chizmalı usul asosida tashkil qilinadi. Bunda ma'lumot yozuvlari grafikning uchlarini ifodalaydi va har bir yozuv oldingi pog'ona uchlariga bog'langan bo'ladi. Bunday tuzulishdagi MB dan tegishli axborotlar hamma vaqt bitta yo'nalish bo'yicha qidiriladi va uning joylashgan o'rni to'liq ko'rsatiladi. Masalan, «Talaba» to'g'risidagi ma'lumotlarni olish uchun «Fakultet», «Kurs», «Guruh» yozuvlarini ko'rsatilishi lozim.

Tarmoqli model ham chizmalı usul yordamida tashkil qilinadi. Lekin, bnda tegishli axborotlar bir necha yo'nalish bo'yicha olinishi mumkin. Masalan, «Talaba» to'g'risidagi ma'lumotlarni olish uchun yuqoridagi tasvirga «Muallim-fan» va «Fan-talaba» tarmoqli modeli hosil bo'ladi.

Relyatsion model jadvalli usul asosida tashkil qilinadi. Bunda tegishli ma'lumotlar jadvalning ustun va qatorlarida joylashadi. Ustunlar ma'lumotning maydonlarini, qatorlar esa yozuvlarini ifodalaydi. Bir ustunda ma'lum sohaga tegishli bo'lgan bir qancha ma'lumotlar joylashadi. Qatorda esa ustunlarda joylashgan ma'lumotlar ko'rsatiladi. Ustun va qator o'rtasidagi bog'lanish munosabat deb ataladi. Har bir ustun, qator va munosabat o'z nomiga ega bo'ladi.

Relyatsion modeldagi munosabatlar quyidagi talablar orqali hosil qilinadi:

-ustun va qator kesishgan yerda joylashgan ma'lumotlar elementlar hisoblanadi;

-munosabatlarda ikkita bir xil qator bo'lmaydi;

-ustun va qatorlarning tartibli joylashishi va nomlanishi majburiy emas.

MB ni tashuvchilarda hosil qilish bosqichi fizik tuzilishni tashkil etadi. Fizik tuzilish tashqi xotiralarda ma'lumotlarni joylashtirish usullari va vositalaridan iborat bo'lib, uni natijasida ichki model hosil qilinadi.

Ichki model ma'lumotning mantiqiy modelini tashuvchilarda aks ettiriladi va yozuvlarning joylanishi, aloqasi va tanlab olinishini ko'rsatadi. Ichki model MBBT orqali hosil qilinadi:

-ma'lumotlarning mantiqiy tuzilishini saqlash;

-tashqi xotiradan maksimal foydalanish;

-MB ni yuritish xarajatlarini kamaytirish;

-ma'lumotni qidirish va tanlash jarayonlarining tezkorligini oshirish va boshqalar.

Ma'lumki, bir algoritm bo'yicha turli tillar yordamida ekvivalent dasturlarini yaratish mumkin. Shunga bog'liq holda bitta mantiqiy model orqali bir qancha kichik (fizik) modellarni yaratish mumkin. Lekin aratilgan modellardan biri optimal bo'ladi. Shu sababli, MB ni hosil qiluvchi mutaxassislar oldida ichki modelning optimal variantini topish masalasi turibdi. Bunda optimallik mezoni sifatida yuqoridagi talablarni olish mumkin.

Saqlanayotgan ma'lumotlarning tuzilishi, ularni qidirish usullari va ifodalanish tillari fizik modellashtirishning asosiy vositalari hisoblanadi.

Ma'lumotlarning tuzilishini fayl yozuvlar ko'rinishida tasvirlash mumkin. Bunday holda yozuvlar maydonlardan, ularning joylanish tartibidan, turi va uzunligidan iborat bo'ladi. Ma'lumotlarni qidirish vaqtini kamaytirish maqsadida turli qidirish usullari yaratilmoqda. Lekin ma'lumotlarning tuzilishi yozuvlarning tezroq topish yo'lini ko'rsatadi. Shuning uchun ham, MB fizik tashkil qilishda ikkita tamoyilga: ma'lumotlarning tuzilishi va qidirish usullari asosida MB ni hosil qilishga rioya qilinadi. Har qanday MB fizik tashkil qilish natijasida fayllar hosil qiladi. Shaxsiy kompyuterlarda bu fayllar ketma-ket yoki ixtiyoriy tartibda joylanishi mumkin. Bunday fayllarni bajarishda, ya'ni MBBT da chiziqli va zanjirli ro'yxat, tartiblashmagan va tartiblashgan qidirish usullaridan foydalaniladi.

Chiziqli ro'yxat-MB ni fizik tashkil qilishning eng oddiy usuli hisoblanadi. Bunda MB ning fayllari bog'lanmagan holda bo'ladi va tegishli yozuvlarni qidirish ma'lum bir algoritmlar asosida amalga oshiriladi. Chiziqli usul orqali xotiradan

samarali foydalanish mumkin, lekin ma'lumotlarni qidirish uchun boshqa usullarga qaraganda ko'p vaqt sarf qiladi.

Zanjirli ro'yxat usulida hosil qilingan faylda har bir yozuv boshqa yozuv bilan bog'langan bo'ladi. Bunda aloqa vositasi sifatida ko'rsatkichlar ro'yxatidan foydalaniladi. Ko'rsatkichlar ro'yxati yozuvning qo'shimcha maydonlarida ko'rsatiladi va ular orqali kerakli ma'lumotlarni olish tartibi o'rnatiladi.

Ro'yxatga kirish uchun ro'yxatning boshlang'ich manzilgohini (RBM) ko'rsatish lozim. Bu manzilgoh ro'yxat sarlavhasida (RS) saqlanadi.

Ma'lumki MB ning yozuvlari asosiy maydon bo'yicha tartiblashgan bo'ladi. Lekin yozuvlarni asosiy maydon bo'lmagan ustunlar orqali tartiblashgan holda izlash mumkin. Buning uchun tartiblashmagan fayllr hosil qilinadi.

Tartiblashmagan fayllar kerakli ma'lumotlarni tez qidirish imkoniyatini bersada, ularda saqlanayotgan ma'lumotlar bir necha marta takrorlanadi. Natijada xotiradan foydalanish samaradorligi kamayadi. Bu kamchilikni tugatish maqsadida fayllar tartiblashgan holga keltiriladi. Bunday holatlarda yozuvlar emas, balki ularning joylashgan manzilgohlari saqlanadi. Kerakli ma'lumotlar manzilgohlar bo'yicha qilinadi va u xotirada kam joyni egallaydi.

Bunday fayl MBBT orqali avtomatik tarzda hosil qilinadi. Tegishli ma'lumotlar manzilgoh indekslarini izlash orqali chiqariladi.

MBBTning asosiy vazifalari va xususiyatlari. Ma'lumki MBBT dasturiy va til vositalarining to'plamidan iborat bo'lib, ular yordamida Mbni hosil qilish, yuritish, tahrirlash va boshqa vazifalarni bajarish mumkin. Bunday tizim yordamida operatsiya tizimining ma'lumotlarini boshqarish bo'yicha imkoniyatlari kengayadi.

MBBTning vazifalarini uch guruhga ajratish mumkin:

-fayllarni boshqarish; ya'ni faylni ochish, nusxa olish, nomini o'zgartirish, tuzilishini o'zgartirish, qayta hosil qilish, tiklash, hisobot olish, bekitish va boshqalar;

-yozuvlarni boshqarish; ya'ni yozuvlarni o'qish, kiritish, tartiblashtirish, o'chirish va boshqalar;

-yozuv maydonlarini boshqarish.

Shuni ta'kidlash lozimki, ma'lumotlarni harflar dastasi yordamida kiritish, hisoblash, takroriy jarayonlarni amalga oshirish, ma'lumotlarni ko'rsatuv oynasi yoki bosmaga chiqarish MBBTning vazifalari qatoriga kirmaydi. Bu vazifalar amaliy dasturlar yordamida bajariladi. Bunday dasturlar MBBTning maxsus dasturlash tillari orqali hosil qilinadi.

Yuqorida keltirilgan vazifalar to'plami MBBTda uch turdagi dasturlarning bo'lishini talab qiladi: boshqaruvchi dastur, qayta ishlovchi (translyator) dastur va xizmat ko'rsatuvchi dastur. MBBT ishga tushishi bilan asosiy boshqaruvchi dastur xotirasiga yuklanadi. Boshqa dasturlar tegishli holda ishga tushiriladi.

MBBTni turkumlashda mantiqiy tuzilish asos qilib olingan. Shuning uchun ham tarmoqli, pog'onali va relyatsion MBBTlari mavjud. Relyatsion MBBTlari keng tarqalgan bo'lib, ular jumlasiga dBase III Plus, FoxBase, Fox Pro, Clipper, dBase IV, Paradox va boshqalar kiradi.

MBBT foydalanuvchi bilan malumotlar bazasi o'rtasidagi aloqani ta'minlovchi dastur sifatida ishtirok etadi. Uning funksiyalari menyu va dasturlar ko'rinishida namoyon bo'ladi.

Menyu tartibida MBBTning funksiyalari ekranda tasvirlanadi. Foydalanuvchi kursorni xarakatlantirish orqali tegishli funktsiyani aniqlash va bajarishga chiqarishi lozim. Tizim aniqlangan funktsiyalarni bajarib bo'lgandan so'ng yana menyu holatiga qaytadi.

Dasturiy tartibda tegishli buyruqlar kiritiladi, dasturlar qayta ishlanadi va bajarishga chiqariladi. Bu holda MBBT interpretator tartibida ishlaydi va foydalanuvchidan dasturlash tillarini bilish talab qilinadi.

MBBTda foydalaniladigan dasturlash tillariga umumiy talablar bilan bir qatorda quyidagilar ham qo'yiladi:

- tilning to'liq bo'lishi;
- vazifalarni bajarish uchun tegishli vositalarning bo'lishi;
- aniqlangan ma'lumotlarni to'liq qayta ishlash va boshqalar.

Dasturlash tillar bir qator belgilarga ko'ra turkumlarga ajratiladi.

- o'zgaruvchanlik;

-jarayonlilik;

-foydalanilayotgan matematik apparat va boshqalar.

MBBTdagi dasturlar tegishli buyruqlarning to'plamidan tashkil topadi. Yechilayotgan masalalarning qiyinligiga qarab, dasturlar oddiy yoki murakkab tuzilishga ega bo'ladi. Oddiy tuzilishga ega bo'lgan dasturlarda buyruqlar ketma-ket joylashadi. Murakkab tuzilishli dasturlarda esa buyruqlar modullar holatida, ya'ni asosiy modul va quyi dasturlar to'plamidan iborat bo'ladi. Ma'lumotlar bazasini hosil qilishda modullik tamoyilidan foydalanish qulay va samaralidir.

1.2 SQL tili.

1.2.1 SQL tili va ma'lumotlar tiplari.

SQL ingliz tilidagi Structured Query Language yoki o'zbek tilidagi so'rovlar asosidagi strukturali til so'zlarining qisqartmasidan tashkil topgan. Bu til turli xildagi ma'lumotlar bazasiga murojaat qilish uchun standart imkoniyatlarga ega bo'lgan til hisoblanadi. Ko'plab bazalar bu til asosida o'zining versiyalariga ham ega. Lekin shunga qaramasdan SQL turli xil ko'rinishdagi ma'lumotlar bazalari bilan ishlash imkonini bera oladi.

SQL maxsus relyatsion ma'lumotlar bazasi bilan ishlashga mo'ljallangan til bo'lib, ma'lumotlar bazasini boshqarish tizimini boshqarish uchun mo'ljallangan sorovlar tilidir. SQL tilining bir necha standartdagi ko'rinishlari mavjud. SQL-89 va SQL-92 eng keng tarqalgan hisoblanadi. SQL-92 standarti ANSI standarti deb ham ataladi.

SQLda boshqa dasturlash tillardagi kabi buyruqlar ko'p emas. Dasturlash tillaridagi ba'zi buyruqlar umuman qo'llanilmaydi. SQL tilidagi buyruqlar SQL so'rov deb ataladi. SQL tili maqsadli ma'lumotlarga murojaat etishga yo'naltirilgan.

Barcha SQL so'rovlarni shartli ravishda ikki turga ajratish mumkin:

- **Statik SQL so'rovlar**-dastur kodiga kiritiladi va dastur bajarilishi jarayonida o'zgarmaydi. Bunday so'rovlarni o'zgartirish uchun faqat parametrlardan foydalanib amalga oshirish mumkin:
 - **Dinamik SQL so'rovlar**- dastur bajarilishi jarayonida hosil bo'ladi va o'zgaradi. Bunday so'rovlar ko'proq moslashuvchan bo'ladi va foydalanuvchining aniq bir harakatiga javob bera oladi.
- SQL tilida quyidagi asosiy ma'lumotlar tiplari ishlatilib, ularning formatlari har xil MBBT lar uchun farq qilishi mumkin.

1.1Jadval. SQL tilida tiplar.

INTEGER	-butun son (odatda 10tagacha qiymatli raqam va ishora).
SMALLINT	-“qisqa butun” (odatda 5 tagacha qiymatli raqam va ishora).
DECIMAL(p,q)	-o'nli son, p raqam va ishoradan iborat ($0 < p < 16$). O'nli nuqtadan so'ng raqamlar soni q orqali beriladi ($q < p$, agar $q = 0$ bo'lsa, tashlab yuborilish mumkin).
FLOAT	-haqiqiy son 15 ta qiymatli raqam va butun darajadan iborat. Darajada MBBT tipi bilan aniqlanadi (masalan: 75 yoki 307).
CHAR(n)	-uzunligi o'zgarmas, n ga teng bo'lgan simvolli qator ($0 < n < 256$).
VARCHAR(n)	-uzunligi o'garuvchi, n simvoldan oshmagan simvolli qator ($n > 0$ va har xil MBBTlarda har xil lekin 4096 dan kam emas).
DATE	-maxsus komanda orqali aniqlanuvchi formatdagi sana (Subaseda ko'zda tutilgan bo'yicha yy/mm/dd); sana maydonlari bizning eramizdan oldin bir necha mingyilliklardan boshlanuvchi va bizning eramiz beshinchi-o'ninchi mingyilligi bilan cheklangan haqiqiy sanalarni o'z ichiga olishi mumkin.

TIME	-maxsus komanda orqali aniqlanuvchi formatdagi vaqt (ko'zda tutilgan bo'yicha hh.mm.ss).
DATETIME	-sana va vaqt kombinatsiyasi. (Subaseda TIMESTAP).
MONEY	- maxsus komanda orqali aniqlanuvchi formatdagi pul. Format o'z ichiga pul birligi simvoli (\$, rub,...) va uning joylashuvi (suffiks yoki prefiks), kasr qism aniqligi va pul qiymatini ko'rsatish shartlarini oladi.

Jadvallarni yaratish.

Jadvallar CREATE TABLE komandasi bilan yaratiladi. Bu komanda qatorlarsiz bo'sh jadval yaratadi. CREATE TABLE komandasi jadval nomini va jadval o'zini ma'lum tartibda ko'rsatilgan ustunlar nomlari ketma-ketligi ta'rifi ko'rinishida aniqlaydi. U ma'lumotlar tiplari va ustunlar o'lchovini aniqlaydi. Har bir jadval juda bo'lmaganda bitta ustunga ega bo'lishi kerak.

CREATE TABLE komandasi sintaksisi:

```
CREATE TABLE <table-name>
(<column name>Mdate type>[(<size>)],
<column name>Mdate type>[(<size>)],...);
```

Argument qiymati kattaligi ma'lumot turiga bog'liqdir. Agar siz maxsus ko'rsatmasangiz, tizim avtomatik qiymatni o'rnatadi.

Jadvallarni o'chirish.

Jadvalni o'chirish imkoniga ega bo'lish uchun, jadval egasi (ya'ni yaratuvchisi) bo'lishingiz kerak. Faqat bo'sh jadvalni o'chirish mumkin. Qatorlarga ega bo'lgan, to'ldirilgan jadvalni o'chirish mumkin emas, ya'ni jadval o'chirishdan oldin tozalangan bo'lishi kerak. Jadvalni o'shirish komandasi quyidagi ko'rinishga ega:

```
DROP TABLE<table name>;
```

Jadvalni yaratilgandan so'ng o'zgartirish.

Jadvalni o'zgartirish uchun ALTER TABLE komandasidan foydalaniladi. Bu komanda jadvalga yangi ustunlar qoshish, ustunlarni o'chirish, ustunlar kattaligini

o'zgartirish, hamda cheklanishlarni qo'shish va olib tashlash imkoniyatlariga ega. Bu komanda ANSI standarti qismi emas, shuning uchun har xil tizimlarda har xil imkoniyatlarga ega.

Jadvalga ustun qo'shish uchun komandaning tipik sintaksisi:

```
ALTER TABLE <jadval nomi> ADD<ustun nomi>
```

1.2.3Jadvallar uchun cheklanishlar.

Cheklanishlarni kiritish.

Jadval yaratayotganimizda (yoki uni o'zgartirayotganimizda), siz maydonlarga kiritilayotgan qiymatlarga heklanishlar o'rnatishingiz mumkin. Bu holda SQL cheklanishlarga to'g'ri kelmaydigan hamma qiymatlarni rad etadi. Cheklanishlarning ikki asosiy turi mavjud: ustun va jadval cheklanishlar. Ularning farqi shundaki ustun cheklanishi faqat ayrim ustunlarga qo'llanadi, jadval cheklanishi bo'lsa bir yoki bir nechta ustunlar guruhiga qo'llanadi. Ustun cheklanishi ustun nomi oxiriga ma'lumotlar tipidan so'ng va vergul oldin qo'yiladi. Jadval cheklanishi jadval nomi oxiriga so'nggi dumaloq verguldan oldin qo'yiladi.

Cheklanishlar hisobga olingan CREATE TABLE komandasi sintaksisi:

```
CREATE TABLE <jadval nomi>  
(column name> <data type><column constraint>,  
(column name> <data type><column constraint>...  
<table constraint>(<column name>  
[<column name>])...);
```

Maydonga bo'sh (NULL) qiymatlar kiritilishi oldini olish uchun CREATE TABLE komandasida NOT NULL cheklanishi ishlatiladi. Bu cheklanish faqat har xil ustunlar uchun o'rnatiladi. Ko'p xollarda ustunga kiritilgan qiymatlar bir biridan farq qilishi kerak. Agar ustun uchun UNIQUE cheklanishi o'rnatilsa, bu ustungacha mavjud qiymatni kiritishga urinish rad etiladi. Bu cheklanish bo'sh bo'lmaydigan (NOT NULL) deb e'lon qilingan maydonlarga qo'llanilishi mumkin.

Unikalligi talab qilinadigan maydonlar (birlamchi kalitlardan tashqari) kandidat kalitlar yoki unikal kalitlar deyiladi.

Birlamchi kalitlar cheklanishlari.

SQL birlamchi kalitlarni to'g'ridan to'g'ri birlamchi kalit (PRIMARY KEY) cheklanishi orqali ta'riflaydi. PRIMARY KEY jadvalni yoki ustunlarni cheklashi mumkin. Bu cheklash UNIQUE cheklanishi kabi ishlaydi, faqat jadval uchun faqat bitta birlamchi kalit (ixtiyoriy sondagi ustunlar uchun) aniqlanishi mumkin bo'lgan xoldan tashqari. Birlamchi kalitlar NULL qiymatga ega bo'lishi mumkin emas.

Cheklanish FOREIGN KEY.

SQL ilovali yaxlitlikni FOREIGN KEY yordamida ta'minlaydi. Tashqi kalit vazifasi ajdod kalitda ko'rsatilmagan qiymatlarni tashqi kalit maydonlariga kiritmaslikdir. FOREIGN KEY cheklanishi sintaksisi:

```
FOREIGN KEY <column list>REFERENCES  
<fktable>[<column list>];
```

Birinchi ro'yxat komanda tomonidan o'zgartiriluvchi ustunlar ro'yxatidir. fktable-bu ajdod kalitli jadval. Ikkinchi ustunlar ro'yxati bu ajdod kalitni tashkil qiluvchi ustunlardir.

Kalitlarga cheklanish.

Ilovali yaxlitlikni ta'minlashni tashqi kalit yoki ajdod kalit maydonlari qiymatlariga cheklanishlar o'rnatishni talab qiladi. Ajdod kalit tarkiblangan bo'lib, tashqi kalit har bir qiymati bitta satrga mos kelishi ta'minlangan bo'lishi kerak. Bu kalit unikal bo'lib, bo'sh (NULL) qiymatlarga ega bolmasligi kerak. Shuning uchun ajdod kalit maydonlari PRIMARY KEY cheklanishiga ega bo'lishi yoki NOT NULL cheklanishi bilan birga UNIQUE deb e'lon qilinishi kerak.

Tashqi kalit ajdod kalitda mavjud qiymatlarga yoki bo'sh (NULL) qiymatga ega bo'lishi mumkin. Boshqa qiymat kiritishda urinish rad etiladi. Tashqi kalitga NOT NULL deb e'lon qilish muki, lekin bu maqsadga muvofiq emas.

1.2.4 Maydonlar qiymatlarini kiritish, o'chirish va o'zgartirish.

Qiymatlarni kiritish.

Hamma satrlarda Sqlda INSERT komandasi yordamida kiritiladi. INSERT quyidagi formatga ega:

```
INSERT INTO <table name | view name> [(column[,column]...)]
```

```
VALUES (<value>[,<value>]...);
```

Satrlarni o'chirish.

Satrlarni jadvaldan DELETE komandasi bilan o'chirish mumkin. U alohida qiymatlarni emas faqat satrlarni o'chiradi.

DELETE quyidagi formatga ega:

```
DELETE FROM< table name | view name>
```

```
[WHERE search-condition];
```

Maydon qiymatlarini o'zgartirish.

Bu o'zgartirish UPDATE komandasi yordamida bajariladi. Bu komandada UPDATE ifodasidan so'ng jadval nomi va SET ifodasidan so'ng ma'lum ustun uchun o'zgartirish ko'rsatiladi. UPDATE ikki formatga ega. Ulardan birinchisi:

```
UPDATE< table name | view name>
```

```
SET column=expression[,column=expression]...
```

```
[WHERE search-condition]
```

Bu yerda expression- bu ustun | ifoda | konstanta |o'zgaruvchi.

Ikkinchi variant:

```
UPDATE< table name>
```

```
SET column=expression,...
```

```
[FROM table-list]
```

```
[WHERE search-condition]
```

1.2.5 SELECT so'rov operatori.

SELECT operatori MB jadvallardan natijaviy to'plam olish uchun mo'ljallangan ifodadir. Biz SELECT operatori yordamida so'rov beramiz, u bo'lsa ma'lumotlar natijaviy to'plamini qaytaradi. Bu ma'lumotlar jadval shaklida

qaytariladi. Bu jadval keyingi SELECT operatori tomonidan qayta ishlanishi mumkin va hokazo.

Operator SQL-92 standartiga ko'ra quyidagi ko'rinishga ega:

Birinchi qoida, SELECT ifodasi o'z ichiga albatta FROM ifodasini olishi kerak. Qolgan ifodalar kerak bo'lsa ishlatiladi.

SELECT ifodasidan so'ng so'rovda qaytariluvchi ustunlar ro'yxati yoziladi.

FROM ifodasidan so'ng so'rovni bajarish uchun jadvallar nomi yoziladi.

WHERE ifodasidan so'ng agar ma'lum satrlarni qaytarish lozim bo'lsa, izlash sharti yoziladi.

GROUP BY ifodasi guruhlariga ajratilgan natijaviy so'rov yaratishga imkon beradi.

HAVING ifodasidan guruhlarini qaytarish sharti yoziladi va GROUP BY bilan birga ishlatiladi.

ORDER BY ifodasi ma'lumotlar natijaviy to'plamini tartiblash yo'nalishini aniqlaydi.

1.3 HTML tili va PHP dasturlash tili.

1.3.1 HTML tili.

Internet sayyoramiz miqyosidagi kompyuter tarmoqlarining birlashmasidir. Agar kosmik stansiya a'zolarining E-mail dan foydalanishini inobatga olsak internetning sayyoramiz sarhadlaridan chiqqanini ham ko'rishimiz mumkin. Butun dunyo o'rgimchak to'ri WWW (World Wide Web)-bu ko'p sonli o'zaro bir-biriga bog'langan hujjatlardir. Web sahifada boshqa boshqa Web sahifalar bilan bog'lovchi chekli gipermurojaat bo'ladi. Gipermurojaat mexanizmini bir necha so'z bilan izohlash qiyin, lekin kamida bir marta Web sahifaga kirib ishlagan odam buni darhol tushunadi.

Internet ehtiyoji uchun SGML tilining ma'lum bir qismi tanlab olindi va HTML (Hyper Text Markup Language) deb nomlandi. HTML tilida yozilgan Web sahifa faylida axborotning qanday tasvirlanishi haqidagi yo'l yo'riqlar ketma-ket joylashgan bo'ldi. Bu fayl oddiy matnli fayl bo'lib uni maxsus brauzer dasturlarsiz o'qish qiyin. Grafik rasmlarni esa umuman ko'rib bo'lmaydi, chunki uning o'rnida

brauzer dasturga biror rasm kerakligini ko'rsatuvchi TEG turadi. Agar biz o'z Web sahifamizni yaratmoqchi bo'lsak, albatt HTML tilini yaxshi bilishimiz kerak bo'ladi.

TEGlar.

HTML ning konsteuksiyasi TEGlar deyiladi. Brauzer TEGlarni oddiy matnlardan farqlashi uchun ular burchak qavslarga olinadi. TEG tasvirlash jarayoni hatti-harakatlarining boshlanishini bildiradi. Agar bu harakat butun hujjatga taaluqli bo'lsa, bunday teg o'zining yopiluvchi juftiga ega bo'lmaydi. Juft teglarning ikkinchisi birinchisining harakatini yakunlaydi. Masalan, har bir Web sahifa <html> tegi bilan boshlanib </html> tegi bilan yopilishi kerak. E'tibor bergan bo'lsangiz yopiluvchi teg ochiluvchidan « / » belgisi bilan farqlanadi. Teg nomlari katta yoki kichik harflar bilan yozilishi mumkin, bularni brauzer bir xil qabul qiladi. HTML tilida boshqa kompyuter tillaridagi kabi izoh berish imkoniyati mavjud. Izoh quyidagi « <- - » va « - ->» belgilar orasiga yoziladi.

Masalan:

<- - Bu izoh- ->

Har qanday Web sahifa ikkita qismdan tashkil topadi. Bular sarlavha qismi va asosiy qism. Sarlavha qismida Web sahifa haqidagi ma'lumot joylashadi, asosiy qismda esa Web sahifaning mazmuni bilan tasvirlanish qoidalari joylashadi. Sarlavha qismi quyidagi ochiluvchi <head> va yopiluvchi </head> teglari orasida joylashadi. Asosiy qism esa <body> va </body> teglari orasida joylashadi. Odatda sarlavha qismi oldindan qo'llanilayotgan HTML standarti haqida ma'lumot yoziladi. Har qanday Web sahifaning umumiy ko'rinishi quyidagicha bo'ladi:

Misol:1

```
<!DOCTYPE HTML PUBLIC "-//W3S//DTD HTML 4.01//EN ">
```

```
<html>
```

```
<head>
```

```
Hujjat sarlavhasi
```

```
</head>
```

```
<body>
```

Asosiy qism

```
</body>
```

```
</html>
```

Natija: Asosiy qism

Birinchi <!DOCTYPE> tegi o'zining parametrlari bilan brauzerga ushbu Web sahifani qaysi HTML versiyada yozilganligi haqida ma'lumot beradi.

Web sahifa ishga tushirilganda brauzerning eng yuqori satrida yuklanayotgan hujjat mazmunini anglatuvchi qisqacha yozuv turadi. Bu yozuvni hosil qilish uchun quyidagi ochiluvchi <title> va yopiluvchi </title> teglaridan foydalanamiz.

Web sahifaning asosiy qismi <body> va</body> teglari orasida joylashadi. Bu oddiy matn bo'lishi mumkin. Brauzer bu matnni to'g'ridan-to'g'ri interpretatsiya qilib ekranda tasvirlaydi. Bizga dastlabki Web sahifamizni yaratish uchun oddiy «Bloknot» matn muharriri kifoya. Quyida ko'rsatilgan misolni matn muharririda yozib, uni xotiraga yozishda kengaytmasini html yoki htm deb kiritishimiz kerak.

Misol:2

```
<html>
```

```
<head>
```

```
<title>Mening birinchi Web sahifam </title>
```

```
</head>
```

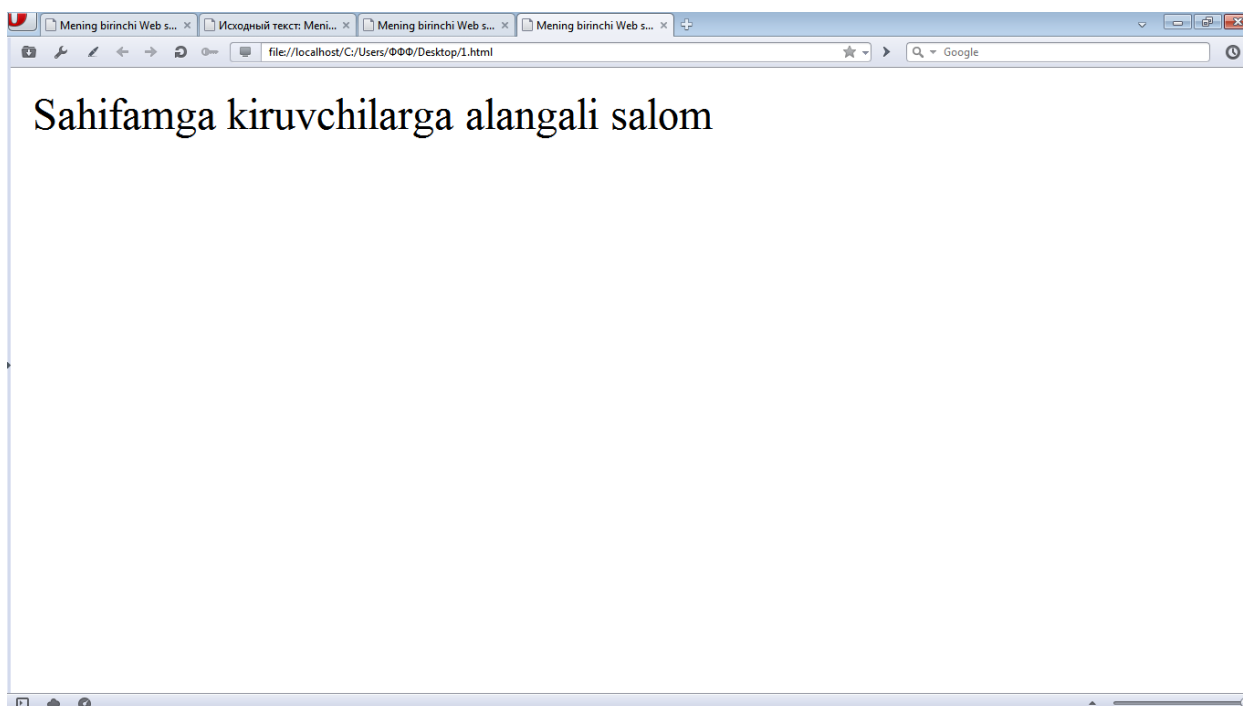
```
<body>
```

```
Sahifamga kiruvchilarga alangali salom
```

```
</body>
```

```
</html>
```

Bu faylni ishga tushirish uchun sicqoncha ko'ratkichini shu fayl ustiga keltirib chap tugmasini ikki marta bosish kerak. Natijada ekranda hosil bo'ladi:



`<body>` tegi bir qancha qo'shimcha parametrlarga ega. Bu parametrlar tegning ochiluvchi qismida joylashadi. Parametrlar ikki qismdan iborat bo'ladi: parametr nomi va parametr qiymati. Masalan *bgcolor* parametri tasvirlanayotgan Web sahifa fonining rangini belgilaydi.

Masalan: `<bgcolor="green">`

Parametrlarning satrli qiymatlari qo'shtirnoq ichida yoziladi. Biz quyidagi `<body>` tegining parametrlari bilan tanishamiz.

- Background-fon sifatida biror bir grafik tasvirdan foydalanish. Parametr qiymati sifatida grafik tasvir joylashgan manzil (URL) beriladi.
- Text- tasvirlanayotgan matn rangi.
- Link-Web sahifadagi matnli gipermurojaat rangi.
- Vlink-foydalanuvchi tomonidan oldin murojaat qilingan gipermurojaat rangi.
- Alink-foydalanuvchi tomonidan tanlangan gipermurojaat rangi.
- Lang-Web sahifa matni yozilgan tilni aniqlash.

Identifikatorlar.

HTML tilida har bir qo'llanilgan tegga unikal identifikator berish imkoniyati mavjud. Masalan matn bir necha abzatlardan tashkil topgan bo'lsin. Har bir

abzatsga maxsus nom berish mumkin, keyinchalik HTML tilining qo'shimcha imkoniyatlari yordamida bu abzatlarni boshqarish mumkin, ya'ni ularning birortasini ko'rinmas qilish yoki shrifti rangini o'zgartirish.

HTML hujjatda *id* parametri qiymatlari takrorlanmasligi lozim, aks holda bu qiymatlar e'tiborga olinmaydi.

HTML hujjatini tashkil etuvchi barcha elementlar ikki turga bo'linadi: *inline* elementlar va *blokli* elementlar. Inline elementlar oddiy matn elementlari bo'lib satr qismi ham bo'lishi mumkin, blokli elementlar esa har safar yangi satrdan boshlanishi shart. Blokli elementlar boshqa blokli elementlar va inline elementlaridan tashkil topishi mumkin, lekin inline elementlar blokli elementlarni o'z ichiga olmaydi. Web sahifa elementlarini bloklarga birlashtirish ularga birvarakayiga shakl berish imkonini beradi, ya'ni birlashtiruvchi yagona tegni aniqlab blok joylashuvini o'zgartirish mumkin. Tabiiyki bu Web sahifa elementlarning har birininh joylashuvini alohida o'zgartirishdan oson.

Blokli tip elementlarini birlashtirish uchun `<div>` va `</div>` yeglari qo'llaniladi. Inline elementlari esa `` va `` teglari orqali birlashtiriladi. `` va `<div>` teglari qo'shimcha parametrlarni ham o'z ichiga olishi mumkin. Bizga ma'lum bo'lgan *id* va *class* parametrlaridan tashqari *style* va *align* parametrlari ham ishlatiladi. *style* parametri shu blokdagi ma'lumot stilini o'rnatadi, *align* parametri shu ma'lumotni qanday tekislashni aniqlaydi. HTML hujjatida sarlavhaning o'z teglari mavjud bo'lib, ular oltitadir. Eng yuqori darajasi bu birinchidir. Har bir sarlavhaning o'z tegi va o'z tasvirlanish qoidasi mavjud. Eng katta ya'ni birinchi darajali sarlavha `<h1>` va `</h1>` teglari orqali, ikkinchi darajali sarlavha `<h2>` va `</h2>` teglari orqali va oxirgi oltinchi darajali sarlavha `<h6>` va `</h6>` teglari orqali ifodalanadi.

Ranglar va o'lchov birliklari.

HTML hujjatining kodida biz hamisha biror bir bezak obyektning o'lchamlarini yoki ularning ranglari xususiyatlarini ko'rsatishimizga to'g'ri keladi.

HTML tilida rang va o'lchov birliklarini qo'llashning standart qoidasi mavjud.

Rang berishning ikkita usuli mavjud. Ko'p qo'llaniladigan usul kerak rangning RGB kodini ko'rsatishdir. Ma'lumki har qanday rangni uchta asosiy:qizil yashil va ko'k ranglarning qarishmasidan hosil qilish mumkin. Brauzerlar bizga o'n olti milliondan ortiq rangni tasvirlash imkonini beradi, chunki asosiy uchta rangdan har birining qiymati 0 dan 255 gacha qiymat qabul qiladi. Ixtiyoriy rang har biri asosiy ranglarning ulushini ifodalovchi 3ta son majmuasidan iborat bo'ladi.

HTML tilida rang qulaylik uchun 16lik sistemadagi 6ta raqamdan tashkil topadi.

Masalan:

Color="#FF0000"

16lik raqamlar oldida "#" belgisi qo'yiladi. Rang ulushlarini ko'rsatib turuvchi raqamlar tartibiga e'tibor berish kerak. Chunki birinchi qizil, ikkinchi yashil va uchunchi ko'k rang ulushlari joylashadi.

height="3"

Karrali o'lcham berish belgisi sifatida yulduzcha(*) belgisi ishlatiladi. karrali son koeffitsiyenti hisoblanganda (*) belgisining chap tomonidagi son 10 ga ko'paytiriladi. Brauzer bunday obyektlarni maksimal o'lchamda tasvirlashga harakat qiladi.

Matnlarni bezash.

HTML tilida matni tasvirlashning bir qancha usullari mavjud. Brauzer ekranida matn satrini tasvirlash uchun hech qanday teg ishlatishga hojat yo'q. Matnni yozish kifoya. Lekin uni hattoki abzatsga bo'lish ham teglarsiz amalga oshmaydi. Har xil kompyuter tizimlarida matnlarni abzatsga bo'lish uchun har xil simvollar ishlatiladi, lekin HTML hujjati kompyuter tizimi qanday bo'lishidan qat'iy nazar bir xil tasvirlanishi lozim va shuning uchun abzatsni ifodalovchi teg kiritilishiga to'g'ri kelgan. Har bir abzats boshida <p> tegi qo'yiladi, oxirida esa yopiluvchi </p> tegi qo'yiladi. Bu teg o'z parametrlariga ega. Bu parametrlar qatoriga identifikatsiya parametrlari *class* va *id*, shaklli bezash parametri *style* va tekislash(tenglash) parametri *align* kiradi.

Abzatsni brauzar oynasining o'ng yoki chap tomoniga tekislash, markazlashtirish yoki to'la eniga yoyib yozish uchun ularga mos ravishda *left*, *right*, *center* va *justify* qiymatlari ishlatiladi.

Ba'zan Web sahifa yaratuvchilar bzatslar oralig'ini kengaytirish uchun bo'sh abzatlardan foydalanishadi, ya'ni abzatsning ochiluvchi va yopiluvchi teglarining ichiga hech narsa yozmasdan qo'llashadi. Brauzerlar esa bu narsani e'tibordan chetda qoldiradilar. Shuning uchun abzatlarni ajratish yoki biror abzatsning ichidagi satrni bo'lib keyingi satrga o'tkazish uchun `
` tegi ishlatiladi. bu teg matnning shu qismini keyingi satrga o'tkazish kerakligini anglatadi.

Font tegi.

Endi matn shriftlarini bezash usullarini ya'ni font tegini ko'rib chiqamiz. Biz `` tegini parametrlari bilan birga abzatsning ixtiyoriy yerida qo'llashimiz mumkin. Bu tegdan keyingi matnlar parametrda ko'rsatilgan qiymatlar bo'yicha ekranda tasvirlanadi. `` tegini yopiluvchi tegi `` shunday ko'rinishda bo'ladi. `` tegi o'zining qo'llanilayotgan shrift o'lchamini ko'rsatuvchi *size*, shrift simvollarining rangiga belgilovchi *color* va matn qaysi shriftda tasvirlanishini belgilovchi *face* parametrlariga ega. *Size* parametri qiymat sifatida 0 dan 7 gacha bo'lgan sonlarni qabul qiladi. Bu sonlar matndagi simvollar o'lchamini bildiradi. Foydalanuvchi brauzeri biz ko'rsatgan shrift o'lchamiga maksimal mos keluvchi o'lchamni tanlashi kerak. *Size* parametri qiymati sifatida biz shrift o'lchamini berishimiz mumkin. Masalan shrift o'lchamini bittaga oshirish uchun quyidagi konstruksiyani yozishimiz kerak:

```
<font size="+1">
```

Shrift (simvol) o'lchamlarini ikki birlikka kichraytirish uchun esa quyidagini yozishimiz kerak:

```
<font size="-2">
```

font tegining navbatdagi parametri bu *color* parametridir. *Color* parametri quyidagicha yoziladi:

```
<font color="rang">
```

Masalan qo'llanilayotgan shrift simvollarini yashil rangda tasvirlash uchun quyidagini yozishimiz lozim:

```
<font color = "green">
```

Navbatdagi *face* parametri esa qo'llanilishi kerak bo'lgan shriftni belgilaydi. Biz matnni *face* yordamida *Times New Roman* yoki *Copperplate Gothic* shriftlari yordamida tasvirlanishini ko'rsatishimiz mumkin.

1.3.2 Ro'yxatlar haqida umumiy tushuncha.

HTML hujjatda ma'lumotlarni qulay shaklda tasvirlash maqadida ro'yxatlar va jadvallar tashkil qilingan. Ro'yxatlar elementlarini nuqtalari bo'yicha tartiblasa, jadvallar esa ma'lumotlarni satrlar va ustunlar bo'yicha joylashtiradi. Ro'yxatlardan to'g'ri foydalanish katta hajmli ma'lumotlarning o'qilish qulayligini oshiradi. Ro'yxatlarni quyidagi uchta holatda qo'llash maqsadga muvofiq:

1. Bir jinsli ma'lumotlarni biror belgilar bo'yicha sinflarga ajratish;
2. Katta hajmli ma'lumotlarni abzats bilan tasvirlashda ularni o'qish qiyinlashsa;
3. Biror bir jarayonni tasvirlashda;

Quyidagi bir misolni ko'ramiz:

Misol:4

```
<html>
```

```
<body>
```

Fanlar

```
<ul>
```

```
<li>Informatika
```

```
<li>Matematika
```

```
</ul>
```

```
</body>
```

```
</html>
```

Natijada: Fanlar

- Informatika

- Matematika

Bundan ko'rinadiki ro'yxatlar ma'lumotlarni oson va tushunarli o'qilishini ta'minlaydi. Ro'yxatlar asosan 3 ta asosiy tiplarga bo'linadi. Bular: markirlangan, nomerlangan va aniqlovchi ro'yxatlar. Bulardan tashqari boshqacha tipli ro'yxatlar ham mavjud, ammo ko'p qo'llaniladiganlari biz sanagan uchtasidir. Yuqorida biz sanab o'tgan tiplar bir-biridan nomerlash usullari va tuzulishi bilan farq qiladi.

Markirlangan ro'yxatlar.

Ko'pchilik hollarda nomerlangan yoki markirlangan ro'yxatlardan foydalaniladi. Bunday ro'yxatlarda har bir punkt kichik piktogramma bilan boshlanadi. Markirlangan ro'yxatlar va teglari yordamida beriladi.

Misol:5

```
<html>
  <head><title>Markirlangan ro'yxatlar</title></head>
  <body>
    <ul>
      <lh><b>Kompyuter avlodlari</b></lh>
      <li> I-avlod
      <li>II-avlod
      <li>III-avlod
      <li>IV-avlod
    </ul>
  </body>
</html>
```

Natija: Kompyuter avlodlari

- I-avlod
- II-avlod
- III-avlod
- IV-avlod

Bu yerda <lh> va </lh> teglari orasida ro'yxat sarlavhasi yozilgan. Barcha va teglari va markirlangan teglar orasiga joylashgan bo'lishi shart.

 tegi *type* parametriga ega va biz bu parametr qiymatlari bilan tanishib chiqamiz. Bu parametr yordamida ro'yxatda hosil qilinadigan ro'yxat elementlari oldidagi markerlar tipi aniqlanadi. Parametr qiymati sifatida quyidagi uchta qiymatlardan foydalanosh mumkin:

1. *disc* –bu qiymat brauzerga markerarni kichik qoraga bo'yalgan doira shaklidaligini ko'ratadi.
2. *circle*-bu qiymat esa brauzerga markerning bo'yalmagan aylana shaklida bo'lishini bildiradi.
3. *square*-bu qiymat markerning kichik to'g'ri to'rtburchak shaklida bo'lishini bildiradi.

Ta'kidlab o'tish kerakki u yoki bu hosil qilinadigan markerlarni bir xilligi hamma vaqt ham bir xil bo'lavermaydi. Brauzer turiga qarab har xil bo'lishi mumkin.

Nomerlangan ro'yxat.

Nomerlangan ro'yxat markerlangan ro'yxatga o'xshaydi. Yagona farqi shundaki nomerlangan ro'yxatda har bir punkt oldiga grafik marker emas balki avtomatik ravishda ketma-ket nomerlangan yoki harflar joylashtiriladi.

Nomerlangan ro'yxat va teglri yordamida beriladi, ro'yxatdagi punktlar esa , bilan aniqlanadi. Jimlik bo'yicha nomer 1 dan boshlanadi.

Misol:6

```
<html>
  <body>
    <ol type=1,2,3..>
      <lh><b>O'tiladigan fanlar ro'yxati</b></lh>
      <li>Informatika
      <li>Dasturlash asoslari
    </ol>
  </body>
</html>
```

Natija: O'tiladigan fanlar ro'yxati

1. Informatika

2. Dasturlash asoslari

Formalar haqida umumiy ma'lumot.

HTML formalar, foydalanuvchilar tomonidan ma'lumotlarni kiritish va kiritilgan ma'lumotlardan keyinchalik foydalanish maqsadida tashkil qilingan. Formalar – matnli oynalar, tugmalar, bayroqchalar, ulagichlar (pereklyuchatel) lar va menyular kabi boshqariladigan elementlardan tashkil topishi mumkin. Formalar `<form>` va `</form>` teglari orasida joylashadi.

Ba'zi hollarda foydalanuvchi bilan brauzer o'rtasida aktiv muloqot qilishga to'g'ri keladi, ya'ni ma'lumotlarni kiritishga va tahrir qilishga to'g'ri keladi. Bunday hollarda biz yuqorida aytganimizdek HTML hujjatning formalar deb ataluvchi bo'limidan foydalanamiz. Foydalanuvchi yuqorida ko'rsatilgan formalarni ma'lumotlar bilan to'ldiradi va uni qayta ishlash uchun serverga jonatadi. Server bu odatda Web-server yoki elektron pochta serveri bo'lishi mumkin.

HTML formalar quyidagi boshqariladigan elementlardan foydalanadi:

Tugmalar (buttons)

Uchta turdagi tugmalar mavjud:

- Jo'natuvchi tugmalar- bu tugmalar yordamida formalar serverga qayta ishlash uchun jo'natiladi.
- Inkor etuvchi tugmalar- bu tugma bosilsa formaning hamma qiymatlari qaytadan boshlang'ich holatiga o'rnatiladi.
- Umumiy tugmalar – bunday turdagi tugmani bosganda shu tugmaga bog'langan protsedura ishga tushadi.

Bayroqchalar (checkboxes)

Bayroqcha- bu ikkita (tanlangan yoki tanlanmagan) holatga ega bo'ladigan element. Bayroqchalar *input* elementi yordamida hosil qilinadi.

Ulagichlar (radiobuttons)

Ulagichlar ham bayroqchalarga o'xshaydi, ammo bitta farqi hamma elementlari bitta nomga ega bo'ladi va shulardan bittasi tanlanadi. Ulagichlar ham *input* yordamida tashkil qilinadi.

Menyular (menus)

Menyu foydalanuvchiga bir nechta variantlardan bir yoki bir nechtasini tanlash imkoniyatini yaratadi. Menyu *select* elementi yordamida *ortgroup* va *option* lar bilan birgalikda tashkil qilinadi.

Matn kiritish maydoni (text input)

Matnni foydalanuvchi tomonidan kiritilishini amalga oshiruvchi ikkita element mavjud. *Input* elementi bitta qatorli kiritishni ta'minlaydi. *Textarea*- ko'p qatorli kiritishni amalga oshirish uchun ishlatiladi.

Formalar tashkil qilish form elementi yordamida amalga oshiriladi.

1.3.3.PHP dasturlash muhiti.

PHP dasturlash tili maxsus internet bilan ishlash uchun mo'ljallangan. Aslida PHP serverda qayta ishlashni amalga oshiradigan HTMLga qo'shimcha script tili deb ataladi. PHP-serverda bajarilishi bilan JavaScriptdan, HTML_kod bilan aralashgan holda yoziladi.

PHP dasturni o'rnatish – bu doim murakkab jarayon, sababi hech bir tizim boshasiga o'xshamaydi, har bir holatda oldin hech kim to'qnashmagan muammolar yuzaga kelishi mumkin. Har holda ma'lumotlar omboringiz o'rnatilgan va kiritilgan, PHP-dvijok kompilirlangan, o'rnatilgan va Web-server bilan bog'langan, bu server .php3 kengaytmali hujjatlar bilan nima qilish kerakligini biladi.

Web sahifalarni va tarmoqni yaratish yo'llari o'zgarishi bilan PHP ham takomillasha bordi. 1990-yil o'rtalariga kelib katta tarmoqlarda ham HTML da yozilgan yuzlab statik sahifalar ishlatilar edi. Hozir esa jarayon o'zgarib bormoqda. Web sahifalarni yaratuvchilar ma'lumotlar bazasi bilan ishlovchi Web sahifalarni yaratish imkoniga ega bo'lib, foydalanuvchilarni qayta ishlash imkoniga ega bo'lgan Web sahifalarni yaratmoqdalar.

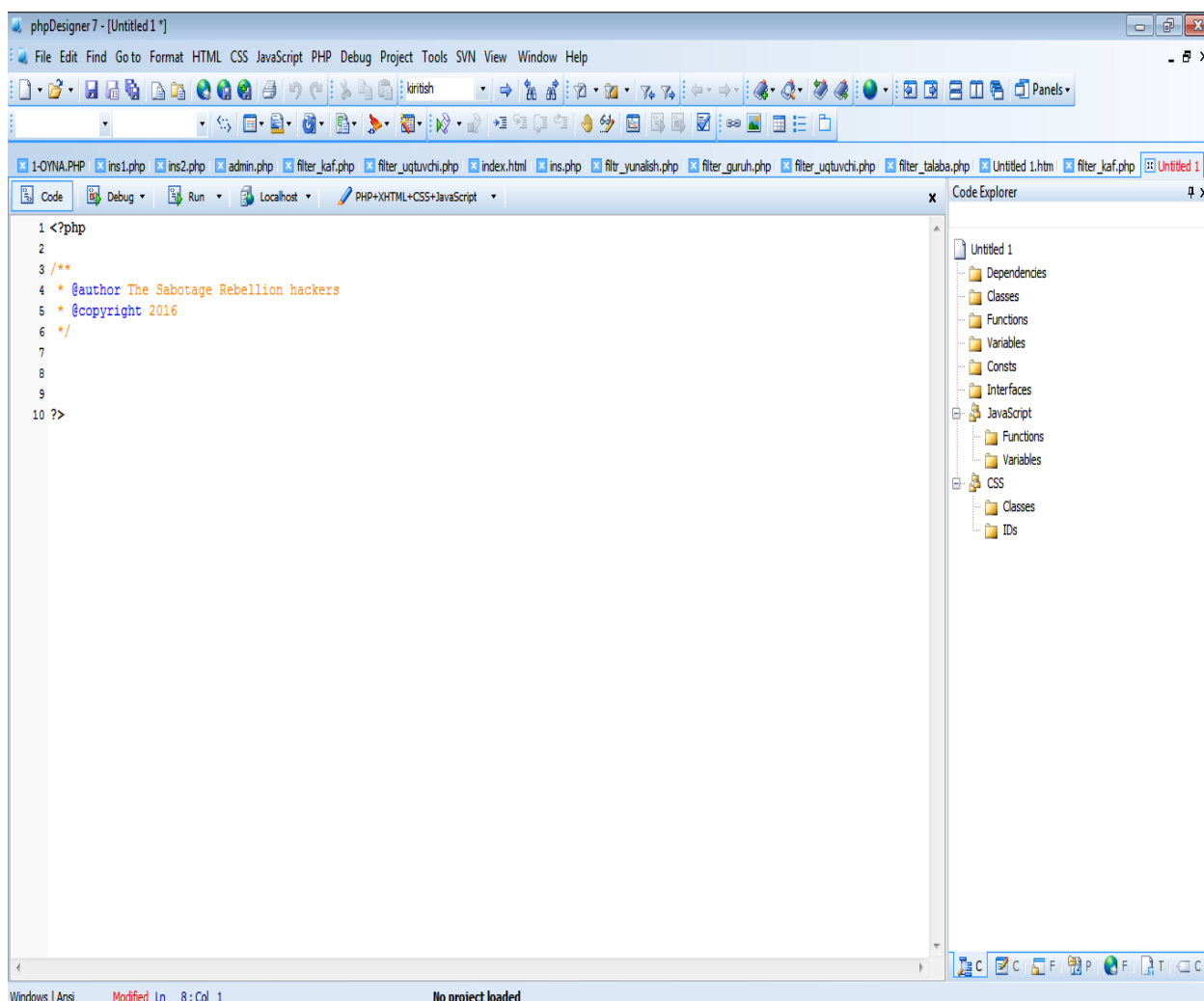
Ma'lumotlarni saqlash va ma'lumotlarga murojaat qilish uchun ma'lumotlar bazasidan foydalanish yanada aktuallashib, mobil telefonlar, raqamli televideniya

va hokazolar. Turli xil qurilmalarda ma'lumotlarni uzatishda sifatni o'sishiga erishishmoqda.

PHP dasturlari ikki usulda bajarilishi mumkin: Web server tomonidan ssenriy ilovasi va konsol dasturi sifatida.

Bizning maqsadimiz Web ilovalarni dasturlash bo'lgani uchun asosan birinchi usulni ko'ramiz.

PHP odatda internet bilan bog'liq dasturlar yaratish uchun ishlatiladi. Serverga brauzerning murojaat qilishi yordamida php-ssenariylari bajarilishini ko'rib chiqamiz. Avval brauzer .php kengaytmali sahifani so'raydi, so'ngra Web server dasturni PHP mashinadan o'tkazadi va natijani html-kod sifatida qaytaradi. Agar standart HTML sahifani olib, kengaytmasini .php ga o'zgartirilsa va PHP mashinadan o'tkazilsa, foydalanuvchiga o'zgartirmasdan qaytaradi. Bu faylga PHP komandani qo'shish uchun, PHP komandalarni maxsus teglar ichiga olish kerak. PHP kodlar alohida maxsus dasturiy vositalardan tashqari Notepad++, Bloknot, Dreamwiaver va shu kabi muhitlarda yozib xotiraga saqlashda kengaytmasini .php deb saqlasak ham ishlayveradi. Endi phpDesigner7 dasturini ishga tushirishni ko'rib chiqamiz. Buning uchun start denver yorlig'i ustiga sichqonchani chap tugmasini ikki marta bosiladi. Keyin esa Пуск-> phpDesigner ni tanlanadi va openni bosib keyingi oynaga o'tiladi. Kod yozish uchun PHP-> File-> New file tanlanadi.



1.2-chizma. phpDesigner7 oynasining ko'rinishi.

PHPda o'zgaruvchilar dollar (\$) belgisidan boshlanadi. Bu simvoldan keyin ixtiyoriy sondagi harf, raqam va ostiga chiziq simvollari kelishi mumkin, lekin birinchi simvol albatta harf bo'lishi kerak. Shuni esda tutish kerakki, PHPda o'zgaruvchilarning nomlari kalit so'zlardan farqli bo'lishi kerak.

PHP da o'zgaruvchilarni tariflaganda oshkora tipini ko'rsatish shart emas va dastur davomida bitta o'zgaruvchi har xil tiplarga ega bo'lishi mumkin. O'zgaruvchi unga qiymat berilganda initsializatsiya qilinadi va dastur bajarilgunicha mavjud bo'ladi. Ya'ni web sahifa holida to so'rov tugamagunicha. PHP dasturlash tili buyruqlari belgilar bilan ajratilgan ma'lum bir blok ichida yoziladi. Yozilgan kodlar serverga jo'natiladi va server php kodlarni shu ajratilgan blok orqali aniqlab oladi.

PHP-programmalarini yaratish, serverga nusxasini ko'chirish va ishga tushirish PHP komandalari va HTML matnini bitta hujjatga bog'lash, dastur matnini tushuntirish uchun izoh keltirish.

PHP-dasturni tuzish uchun ixtiyoriy matn redaktorini ishga tushiramiz. PHP-dasturi HTML hujjatlari kabi oddiy matndan tashkil topadi. Shuning uchun dasturni ixtiyoriy matn redaktorida yozish mumkin.

1-misol.

```
<?
Echo "<html><body><h1>";
Echo "PHP,$uzg";
Echo "</h1></body></html>";
?>
```

Dasturni misol.php fayli ko'rinishida saqlaymiz. Local serverning DocumentRoot katalogida joylashtiramiz. Bu katalog windowsda \www\html bo'ladi. Endi brauzer oynasida [http://localhost/misol.1.php?\\$uzg=tili](http://localhost/misol.1.php?$uzg=tili) adresini teramiz. Natijada "PHP tili" matni brauzer oynasida hosil bo'ladi.

Brauzer oynasida ushbu sahifa HTML kodi ko'zdan kechiramiz:

```
<html><body><h1>
PHP tili
</h1></body></html>
```

Demak, PHP kodning natijasi HTML sahifani hosil qilishdan iborat. Bu misolda PHP kod <? va ?> ichida yozilgan edi. <?-ssenriy boshlanganligini, ?>-ssenariy yakunlanganligini bildiradi. Echo-axborotni chop etish uchun xizmat qiladi. Chop etiladigan axborot qo'shtirnoq ichida yoziladi.

O'zgaruvchi va konstantalar.

Boshqa tillardagi kabi PHP da ma'lumotlarni saqlash uchun o'zgaruvchi va konstantadan foydalaniladi.

2-misol.

```
<?
$number=7; //o'zgaruvchiga 7 qiymatni taqdim etish
```

```
Echo $number; //$number o'zgaruvchisini qiymatini chiqarish.
```

```
?>
```

O'zgaruvchilarning qiymatini osonlikcha o'zgartirish mumkin.

3-misol.

```
<?
```

```
$number=7; //o'zgaruvchiga 7 qiymatni taqdim etish
```

```
$number=$number+3; //o'zgaruvchi qiymatini 3 taga oshirish.
```

```
Eco$number; ekranga 10 chiqariladi.
```

```
?>
```

O'zgaruvchi nomi shunday tanlanishi kerakki, o'zgaruvchi o'zida nima saqlayotgani uning nomidan aniq bo'lishi kerak. Bu katta dasturlar bilan ishlaganda juda muhim hisoblanadi. Biroq o'zgaruvchi nomi juda ham uzun bo'lmasligi lozim.

Dasturlashda ma'lumot turini bilish ham katta ahamiyatga ega. Bu narsa ushbu ma'lumot ustida qancha amallarni bajarish mumkinligini aniqlab beradi. PHP – to'rtta skalyar (Integer, Double, Boolean, String) va (Array, Object) turdagi ma'lumotlar bilan ishlaydi.

Integer turi.

Integer turi 32 bitli platformalarda -2147483648 dan 2147483647 oraliqdagi sonlar bilan amal bajaradi.

4-misol.

```
<?php
```

```
$dec_int_number=7; //butun turdagi o'zgaruvchi
```

```
Echo 7; //butun sonni chiqarish
```

```
?>
```

Double turi.

Double (float) turiga ikkilik aniqlikdagi yoki suzuvchi nuqtali deb ataladigan haqiqiy sonlar to'plami mos qo'yiladi.

5-misol

```
<?php
```

```
$first_double_number=123.91;  
$second_double_number=1.2e2; //120ga mos keladi  
$thrid_double_number=1E-3; //0.001 ga mos keladi.
```

```
?>
```

Double turidagi o'zgaruvchilar katta sonlar (1.8e308), shuningdek yuqori aniqlikdagi (kasr qismi 14 xonagacha) masalalarni yechishda qo'llaniladi.

Boolean turi.

Boolean turi eng sodda ma'lumot turi hisoblanadi. Faqat ikkita qiymat-TRUE(rost), FALSE(yolg'on) qiymatlarini qabul qiladi.

6-misol

```
</php  
#boolean_var=TRUE;  
?>
```

Boolean turidagi o'zgaruvchilar bir qiymatning rost yoki yolg'on ekanligini aniqlash uchun qo'llaniladi. TRUE, FALSE qiymatlarini kichik harflarda ham yozish mumkin.

Array turi.

Massiv ma'lumotlarning tartiblangan to'plamidir. Uning har bir elementi indeks yoki kalitga ega. Bitta massivda ikkita bir xil indeksga ega element bo'lmaydi. Massivlar bilan ishlash juda qulay. Array ko'rsatmasi maxsus massiv hosil qilish uchun qo'llaniladi. U bo'sh massivlar hosil qilish imkonini ham beradi.

7-misol.

```
<?php  
$A=array("Fayziyeva"=>"O'lmasoy","Mudirova"=>"Maysara");  
//ikki elementli assotsiativ massiv hosil qiladi.  
$B=array("Fayziyeva","Mudirova");  
//ikki elementli ro'yxat hosil qiladi. Indekslar 0 dan boshlanadi.  
?>
```

String turi.

PHP da ma'lumotlarning string turi uzunligi amalda chegaralangan simvollar satriga mos qo'yiladi.

8-misol.

```
<?php
$atr="PHP tili"; //string turidagi o'zgaruvchi.
$bo'sh_atr=""; //PHP da bo'sh satrlar
$atr='salom' //atr yakka tirnoqda bo'lishi mumkin.
?>
```

Object turi.

Object turi obyektga yo'naltirilgan dasturlashda qo'llaniladi.

Ma'lumotlarning boshqa turlari.

PHP da ma'lumotlarning skalyar va aralash turidan boshqa ikkita maxsus (resource, NULL), bir nechta psevd (mixed, number, callback) ma'lumot turlari mavjud. Ular amaliyotda kam qo'llaniladi.

PHP da konstanta.

PHPda konstantalarni aniqlash uchun define() funksiyasi qo'llaniladi. Konstantalarni ham bosh, ham kichik harflar bilan yozish mumkin. Faqat bir xil nomlangan turli harflarda yozilgan nomlar turlicha hisoblanadi. O'zgarmlar quyidagi shaklda beriladi:

```
Define($name, $value, $case_sen)
```

Bu yerda \$name- konstanta nomi, \$value konstanta qiymati, \$case_sen- mantiqiy turdagi majburiy bo'lmagan ko'rsatkich bo'lib, harf katta yoki kichik ekanligini e'tiborga olish kerak yoki kerak emasligini ko'rsatadi.

Konstantalarni e'lon qilishda ikkita asosiy narsaga e'tibor qaratish lozim.

- Konstanta nomi apostrofga emas balki qo'shtirnoqqa olib yoziladi.
- Bir xil nomdagi ikkita konstanta e'lon qilish mumkinmas.

PHP da amallar va ifodalar.

Ifodalar PHPning eng muhim qurilish elementlari hisoblanadi. PHPda ma'noga ega yozuvlar ifoda hisoblanadi. Ifodalarga murakkabroq misol sifatida funksiyalarni keltirish mumkin.

9-misol.

```
<?php
Function foo()
{
Return 7;
}
?>
```

Bu yerda 'foo' ifodasining natijasi 7 ga teng.

\$x=100;-ushbu o'zgarmas ham ifodaga teng bo'ladi. Bu yerda \$x- o'zgaruvchi, 100- ifoda hisoblanadi. O'zgaruvchi ham ifoda vazifasini bajarishi mumkin. Misol:

```
$x=10; //10 ifoda
$y=$x; //$x-o'zgaruvchi ifoda.
```

PHPda amallar bajarish.

PHPda ham arifmetik amallardan foydalaniladi:

X+Y-qo'shish;
X-Y-ayirish;
X*Y-ko'paytirish;
X/Y-bo'lish;
X%Y-qoldiqli bo'lish;

Shuningdek qavslardan foydalaniladi. Arifmetik amallarga increment va decrement amallarini ham kiritish mumkin:

\$x++ - increment amali qiymati 1 ga oshiradi.

\$x-- - decrement amali qiymatni 1 ga kamaytiradi.

Funksiyalar.

PHP da funksiya dasturning istalgan qismida (<? va ?>oralig'ida) e'lon qilinishi mumkin. Yozilish qoidasi quyidagicha:

```
Function nomi (argumentlar ro'yxati)
{
Funksiya tanasi
```

}

Funksiya doimo function xizmatchi so'zi bilan beriladi. Undan so'ng funksiya nomi va qavs ichida argumentlar ro'yxati beriladi. Funksiyaga nom berishda quyidagi qoidalarga amal qilish lozim:

- Funksiya nomida kirilcha harflar qatnashishi mumkin, biroq tavsiya etilmaydi;
- Funksiya nomida bo'sh joy va bo'sh simvollar qatnashadi;
- Funksiya nomi unikal, ya'ni takrorlanmaydigan bo'lishi lozim. Masalan, F1() va f1() funksiyalari bitta funksiya hisoblanadi. Funksiyaga o'zgaruvchilardagi kabi nom berish mumkin bo'lib, faqat "\$" belgisi ishlatilmaydi.

Funksiya qaytaradigan qiymat istalgan ma'lumot turida bo'lishi mumkin. Funksiya natijasini asosiy dasturga uzatish uchun return ko'rsatmasidan foydalaniladi. Misol:

```
<?php
Function func(){
$number=888;
Return $number;
}
$a=func();
Echo $a;
?>
```

Berilgan misolda func funksiyasi return konstruksiyasi yordamida 888 ni qaytaradi. Bu qiymat \$a o'zgaruvchisiga taqdim etiladi, echo operatori \$a o'zgaruvchisining qiymatini brauzer oynasiga chiqaradi. Funksiyaga qiymat berish uchun uni e'lon qilish jarayonida qiymatlar argument sifatida ko'rsatiladi.

Misol:

```
<?php
Function fun($a, $b, ....., $z){
.....
```

```
}  
?>
```

Yoki funksiya qiymatini to'g'ridan to'g'ri berishga misol:

```
<?php  
Function ff ($string)  
{  
Echo"<h3>natija=$string</h3>";  
}  
$str=999;  
Ff(999);  
Ff($str);  
// 'ff' funksiyasi 'natija=999' ni ikki marta chiqaradi.  
?>
```

Misol:funksiya qiymatini ishorat shaklida berish.

```
<?php  
Function fun($string)  
{  
$string='bu funksiya ichidagi satr';  
}  
Fun($str);  
Echo $str;  
// natija 'bu funksiya ichidagi satr';  
?>
```

PHPda local o'zgaruvchilar faqat funksiya ichida, global o'zgaruvchilar butun dasturda ishlaydi. Global o'zgaruvchilarni barcha funksiyalarda ishlatish mumkin.

Misol:

```
<?php  
$i=0;  
$b=7;  
Function fl()
```

```

{
$x=5;
$i=0;
Global $b;
Echo"b=$b\n";
Echo"i=$i\n";
}
Fl();
Echo"x\n";
?>

```

\$i va \$b- global o'zgaruvchilar bo'lib, butun dasturda ishlatish mumkin. Global o'zgaruvchilarni funksiyaga taqdim etish uchun alohida ko'rsatmadan foydalaniladi.

Local va global o'zgaruvchilardan tashqari PHP da static o'zgaruvchilar ham mavjud. Agar funksiya tanasida static o'zgaruvchi e'lon qilingan bo'lsa, funksiya ishini yakunlagandan so'ng, kompilyator bu o'zgaruvchini o'chirib tashlamaydi.

Misol: PHPda static o'zgaruvchilar

```

<?php
Function fl()
{
Static $a;
$a++;
Echo"$a";
}
For($i=0;$i++<10)
Fl();
?>

```

Funksiyani rekursiyada ishlatish ham mumkin. Masalan, fact() funksiyasini biror sonning faktorialini hisoblasin:

```

<?php

```

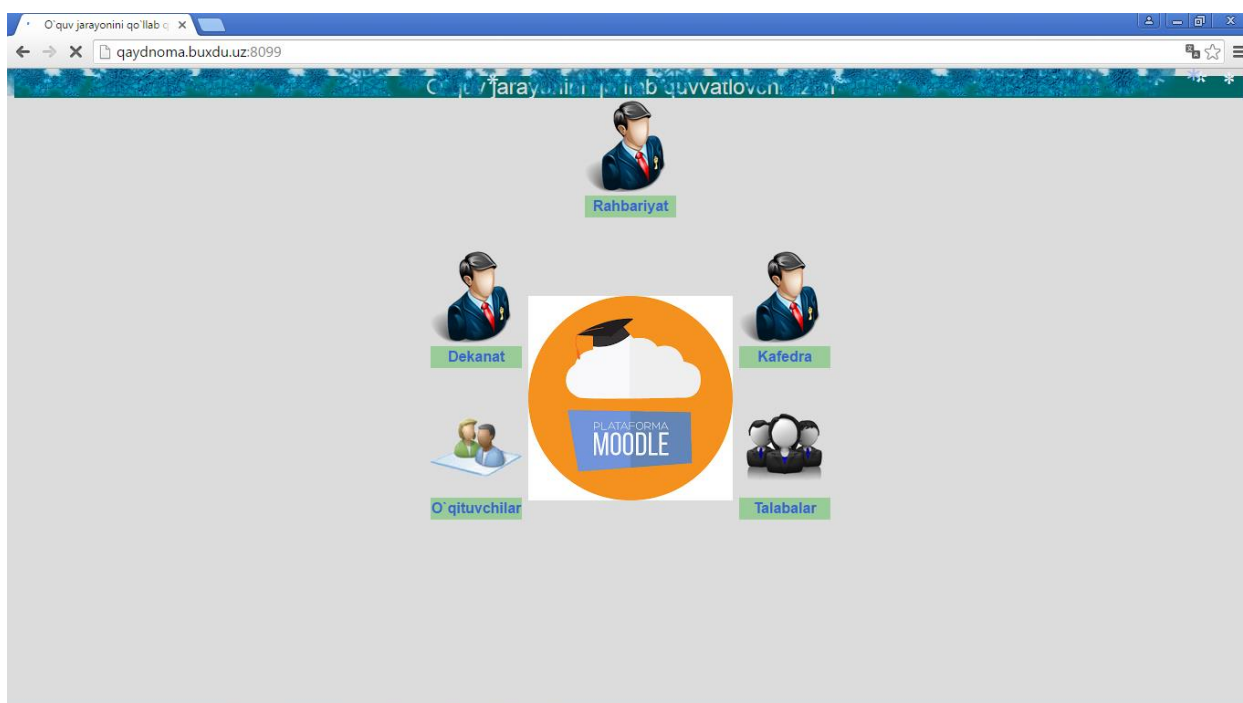
```

Function fact()
{
If($x==0) return 1;
Else return $x*fact($x-1);
}
Echo fact(7);
?>

```

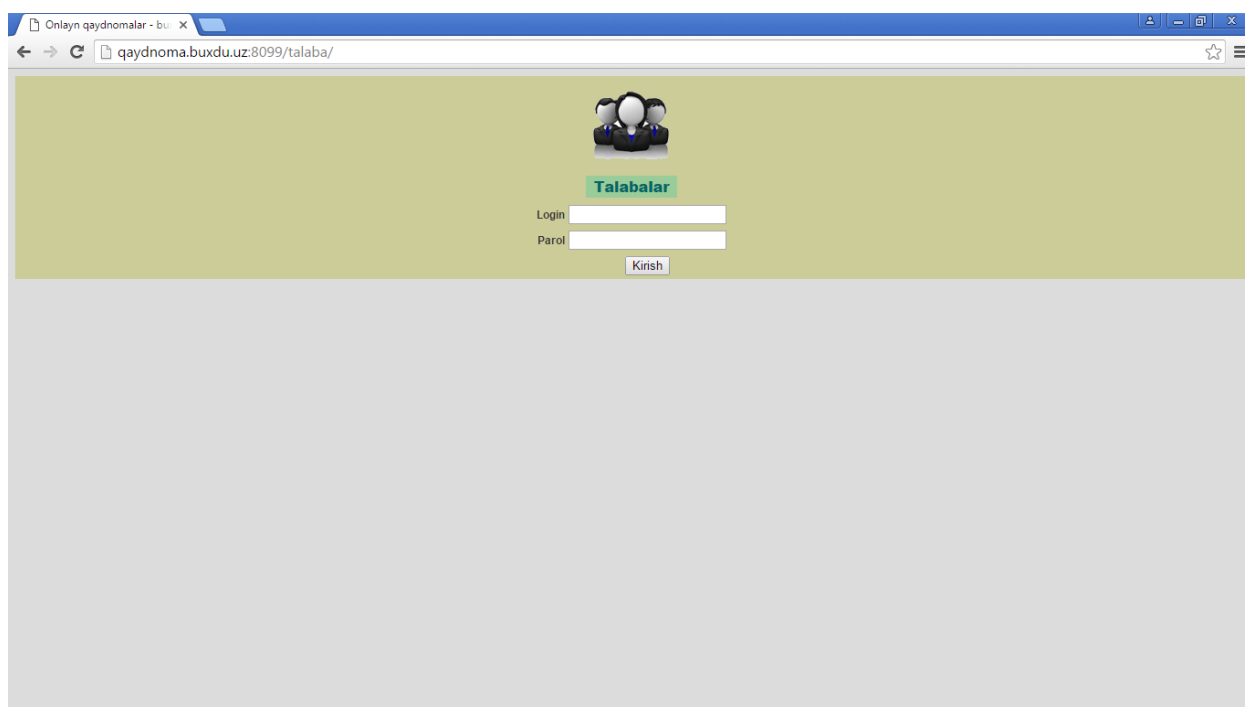
1.4. Universitet information tizimining funksional imkoniyatlari.

Buxoro davlat universitetining rasmiy sayti 2011-yildan boshlab yo'lga qo'yilgan. Undagi reyting tizimi esa 2013-yildan boshlab amalga oshirildi. Ushbu tizim orqali o'qituvchi va talabalar haqidagi umumiy ma'lumot keltirilgan. Buxoro davlat universitetida avtomatlashtirilgan axborot tizimi mavjud bo'lib u o'qituvchi va talabalar uchun xizmat qilmoqda. Bu tizimga kirish uchun qidiruv tizimiga "buxdu.uz" deb kiramiz. O'quv jarayonini qo'llab quvvatlovchi tizimning yana bir asosiy qimi bu reyting qaydnomalaridir. Bu tizimning afzalligi shundaki o'qituvchilar va talabalarda reyting qaydnomalariga baholarni kiritish va ulardan foydalanish hech qanday ortiqcha vaqt yoki ovvoragarchilikni talab qilmaydi.



1.3-chizma. Universitet o'quv jarayonini qo'llab quvvatlovchi dastur oynasi. Chunki o'qituvchilar reyting qaydnomalarini to'ldirish uchun berilgan vaqt mobaynida istalgan joydan turib va istalgan vaqtda kiritishlari mumkin. Talabalar ham istalgan vaqtda o'z ballarini tarmoqdan foydalangan holda bemalol ko'rishlari mumkin. Shuningdek bu tizimdan rahbariyat, dekanat, kafedra hodimlari foydalanadilar.

Bu tizimga kirish juda kuchli himoyalangan bo'lib, har bir xodim yoki o'qituvchi, talaba o'z login va parollari orqali kiradilar.



1.4-chizma. Login,parol formasi.

Ularning bajargan ishlari va bajarish vaqtlari esa registratsiya qilinib boriladi. Har bir xodim yoki o'qituvchi faqat o'ziga ruxsat etilgan amallarnigina bajara oladilar. Ular boshqa biror kishining imkoniyatlaridan foydalana olmaydilar.

Bu tizim shunday ishlanganki har bir xodim yoki o'qituvchi, talaba login parolini tersa har biriga boshqa xil oynalar chiqadi, ya'ni faqat o'ziga aloqador bo'lgan ma'lumotlarni ko'radi. Masalan talaba o'z login paroli bilan kirsam unga faqat o'z ma'lumotlari chiqadi. Bularga tanlangan semester davomida o'tilgan fanlar, dars jadvali va o'z shaxsiy parolini o'zgartirishlar kiradi. Talabalar o'z

sahifalaridan turib faqatgina o'z ma'lumotlarini ko'ra oladilar. Shaxsiy paroldan boshqa biror nimani o'zgartira olmaydilar. Shuning uchun endi o'qituvchi hech qanday muammolarsiz reyting ballarini qo'yishlari, talabalar esa o'z ballarini istalgan joydan turib ko'rishlari mumkin. Bu tizimda buzg'unchilik qilish, tizimni ishdan chiqarish imkoni deyarli yo'q. Shuning uchun bu tizim ishonchli va qulay hisoblanadi.

O'qituvchi oynasi.

O'qituvchi va xodimlar ham talabalardek login va parol bilan kiradilar. Faqat ular o'ziga daxldor biror nimani o'zgartirishi mumkin, shunisi bilan ularning oynasi talababalarining oynasidan farq qiladi. O'qituvchilar tizimga o'z login parollari bilan kirgach tizim shu o'qituvchiga tegishli oynani aniqlab ochib beradi. Bu oynaning yuqori qismida o'qituvchining ism sharifi chiqadi va unga tegishli bir nechta bandlar mavjud: baholash, o'qituvchi yuklamasi, dars jadvali, yangiliklar, xabarlar, pochta, forum va hokazolar.

Biz shulardan o'qituvchi baho qo'yish jarayonini ko'rib chiqamiz. O'qituvchi bu oynadan o'quv yili va semestri tanlaydi va u yerda shu o'qituvchi dars o'tadigan fanlar va guruhlar chiqadi. O'qituvchi bu yerdan kerakli guruh va fanni tanlagach nazorat turini (joriy, oraliq, yakuniy) tanlaydi. O'qituvchi kerakli talabaga olgan balini qo'ygach saqlash tugmasini bosadi va reyting ballari tizimidagi server xotirasiga saqlanadi. Agar o'qituvchi to'liq qaydnomani ya'ni shu guruh qaydnomasini to'liqligicha ko'rmoqchi bo'lsa "toliq" tugmasini bosadi va ekranga qaydnoma to'laligicha chiqadi.

O'qituvchiga tegishli ma'lumotlarni faqat shu o'qituvchining o'zi ko'ra oladi. Shu sababli bu tizim ishonchlidir. Tizim xavfsizligini ta'minlash uchun o'qituvchi parolini tez-tez yangilab turadi.

Xulosa.

I BOB.Foydalanilgan dasturiy ta'minot va texnologiyalar deb nomlanadi. Bu bobning 1.1 bo'limi Microsoft SQL server berilganlar bazasini boshqarish to'g'risida bo'lib,unda jadvallar yaratish va boshqa ma'lumotlar keltirilgan. 1.2 bo'limida SQL tili unda jadvallar yaratish, ularga ma'lumotlar kiritish, o'chirish,

ma'lumotlarni o'zgartirish haqida to'xtalib o'tilgan. 1.3 bo'limi HTML tili va PHP dasturlash tili haqida bo'lib, HTML tilida yozilgan Web sahifalar, html tilining asosiy teglari, html tilida ishlatiladigan identifikatorlar, matnlarni bezash, ro'yxatlar, jadvallar, formalar haqida umumiy ma'lumot berilgan. Shuningdek PHP dasturlash tili haqida, unda PHP dasturlash muhiti, PHP da o'zgaruvchi va konstantalar, PHPda tiplar, PHP da massivlar bilan ishlash, PHP da funksiyalar haqida keltirilgan va misollar orqali yoritilgan. 1.4 bo'lim Universitet information tizimining funksional imkoniyatlari haqida bo'lib, universitetning rasmiy sayti va reyting tizimi unda talabalarning fanlar bo'yicha olgan baholarini o'tiladigan fanlar o'qituvchilari tomonidan saytga joylashtirilishi, talabalar va ularning ota-onalari bemalol shu saytga kirib ko'rishi haqida yozilgan.

II BOB. Universitet informatsion tizimida iqtidorli talabalar bilan ishlash monitoringi modulini yaratish.

2.1.Modul spetsifikatsiyasi.

Universitet informatsion tizimida iqtidorli talabalar bilan ishlash o'z ichiga iqtidorli talabalarga biriktirilgan o'qituvchilar, rejalashtirilgan ishlar va bajarilgan ishlar ro'yxatini o'z ichiga oladi. Kafedra yig'ilishida kafedra mudiri tomonidan iqtidorli talabalar ro'yxati kafedra o'qituvchilari tomonidan ko'rib chiqiladi. Kafedra mudiri tomonidan iqtidorli talabalarga o'qituvchilar biriktirilib chiqiladi. Ushbu iqtidorli talabalar monitoringi web texnologiya orqali quyidagi holatda amalga oshiriladi. Bosh sahifa uchta asosiy bo'limdan iborat bo'ladi bular quyidagilar: Rahbar, Kafedra, O'qituvchi, Talaba.

Rahbar bo'limi. Rahbar bo'limida tizimga kirish uchun rahbar oynasida login va parol oynasi mavjud bo'ladi. Rahbar tizimga kirgandan so'ng yillik o'quv yiliga mos kafedralarda iqtidorli talabalar ro'yxatini ko'radi. Tanlangan o'quv yiliga mos yo'nalishlardagi o'sha o'quv yilida iqtidorli talabalarning biriktirilgan o'qituvchilarini ham ko'radi.

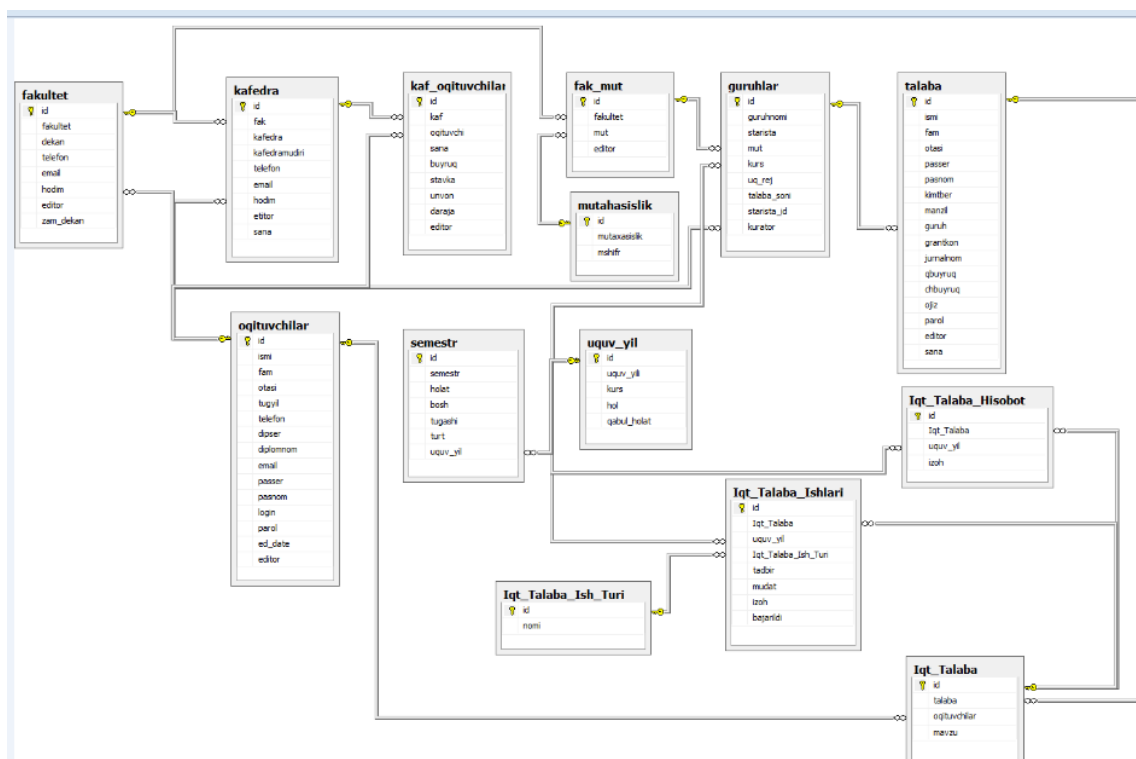
Kafedra bo'limi. Kafedra bo'limida ham tizimga kirish uchun login va parol oynasi bo'ladi. Ushbu bo'limda faqat kafedra mudirlari kirish huquqiga ega. Kafedra mudiri tizimga kirgandan so'ng kafedra mudiri tomonidan ishning turi, iqtidorli talaba va uning mavzusi, iqtidorli talabaga biriktirilgan o'qituvchilar kiritiladi. Kafedra mudiri oynasida iqtidorli talabalar ishlarining hisobotlari ham joylashtiriladi.

O'qituvchi bo'limi. Ushbu bo'limda har bir o'qituvchi Universitet informatsion tizimida mavjud login va paroli orqali tizimga kirish huquqiga ega bo'ladi, yani o'qituvchi bo'limida login parol oynasi mavjud bo'ladi. O'qituvchi tizimga kirgandan so'ng o'ziga biriktirilgan talabalardan birini tanlaydi. Undan oldin o'quv yilini tanlaydi. So'ngra talaba rejasi nazorati tugmasi yordamida yangi bo'limni hosil qilib bu bo'limda talabaning ishini nazorat qilib boradi. Yana o'qituvchi kerakli hisobotni oladi.

Talaba bo'limi. Ushbu bo'limda talaba o'zining bajaradigan ishini rejalashtiradi va bu ishning nomini kiritadi. Agar o'qituvchi bu rejalashtirilgan ishini nazorat qila boshlasa (izoh yozsa yoki bajargan bajarmaganligini belgilasa) buishni o'chirolmaydi ham o'zgartirolmaydi ham. Yana yangi ishni rejasini ham kiritadi.

2.2.ER modeli loyihasi.

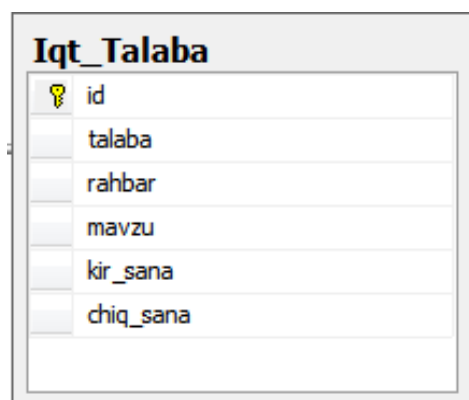
Universitet informatsion tizimida iqtidorli talabalar bilan ishlash monitoringi modulini yaratishda mavjud “uquv” ma'lumotlar bazasidagi jadvallarga qo'shimcha jadvallar yatomiz. Iqtidorli talabalar bilan ishlash monitoringi modulini yaratishda ER modeli quyida keltirilgan



2.1-chizma. ER model.

Iqtidorli talabalar bilan ishlash monitoringi modulini yaratishda “uquv” ma'lumotlar bazasidagi quyidagi jadvallardan foydalandik: fakultet, kafedra,

kafedra_oqituvchilari, fak_mut, mutahassislik, guruhlar, talaba, oqituvchilar, semestr, uquv_yili. Bu jadvallarga qo'shimcha tarzda quyidagi yangi jadvallarni yaratdik: Iqt_Talaba, Iqt_Talaba_Ishlari, Iqt_Talaba_Ish_Turi yuqoridagi ER modelda bu jadvallarning o'zaro bog'lanishi va ular tashkil topgan ustunlari bilan tasvirlangan. Bu jadvallar o'zaro birga ko'p va birga bir bog'langan.



Iqt_Talaba	
id	
talaba	
rahbar	
mavzu	
kir_sana	
chiq_sana	

2.2-chizma. Iqt_Talaba jadvali.

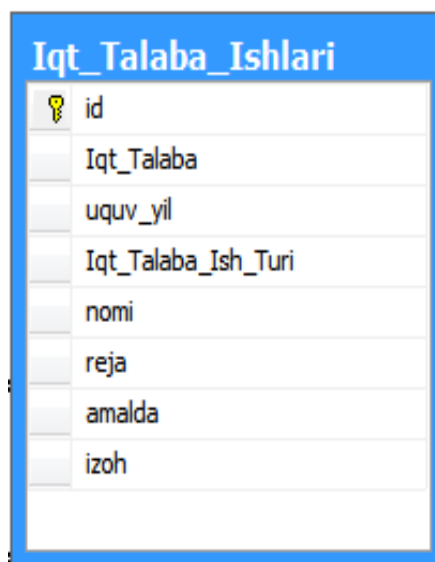
Iqt_Talaba jadvali yangi yaratilgan jadval bo'lib, bu jadvalda id, talaba, rahbar, mavzu kir_sana, chiq_sana kabi ustunlar mavjuddir. Id ustuni int tipida bo'lib, sanagich o'rnatilgan. Talaba va rahbar ustunlari numeric tipidadir. Talaba ustuni talaba jadvalining id ustuni bilan ko'pga bir bog'langan. varchar tipida mavzu ustuni bo'lib bunda iqtidorli talabaning mavzusi kiritiladi. Date tipida kir_sana va chiq_sana ustunlari bo'lib, ularda talabaning iqtidorli talabalar safiga kiritilgan va chiqarilgan sanalari saqlanadi.



Iqt_Talaba_Ish_Turi	
id	
nomi	

2.3-chizma. Iqt_Talaba_Ish_Turi jadvali.

Iqt_Talaba_Ish_Turi jadvali yangi yaratilgan jadval bo'lib, unung id, nomi ustunlari mavjud. Id ustuni int tipida bo'lib bu ustun Iqt_Talaba_Ishlari jadvalining turi ustuni bilan ko'pga bir bog'langandir. Ikkinchi nomi ustuni varchar tipida bo'lib unda iqtidorli talabalarning ishlarining turlari keltirilgan.



The image shows a screenshot of a database table structure for 'Iqt_Talaba_Ishlari'. The table has the following columns:

id
Iqt_Talaba
uquv_yil
Iqt_Talaba_Ish_Turi
nomi
reja
amalda
izoh

2.4-chizma.Iqt_Talaba_Ishlari jadvali.

Iqt_Talaba_Ishlari jadvali ham yangi yaratilgan jadvaldir. Bu jadvalda Iqt_Talaba ustuni numeric tipida bo'lib, bu ustun Iqt_Talaba jadvalining id ustuni bilan birga ko'p bog'langandir. Yana numeric tipida uquv_yili ustuni bo'lib, bu ustun semestr jadvalining uquv_yili ustuni bilan ko'pga ko'p bog'langan. Iqt_Talaba_Ish_Turi ustuni ham numeric tipida va u Iqt_Talaba_Ish_Turi jadvalining id ustuni bilan birga ko'p bog'langan. Varchar tipida nomi va izoh ustunlari bo'lib, nomi ustunida talabaning bajarayotgan ishining nomi saqlansa, izoh ustunida izohlar saqlanadi. Reja va amalda ustunlari date tipida bo'lib bu ustunlarda mos ravishda talaba bajarayotgan ishning rejalashtirilgan sanasi va amalda bajarilgan sanasi kiritiladi.

2.3.Foydalanuvchi grafik interfeysi va funksional imkoniyatlari.

Universitet informatsion tizimida iqtidorli talabalar bilan ishlash monitoringi modulini yaratishda quyidagi foydalanuvchi grafik interfeysi oynalari hosil bo'ladi.



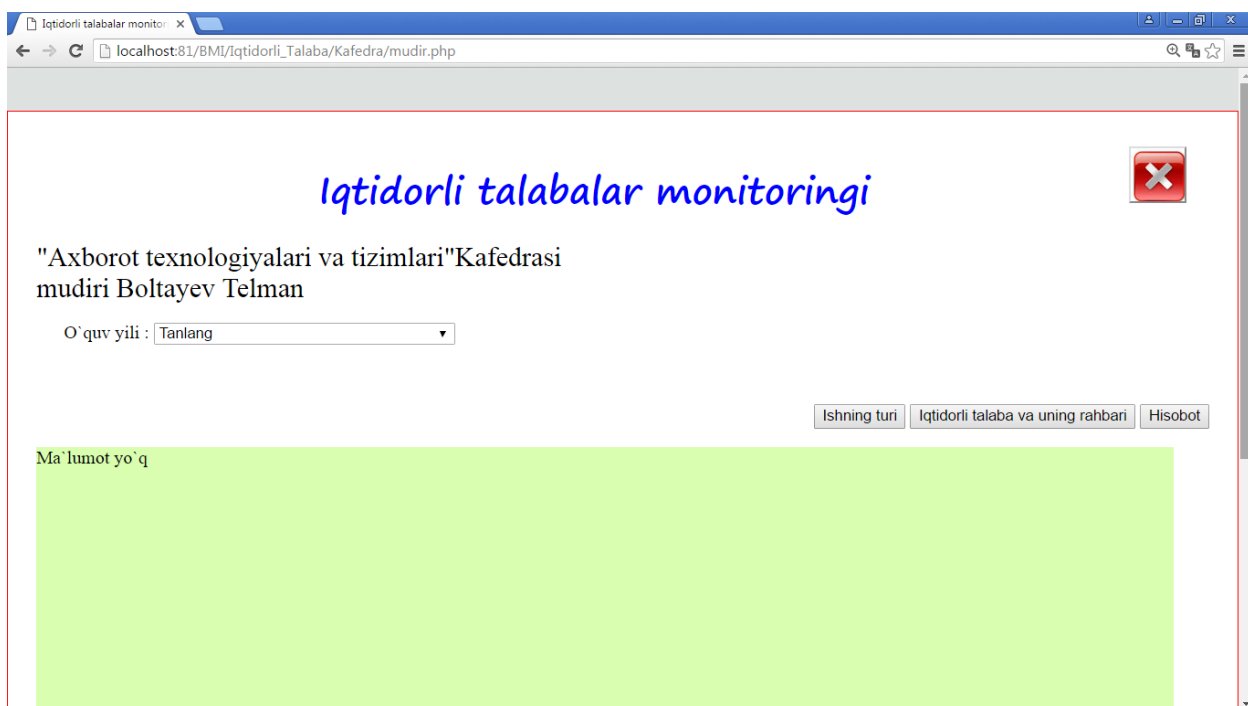
2.5-chizma.Iqtidorli talabalar bilan ishlash monitoringining asosiy oynasi.

Ushbu oynada 4 ta foydalanuvchilari mavjud bular: Rahbar, Kafedra, O'qtuvchi,talaba.Asosiy oynadan bir foydalanuvchi o'ziga tegishli bo'lgan bo'limni tanlaydi. Har bir foydalanuvchi bo'limni tanlagandan so'ng quyidagi oyna hosil bo'ladi:



2.6-chizma. Login parol oynasi.

Bu oyna login parol oynasi bo'lib bunda har bir foydalanuvchi o'ziga birlashtirilgan login va parolni ko'rsatilgan sohalarga kiritib dasturga kiradi va dastur bu foydalanuvchi qaysi turdagi foydalanuvchi ekanini avtomatik tarzda aniqlab unga mos oynani chiqarib beradi. Agar u Kafedra mudiri bo'lsa unda quydagicha oyna hosil bo'ladi:



2.7-chizma. Kafedra mudiri oynasi.

Ushbu oynadan kafedra mudiri foydalanadi oynaning yuqori chap qismida tanlagich qo'yilgan, tanlagichdan o'quv yili tanlaniladi. Undan so'ng oynada garizontal ravishda quyidagi uchta tugma joylashgan. 1 – tugma (ishning turi tugmasi) Ushbu tugma orqali ishning kiritilgan turlarini ko'rish imkoniyati mavjud va yangi ish turini kiritish imkoniyati ham mavjud bu holatni quyidagi oynada qaraymiz:



2.8-chizma. Kafedra mudiri oynasida yangi ish turini qo'shish.

2 – tugma (Iqtidorli talaba va unig rahbari tugmasi) Ushbu tugmada murojaat qilinganda tanlangan yildagi iqtidorli talabalar ro'yxatini ko'rish imkoniyati va yangi iqtidorli talabani qo'shish imkoniyati mavjud va u quyidagi ko'rinishda bo'ladi:

Iqtidorli talabalar monitoringi

"Axborot texnologiyalari va tizimlari"Kafedrası
mudiri Boltayev Telman

O'quv yili : 2014-2015

Ishning turi Iqtidorli talaba va uning rahbari Hisobot

Mutaxassislik	Guruh	FIO	Mavzu	Rahbar	O'chirish	O'zgartirish
5111018 Kasb ta'limi:informatika va axborot texnologiyalari	1-1PMI-12	AKMALOVA GULSANAM AKMAL QIZI	Integrallarni taqribiy hisoblash	Jalolov Ozod Isomiddinovich	O'chirish	O'zgartirish
5111018 Kasb ta'limi:informatika va axborot texnologiyalari	1-1PMI-12	FAYZIYEVA O'LMASOY ISTAM QIZI	Dasturiy ta'minot yaratish	Jalolov Ozod Isomiddinovich		O'zgartirish
5111018 Kasb ta'limi:informatika va axborot texnologiyalari	1-1PMI-12	FAYZIYEV MUHRIDDIN BAHRIIDDIN O'G'LI	Kurs ishi, referat monitoringi	Eshonqulov Hamza Ilhomovich	O'chirish	O'zgartirish

Yangi ma'lumot kiritish

2.9-chizma.Kafedra mudiri oynasida iqtidorli talaba va uning rahbari bo'limi.

3 – tugma (Hisobotlar tugmasi) Ushbu tugmaga murojaat qilinganda iqtidorli talabalarga doir hisobot bo'limi hosil bo'ladi u quyidagicha ko'rinishda:

Iqtidorli talabalar monitoringi

"Axborot texnologiyalari va tizimlari"Kafedrası
mudiri Boltayev Telman

O'quv yili : 2014-2015

Ishning turi Iqtidorli talaba va uning rahbari Hisobot

Iqtidorli talabalar

tanlangan Iqtidorli talabaning rejası

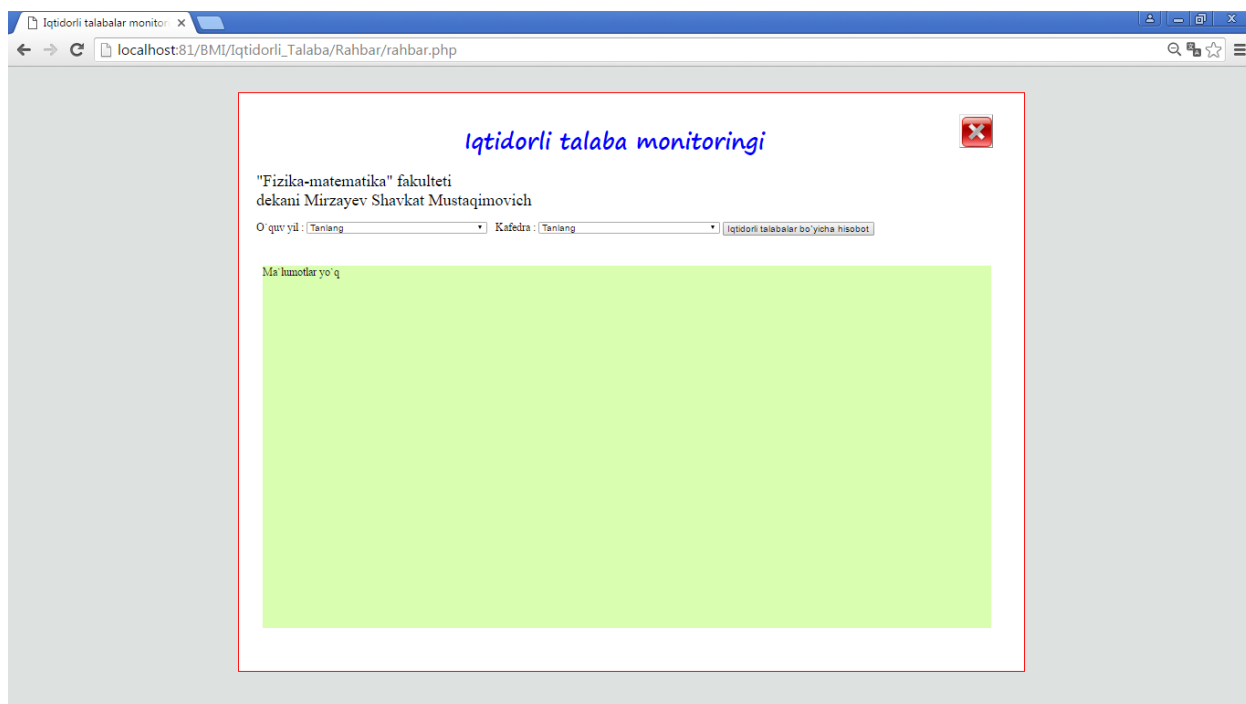
Mutaxassislik	Guruh	FIO	Mavzu	Rahbar	O'chirish	O'zgartirish
5111018 Kasb ta'limi:informatika va axborot texnologiyalari	1-1PMI-12	AKMALOVA GULSANAM AKMAL QIZI	Integrallarni taqribiy hisoblash	Jalolov Ozod Isomiddinovich	O'chirish	O'zgartirish
5111018 Kasb ta'limi:informatika va axborot texnologiyalari	1-1PMI-12	FAYZIYEVA O'LMASOY ISTAM QIZI	Dasturiy ta'minot yaratish	Jalolov Ozod Isomiddinovich		O'zgartirish
5111018 Kasb ta'limi:informatika va axborot texnologiyalari	1-1PMI-12	FAYZIYEV MUHRIDDIN BAHRIIDDIN O'G'LI	Kurs ishi, referat monitoringi	Eshonqulov Hamza Ilhomovich	O'chirish	O'zgartirish

2.10-chizma. Kafedra mudiri oynasida hisobotlar bo'limi.

Ushbu hisobotlar bo'limi kafedra mudiri uchun yaratilgan. U quyidagi axborotlarni kafedra mudiriga beradi:

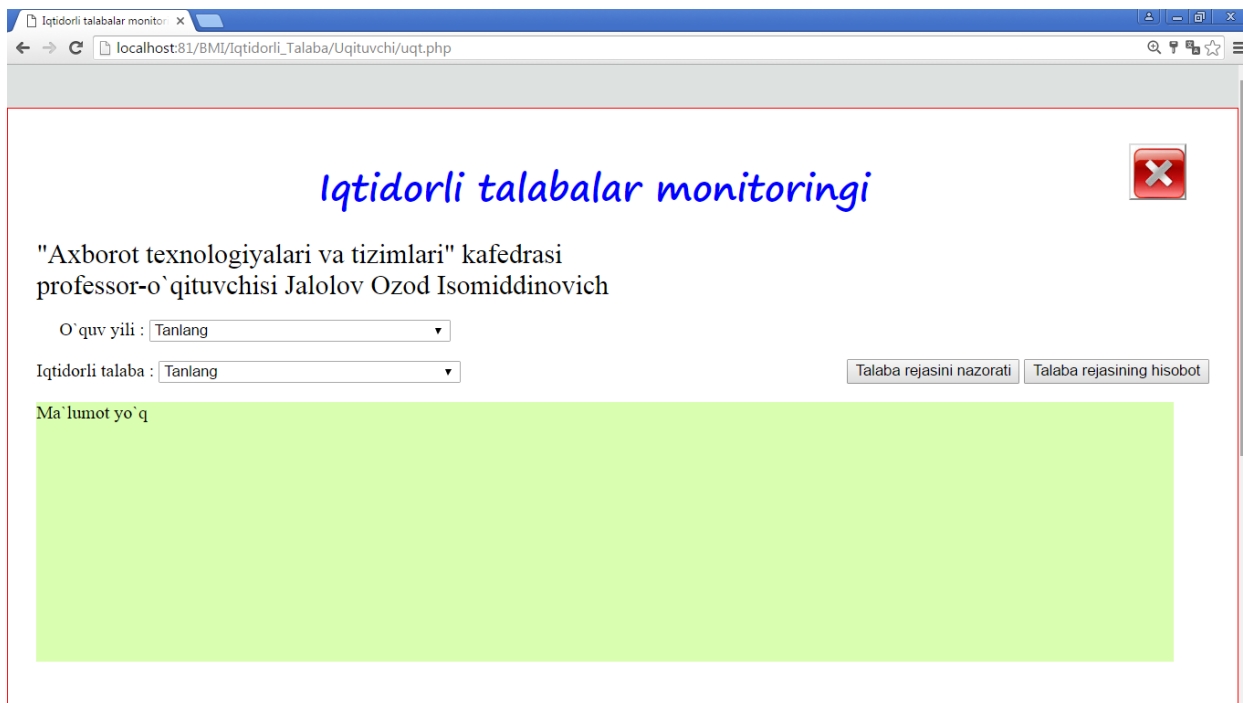
Tanlangan o'quv yillaridagi barcha iqtidorli talabalar ro'yxati haqidagi hisobotini tuzib beradi.

Tanlangan iqtidorli talabaning barcha rejalashtirilgan ishlarining ro'yxati va barcha bajarilgan ishlari ro'yxatining hisobotini tuzib beradi.



2.11-chizma. Rahbar oynasi.

Ushbu oyna barcha fakultetlarning dekanlari uchun bo'lib hisoblanadi. Joriy foydalanuvchi grafik interfeysida rahbar o'quv yili va kafedrani tanlaydi, tanlangan o'quv yiliga mos kafedraga tegishli iqtidorli talabalarning rejalashtirilgan ishlari ro'yxati va bajargan ishlari ro'yxati rahbarga hisobot tarzida hosil bo'ladi.



2.12-chizma. Iqtidorli talabalar bilan ishlash monitoringining o`qituvchi oynasi.

Ushbu oynadan iqtidorli talabalarning birlashtirilgan o`qituvchilari foydalanishadi. Ushbu oynada o`quv yili tanlangandan so`ng tanlangan o`quv yilida joriy o`qituvchiga birlashtirilgan iqtidorli talabalarning ro`yxatidan ikkinchi tanlagichdan kerakli iqtidorli talaba tanlanadi. Ushbu oynada gorizontal joylashgan 2ta tugma bo`lib, ulardan birinchisi (talaba rejasini nazorati tugmasi) bosilganda tanlangan iqtidorli talabaning barcha ishlarining rejalari ro`yxati hosil bo`ladi va bu ro`yxatdan o`qituvchi izoh va bajarilganligini kiritish imkoniyatiga ega bo`ladi va u quyidagicha ko`rinishda bo`ladi:



2.13-chizma. O'qituvchi oynasining talaba rejasi nazorati bo'limi.

2-tugma (Talaba rejasining hisoboti tugmasi) orqali tanlangan iqtidorli talabanning rejalashtirilgan barcha ishlarining ro'xatini va bajarilgan ishlarining ro'yxatining hisobotini olishi mumkin.

Iqtidorli talabalar monitoringi

"Fizika-matematika" fakulteti
5111018 - "Kasb ta'limi: informatika va axborot texnologiyalari" mutaxassisligi
"1-IPMI-12" guruhi
talabasi FAYZIYEVA O'LMASOY ISTAM QIZI
mavzusi : "Dasturiy ta'minot yaratish"
rahbari : Jalolov Ozod Isomiddinovich

O'quv yili : 2014-2015

Ish turi	Taddir nomi	Mudat	O'quv yili	O'chirish	O'zgartirish
Maqola	Loyiha dasturi	2016-02-17	2014-2015		
Ushbu qo'llanma	Java dasturlash tili	2016-03-16	2014-2015		

2.14-chizma. Iqtidorli talabalar bilan ishlash monitoringining talaba oynasi.

Ushbu oynada talabanning barcha rejalashtirilgan va bajargan ishlari ro'yxati bo'ladi. Bu ro'yxatga yangi ishning rejasini qo'shish imkoniyati bo'lib u quyidagicha ko'rinishda:

Iqtidorli talabalar monitoringi

"Fizika-matematika" fakulteti
5111018 - "Kasb ta'limi: informatika va axborot texnologiyalari" mutaxassisligi
"1-IPMI-12" guruhi
talabasi FAYZIYEVA O'LMASOY ISTAM QIZI
mavzusi : "Dasturiy ta'minot yaratish"
rahbari : Jalolov Ozod Isomiddinovich

O'quv yili : 2014-2015

Ish turi	Taddir nomi	Mudat	O'quv yili	O'chirish	O'zgartirish
Maqola	Loyiha dasturi	2016-02-17	2014-2015		
Ushbu qo'llanma	Java dasturlash tili	2016-03-16	2014-2015		

Ish turi :

Nomi :

Mudat :

2.15-chizma. Talaba oynasining yangi ma'lumot kiritish bo'limi.

2.4.Qo'yilgan masalaning dasturini amalga oshirish.

Ushbu oynada 4 ta foydalanuvchilari mavjud bular: Rahbar, Kafedra, O'qtuvchi,talaba.Asosiy oynadan bir foydalanuvchi o'ziga tegishli bo'lgan bo'limni tanlaydi. Har bir foydalanuvchi bo'limni tanlagandan so'ng boshqa oyna hosil bo'lad.

Iqtidorli talabalar bilan ishlash monitoringining asosiy oynasida 4 ta rol mavjud bo'lib bular: Rahbar, Kafedra, O'qtuvchi, talaba. Asosiy oynadan bir foydalanuvchi o'ziga tegishli bo'lgan bo'limni tanlaydi. Har bir foydalanuvchi bo'limni tanlagandan so'ng boshqa oyna hosil bo'lad. Bu oyna [2.5-chizmada](#) keltirilgan. Asosiy oynaning kodi quyidagicha:

```
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=Windows-1251" />
  <meta name="author" content="Ulmasoy" />
  <title>Iqtidorli talabalar monitoringi </title>
  <link rel="stylesheet" type="text/css" href="css/styles2.css" />
  <script language="JavaScript" type="text/javascript"
    src="js/jquery-2.1.1.min.js"> </script> </head>
<body> <div class="cont"> <div class="a"> <p align="center">
<font size="7" face="Segoe Print" color="blue" >
<b> Iqtidorli talabalar monitoringi </b> </font> </p>
<div class="rahbar"> 
<a href="Rahbar/"> <br />
<font size="5" face="Algerian"> Rahbar </font> </a> </div>
<div class="kafedra">  <br />
<a href="Kafedra/"> <font size="5" face="Algerian"> Kafedra </font></a></div>
<div class="uqituvchi">  <br />
<a href="Uqituvchi/"> <font size="5" face="Algerian"> O`qituvchi </font> </a>
</div> <div class="talaba">  <br />
<a href="Talaba/"><font size="5" face="Algerian"> Talaba</font> </a> </div>
</div> </div> </body> </html>
```

Kafedra foydalanuvchi grafik interfeysi [2.7-chizmada](#) keltirilgan bo'lib, bu oynaning kodi quyida keltirilgan:

```
<html>
<head>
```

```

<meta http-equiv="content-type" content="text/html; charset=Windows-1251" />
<meta name="author" content="XayatovXU" />
<title>Iqtidorli talabalar monitoringi</title>
<link rel="stylesheet" type="text/css" href="../css/styles.css" />
<script language="JavaScript" type="text/javascript"
src="../js/jquery-2.1.1.min.js"></script>
<script language="JavaScript" type="text/javascript">
$(document).ready(function () {
    $.ajax({
        type: 'POST',
        url: 'table/uqyil4.php',
        success: function (data) {
            $("#editor4 select[name='uqyil4']").html(data);} });
    $("#editor4 select[name='uqyil4']").change(function() {
var uy4 = $("#editor4 select[name='uqyil4'] option:selected").val();
    if (uy4=="0") { $("#insert1").addClass('hidden');
        $("#insert2").addClass('hidden');
        $("#editor1").addClass('hidden');
        $("#editor2").addClass('hidden');
        $("#form3").addClass('hidden'); }
    if (!$("#insert1").hasClass("hidden")) view1();
    if (!$("#insert2").hasClass("hidden")) view2();
    if (!$("#form3").hasClass("hidden"))
        document.getElementById(
            "view").innerHTML="Ma`lumot yo`q"; });
var s ="<option value=0>Tanlang</option>";
$("#editor2 select[name='itr2']").html(s);
$("#editor1 select[name='mut2']").html(s);
$("#editor2 select[name='gur2']").html(s);
$("#editor2 select[name='std2']").html(s);
    var kaf4 = $("#editor4 select[name='kaf4'] option:selected").val();
$.ajax({ type: 'POST', data: { kaf:kaf4 }, url: 'table/itr2.php', success: function
(data) {
    $("#editor2 select[name='itr2']").html(data);} });
var fak4 = $("#editor4 select[name='fak4'] option:selected").val();
$.ajax({ type: 'POST', data: { fak:fak4},url: 'table/mut2.php',success: function
(data) {
    $("#editor2 select[name='gur2']").html(s); $("#editor2
select[name='std2']").html(s);
    $("#editor2 select[name='mut2']").html(data);} });
$("#editor2 select[name='mut2']").change(function() {
var mut2 = $("#editor2 select[name='mut2'] option:selected").val();
if (!(mut2>0)) mut2 = 0
$.ajax({ type: 'POST',data: { mut:mut2},url: 'table/gur2.php', success: function
(data) {

```

```

    $("#editor2 select[name='std2']").html(s);
    $("#editor2 select[name='gur2']").html(data).change(); } }); });
$("#editor2 select[name='gur2']").change(function() {
    var gur2 = $("#editor2 select[name='gur2'] option:selected").val();
    if (!(gur2>0)) gur2 = 0 $.ajax({ type: 'POST', data: {gur:gur2}, url:
'table/std2.php',
    success: function (data) { $("#editor2
select[name='std2']").html(data).change();} }); });});
function edit2( Obj ) {
    $("#editor2 input[name='id2']:hidden").val(Obj);
    var c = []; c[0] = Obj;$.ajax({ type: "POST", data: "data="+JSON.stringify(c),
url: "table/read2.php", dataType: "json", success: function(data) {
    $("#editor2 select[name='mut2']").val(data.mut).trigger('change');
    $("#editor2 select[name='gur2']").val(data.gur).change();
    //$($("#editor2 select[name='gur2']").val(29);
    $("#editor2 select[name='std2']").val(data.std).change();
    $("#editor2 select[name='itr2']").val(data.itr);
    $("#editor2 input[name='mavzu2']").val(data.mavzu);
    $("#editor2").removeClass('hidden');} });}
function delete2( Obj ) {
    if (confirm("Ma`lumotnio`chiraymi?")) {
        $.ajax({
            type: "POST",
            data: {id:Obj },
            url: "table/delete2.php",
            success: function(data)
                { if (data>-1)
                    { view2();
                    if ($("#editor2 input[name='id2']:hidden").val()===Obj)
                        { view2();
                        $("#editor2 select[name='mut2']").val(0);
                        $("#editor2 select[name='gur2']").val(0);
                        $("#editor2 select[name='std2']").val(0);
                        $("#editor2 select[name='itr2']").val(0);
                        $("#editor2 input[name='id2']:hidden").val(0);
                        $("#editor2 input[name='mavzu2']").val("");
                        $("#editor2").get(0).reset();
                        $("#editor2").addClass('hidden');
                        }} }); } } function save2() {
    var std2 = $("#editor2 select[name='std2'] option:selected").val();
    var itr2 = $("#editor2 select[name='itr2'] option:selected").val();
    var id2 = $("#editor2 input[name='id2']:hidden").val();
    var mavzu2 = $("#editor2 input[name='mavzu2']").val();
    var kaf4 = $("#editor4 select[name='kaf4'] option:selected").val();
    mavzu2=mavzu2.trim();

```

```

if ( mavzu2.length>0 && (!std2=="") && std2>0
  && (!itr2=="") && itr2>0)
{
  if (id2==0) // insert
  {
    $.ajax({
      type: 'POST',
      data: {std:std2, itr:itr2, mavzu:mavzu2, kaf:kaf4},
      url: "table/insert2.php",
      success: function(data)
      {
        if (data>0) { view2();
          $("#editor2 select[name='mut2']").val(0);
          $("#editor2 select[name='gur2']").val(0);
          $("#editor2 select[name='std2']").val(0);
          $("#editor2 select[name='itr2']").val(0);
          $("#editor2 input[name='id2']:hidden").val(0);
          $("#editor2 input[name='mavzu2']").val("");
          $("#editor2").get(0).reset();
          $("#editor2").addClass('hidden');
        } } }); } else { $.ajax({ type: "POST",
      data: {id:id2, std:std2, itr:itr2, mavzu:mavzu2},
      url: "table/update2.php",
      success: function(data)
      {
        //alert(data);
        if (data>0)
        {
          view2(); $("#editor2 select[name='mut2']").val(0);
          $("#editor2 select[name='gur2']").val(0); $("#editor2
select[name='std2']").val(0); $("#editor2 select[name='itr2']").val(0);
          $("#editor2 input[name='id2']:hidden").val(0); $("#editor2
input[name='mavzu2']").val(""); $("#editor2").get(0).reset();
          $("#editor2").addClass('hidden'); } } }); } } else alert(" Kerakli ma`lumotlar
yetarli emas ");
} function view2() { var kaf4 = $("#editor4 select[name='kaf4']
option:selected").val();
$.ajax({ type: 'POST', data: {kaf:kaf4}, url: 'table/view2.php',
success: function (data) { document.getElementById(
"view").innerHTML=data; } }); }function edit1( Obj ) {
$("#editor1 input[name='id1']:hidden").val(Obj);
$.ajax({ type: "POST", data: {id : Obj}, url: "table/read1.php",
success: function(data) { $("#editor1 input[name='nom1']:text").val(data);
} }); $("#editor1").removeClass('hidden'); }
function delete1( Obj ) { if (confirm("Ma`lumotnio`chiraymi?")) {
$.ajax({ type: "POST", data: {id:Obj }, url: "table/delete1.php",
success: function(data) { if (data>-1) { view1();
if ($("#editor1 input[name='id1']:hidden").val()===Obj)

```

```

    { $("#editor1 input[name='id1']:hidden").val('0');
      $("#editor1 input[name='nom1']:text").val("");
      $("#editor1").get(0).reset(); $("#editor1").addClass('hidden');
    } } })); } } function save1() {
var nom1 = $("#editor1 input[name='nom1']").val();
var id1 = $("#editor1 input[name='id1']:hidden").val();
if ( nom1.trim().length>0) { if (id1==0) // insert
  { $.ajax({ type: 'POST', data: {nomi:nom1}, url: "table/insert1.php",
    success: function(data) { if (data>0) {
view1(); $("#editor1").get(0).reset();
$("#nom1").val(""); $("#editor1").addClass('hidden');
} } }); } else //update { $.ajax({ type: "POST", data:
{id:id1,nomi:nom1}, url: "table/update1.php",
success: function(data) { if (data>0)
  { $("#editor1").get(0).reset(); $("#nom1").val(0);
$("#id1").val(""); $("#editor1").addClass('hidden');
view1(); } } }); } } else alert(" O`quv yilini tanlang ");
} function view1() { $.ajax({ type: 'POST', url: 'table/view1.php',
success: function (data) { document.getElementById(
"view").innerHTML=data; } });}
function T1() { if( $("#uqyil4").val() && $("#uqyil4").val() !=0 )
{ $("#insert1").removeClass('hidden'); $("#insert2").addClass('hidden');
$("#editor1").addClass('hidden'); $("#editor2").addClass('hidden');
$("#form3").addClass('hidden'); $("#id1").val(0);
$("#id2").val(0); $("#view").css("height","6cm");
view1(); } else alert("O`quv yilini tanlang");
} function T2() { if( $("#uqyil4").val() && $("#uqyil4").val() !=0) {
$("#insert1").addClass('hidden'); $("#insert2").removeClass('hidden');
$("#editor1").addClass('hidden'); $("#editor2").addClass('hidden');
$("#form3").addClass('hidden'); $("#id1").val(0);
$("#id2").val(0); $("#view").css("height","6cm");
view2(); } else alert("O`quv yilini tanlang");
} function T3() { if( $("#uqyil4").val() && $("#uqyil4").val() !=0 ) {
$("#insert1").addClass('hidden'); $("#insert2").addClass('hidden');
$("#editor1").addClass('hidden'); $("#editor2").addClass('hidden');
$("#form3").removeClass('hidden'); $("#id1").val(0);
$("#id2").val(0); $("#view").css("height","11cm");
} else alert("O`quv yilini tanlang");
} function insert1() { if($("#uqyil4").val() && $("#uqyil4").val() !=0 ) {
$("#editor1").removeClass('hidden'); $("#editor2").addClass('hidden');
$("#form3").addClass('hidden'); $("#id1").val(0);
$("#editor1").get(0).reset(); $("#id2").val(0);
$("#editor2").get(0).reset(); $("#form3").get(0).reset();
} else alert("O`quv yilini tanlang"); } function insert2() {
if($("#uqyil4").val() && $("#uqyil4").val() !=0 {

```



```

<p> Iqtidorli talabaning ish turi : <input type="text" name="nom1" id="nom1"
/><br />
<p><input type="button" name="save" id="save" value="Saqlash"
onclick="save1();" />
<input type="button" name="cancel" id="cancel" value="Inkor"
onclick="cancel1();" />
</form> <input type="button" name="insert2" id="insert2" value="Yangi
ma`lumot kiritish"
onclick="insert2()" class="hidden" /> <p>
<form name="editor2" id="editor2" class="hidden">
<input type="hidden" name="id2" id="id2" value="0" class="hidden">
<pre>Mutaxassislik : <select name="mut2" id="mut2"><option
value=0>Tanlang</option></select>
    Guruh : <select name="gur2" id="gur2"><option
value=0>Tanlang</option></select>
    talaba : <select name="std2" id="std2"><option
value=0>Tanlang</option></select></pre>
Rahbar :
<select name="itr2" id="itr2"><option value=0>Tanlang</option></select><br>
Mavzu :
<input type="text" name="mavzu2" id="mavzu2" />
<p><input type="button" name="save" id="save" value="Saqlash"
onclick="save2();" />
<input type="button" name="cancel" id="cancel" value="Inkor"
onclick="cancel2();" />
</form></div></div>
</body>
</html>

```

O'qituvchi oynasi 2.12-chizmada keltirilgan bo'lib uning kodi quyida keltirilgan:

```

<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=Windows-1251" />
<meta name="author" content="XayatovXU" />
<title>Iqtidorli talabalar monitoringi</title>
<link rel="stylesheet" type="text/css" href=" ../css/styles.css" />
<script language="JavaScript" type="text/javascript"
src=" ../js/jquery-2.1.1.min.js"></script>
<script language="JavaScript" type="text/javascript">
$(document).ready(function () {
    $.ajax({
        type: 'POST',
        url: 'table/uqyil4.php',
        success: function (data) {

```

```

    $("#editor4 select[name='uqyil4']").html(data);
    } }); //e4 ko_id4
var ko_id4 = $("#editor4 select[name='ko_id4'] option:selected").val();
$.ajax({ type: 'POST', data: {ko_id:ko_id4}, url: 'table/it4.php',
    success: function (data) {
        $("#editor4 select[name='it4']").html(data);
    } });var s = "<option value=0>tanlang</option>";
$("#editor4 select[name='uqyil4']").change(function() { a(); });
$("#editor4 select[name='it4']").change(function() {
    a();});});function a() { cancel1();
document.getElementById("view").innerHTML="Ma`lumot yo`q2";
} function save1() {
var id1 = $("#editor1 input[name='id1']:hidden").val();
var baj1 = $("#editor1 input[name='baj1']").prop("checked");
if (baj1==false) baj1 = 0; else baj1 = 1;
var izoh1 = $("#editor1 input[name='izoh1']").val();
izoh1 = izoh1.trim(); if ( izoh1.length>0 ) {
    if (id1>0) { $.ajax({
        type: "POST", data: {baj:baj1, izoh:izoh1,id:id1}, url: "table/update1.php",
        success: function(data) { if (data>0){$("#editor1
input[name='id1']:hidden").val(0);
        $("#editor1 input[name='izoh1']").val("");
        $("#editor1 input[name='baj1']").prop("checked",false);
        $("#editor1").get(0).reset();
        $("#editor1").addClass('hidden');
        view1(); } } }); }
    } else alert("Ma`lumot hammasi berilmagan "); }
function edit1( Obj ) {
    $("#editor1 input[name='id1']:hidden").val(Obj);
    var c = [];c[0] = Obj;$.ajax({
    type: "POST", data: "data="+JSON.stringify(c),
    url: "table/read1.php", dataType: "json",
    success: function(data) { $("#editor1 input[name='izoh1']").val(data.izoh);
        if (data.baj==0 || data.baj==null)
            $("#editor1 input[name='baj1']").prop("checked",false);
        else $("#editor1 input[name='baj1']").prop("checked",true);
        $("#editor1").removeClass('hidden');
    } });}
function view1() { var uy4 = $("#editor4 select[name='uqyil4']
option:selected").val();
var it4 = $("#editor4 select[name='it4'] option:selected").val();
$.ajax({ type: 'POST', data: {uy:uy4, it_id:it4 },
url: 'table/view1.php', success: function (data) {
    document.getElementById( "view").innerHTML=data;
} });} function T1() {

```


Xotima.

Bitiruv malakaviy ishi 2 ta bobdan iborat bo'lib, birinchi bob foydalanilgan dasturiy ta'minot va texnologiyalay deb nomlanadi. Bu bobning 1.1-bo'limi SQL server 2008 dasturi va uning imkoniyatlari haqida bo'lib, unda jadvallar yaratish va boshqa ma'lumotlar keltirilgan. 1.2-bo'lim SQL tili va unda jadvallar yaratish, ularga ma'lumotlar kiritish to'g'risida. 1.3-bo'lim HTML tili va PHP dasturlash tili haqida bo'lib, bu bo'limda ikkala til haqida qisqacha va boshlang'ich ma'lumotlar keltirib o'tilgan. 1.4-bo'lim universitet informatsion tizimining funksional imkoniyatlari haqida bo'lib, unda universitetning rasmiy sayti va reyting tizimi haqida yozilgan. Ikkinchi bob universitet informatsion tizimida iqtidorli talabalar bilan ishlash monitoringi modulini yaratish deb nomlangan bob jami 4 ta bo'limdan iborat bo'lib, ular 2.1- Modul spetsifikatsiyasi, 2.2- ER modeli loyihasi, 2.3- Foydalanuvchi grafik interfeysi, 2.4- Qo'yilgan masalaning dasturini amalga oshirish deb nomlanadi va mavzuning amaliy qismini o'zida jamlagandir. PHP da dasturlar tuzilib, PHP dasturlash tili ma'lumotlarini saqlashda MS SQL Serverda ma'lumotlar bazasi yaratildi. Berilganlar bazasidagi jadvallarni vazifalari tafsiloti keltirildi.

PHP web dasturlash tili va SQL server berilganlar bazasini boshqarish tizimi yordamida men yaratgan universitet informatsion tizimida iqtidorli talabalar bilan ishlash monitoringi modulini yaratish dasturi oliy ta'lim muassasalarida iqtidorli talabalarning ilmiy ishlarini monitoring qilish va ularning hisobotlarini olish mumkin bo'ladi. Kafedra mudiri iqtidorli talabaning ism sharifi, ish turi, birlashtirilgan o'qituvchisini hamda mavzusini kiritadi. Talaba esa bajaradigan ishining nomi, rejalashtirilgan sanasi, ish turini kiritadi. Fakultet dekanlari belgilangan o'quv yili va kafedralarning iqtidorli talabalari haqida hisobotlarni olishlari mumkin. Shuningdek kafedra mudirlari ham belgilangan o'quv yili uchun iqtidorli talabalarning bajargan ishlari va rejalashtirilgan ishlari haqidagi hisobotlarni olishlari mumkin.

Adabiyotlar.

1. I.A. Karimov “Yuksak ma’naviyat yengilmas kuch”. Toshkent. 2008.
2. Sharipov . M. S, Erjanov. X. D. “HTML kitob”. Urganch 2008y.
3. Sh. Q. Shoyqulov “Web dasturlash”. “Qarshi davlat universiteti” nashriyoti. Qarshi 2013.
4. N. S. Sayidova “ Web texnologiyalar” o’quv uslubiy qo’llanma.
5. Sayt MoodleDocs- <http://docs.moodle.org/ru>
6. Sayt <http://dasturchi.uz>