

O'ZBEKISTON RESPUBLIKASI OLIY VA O'RTA MAXSUS

TA'LIM VAZIRLIGI

BUXORO DAVLAT UNIVERSITETI

Fizika–matematika fakulteti

“Axborot texnologiyalari” kafedrası

**BAXSHILLOYEVA MUKARRAM SAYFULLO
QIZI**

Onlayn buyurtmalar qabul qilish tizimini yaratish

“5111000- Kasb ta`limi(5330200- Informatika va axborot texnologiyalari)”ta`lim
yo`nalishi bo`yicha bakalavr

darajasini olish uchun

BITIRUV MALAKAVIY ISHI

“Ish ko`rildi va himoya qilishga

Ilmiy rahbar _____ o`qit. F.F.Qosimov

ruxsat berildi”

“ _____ ” _____ 2017 y.

Kafedra mudiri

Taqrizchi _____ kat.o`qit.U.Narziyev

_____ dots.T.B.Boltayev

“ _____ ” _____ 2017 y.

“ _____ ” _____ 2017 y.

“Himoya qilishga ruxsat berildi”

Fakultet dekani _____ SH.M.Mirzayev

“ _____ ” _____ 2017y.

Buxoro-2017

Mundarija

KIRISH	3
I. BOB. ONLAYN TIZIMLAR BO'YICHA NAZARIY BILIMLARNI TO'PLASH VA O'RGANISH.....	8
1.1PHP va MySQLda bog`lanishlar.	8
1.2. Jquery va Ajax texnologiyalari.....	23
Xulosa	39
II BOB. ONLAYN BUYURTMA OLISH TIZIMINI TASHKIL ETISH JARAYONI	40
II.2. Onlayn buyurtma olish tizimining mijoz qismini shakllantirish	49
II.3. Onlayn buyurtma tizimida hisobotlarni shakllantirish	54
Xulosa	68
Xotima	69
Foydalanilgan adabiyotlar	70

KIRISH

O'zbekiston Respublikasi mustaqil davlat sifatida istiqloq yo'lidan ildamlab bormoqda. Mustaqillik tufayli jamiyatimiz hayotida bo'lgani kabi ta'lim tizimida ham tub o'zgarishlar ro'y berdi. Ayniqsa, «Ta'lim to'g'risida»gi qonun va «Kadrlar tayyorlash milliy dasturi»ning qabul qilinishi respublika ta'lim tizimida muhim voqea hisoblanadi. Bu hujjatlarda Respublika ta'lim muassasalarida yuqori malakali, bilimdon kadrlar tayyorlash usul va yo'llari, bosqichlari aniq belgilab berilgan va ular ta'lim sohasida amalga oshiriladigan islohotlar dasturi hisoblanadi.

O'zbekiston Respublikasi Prezidenti I.A. Karimov aytganlaridek: «Vatanimizning kelajagi, xalqimizning ertangi kuni, mamlakatimizning jahon hamjamiyatidagi obro'-e'tibori avvalambor, farzandlarimizning unib-o'sib, ulg'ayib, qanday inson bo'lib hayotga kirib borishiga bog'liqdir. Biz bunday o'tkir haqiqatni hech qachon unutmasligimiz kerak».

Shunday ekan, O'zbekiston Respublikasi taraqqiyotining muhim sharti xalqning boy intellektual merosi va umumbashariy qadriyatlar asosida, zamonaviy madaniyat, iqtisodiyot, fan, texnika va texnologiyalarning yutuqlari asosida kadrlar tayyorlashning mukammal tizimini shakllantirishdir.

Kadrlar tayyorlash milliy dasturining maqsadi ta'lim sohasini tubdan isloh qilish, uni yaqin o'tmishdan qolgan mafkuraviy qarashlar va sarqitlardan to'la xalos etish, rivojlangan demokratik davlatlar darajasida, yuksak ma'naviy va axloqiy talablarga javob beruvchi yuqori malakali kadrlar tayyorlash Milliy tizimini yaratishdir. Kadrlar tayyorlash milliy dasturi milliy tajribaning tahlili va ta'lim tizimidagi jahon miqiyosidagi yutuqlar asosida tayyorlangan hamda yuksak umumiy va kasb-hunar madaniyatiga, ijodiy va ijtimoiy faolikka, ijtimoiy-siyosiy hayotda mustaqil ravishda mo'ljalni to'g'ri o'qish mahoratiga ega bo'lgan, istiqbol vazifalarini ilgari surish va hal etishga qodir kadrlarning yangi avlodini shakllantirishga yo'naltirilgandir.

XXI asrning intellektual yoshlar asri, axborot texnologiyalari asri deb atalgani sababli, shu davr mobaynida axborotlar bilan ishlash va axborotlashtirish juda rivojlandi. Bunga sabab shundaki, kundalik turmushda, axborotlar, ularni qayta

ishlash va uzatishning ahamiyati tobora ortib borishidadir. Bu esa o'z navbatida jamiyatning har bir a'zosidan axborotlashtirish va axborot texnologiyalari sirlarini, uning qoida va qonuniyatlarini mukammal bilishni taqozo etadi. Fan va texnologiyalarning rivojlanishi, taraqqiy etishi va o'zaro integratsiyalashuvi jadallashmoqda, shu bilan birga yangi innovatsion texnologiyalar yaratish hamda ishlab chiqarishning turli jabhalarida qo'llanishi har bir yoshni o'z mutaxassisligini mukammal egallashini taqozo etadi.

Ta'lim-tarbiya jarayonida zamonaviy axborot texnologiyalarni joriy etish, ta'limni kompyuterlashtirish muammolarini hal qilish muhim ahamiyat kasb etadi. Bu muammoga hukumatimiz tomonidan alohida e'tibor bilan qaralmoqda. Respublikamiz Prezidenti I.A.Karimovning 2001-yil Oliy Majlisning 5-sessiyasida so'zlagan nutqida axborot texnologiyalari va kompyuterlarni jamiyat hayotiga kishilarning turmush tarziga, maktab va oliy o'quv yurtlariga jadallik bilan olib kirish g'oyasi ilgari surilgan edi.

Kompyuterning ta'lim-tarbiya sohasidagi ahamiyati beqiyos. U xalq ta'limi tizimini ma'muriy boshqarishdan tortib maktab faoliyatini alohida tashkil etish, boshqarish nazorat qilishgacha, o'quv predmetlarini o'rganishni tashkil qilishdan tortib, o'quvchilarning individual mashg'ulotlarini tashkil etishgacha bo'lgan muammolarni qamrab oladi. Bu muammolarni hal qilish bo'yicha barcha iqtisodiy rivojlangan mamlakatlarda, jumladan, bizning Respublikada ham turli yo'nalishdagi tadqiqot ishlari o'tkazilmoqda.

Ta'lim sohasida kompyuterlashtirish, unga axborot texnologiyalarni joriy qilish bilan bog'liq muammolarni tadqiq qilishda muayyan yutuqlarga erishilgan bo'lsada, bu sohada yechimini kutayotgan masalalar ham talaygina. Bu masalalar qatoriga umumiy o'rta ta'limda kompyuterli ta'lim metodologiyasining o'rnatilmaganligi, pedagogik dasturiy vositalari (PDV) ni yaratish prinsipi, vosita bosqichlari va mexanizmining yetarlicha tadqiq qilinmaganligi; o'quv materialiga pedagogik dasturiy vositaga mo'ljallab ishlov berish, ma'lumotlar omboriga joylashtirish usullari, ulardan foydalanish tizimining ilmiy asoslanganligi, dars jarayonidagi bosqichlarni to'liq kompyuterlashtirish masalalasining majmuaviy

o'rganilmaganligi kabilar kiradi. Respublikamizning pedagogik olim va amaliyotchilari ilmiy asoslangan hamda O'zbekistonning ijtimoiy-pedagogik sharoitiga moslashgan ta'lim texnologiyalarini yaratish va ularni ta'lim-tarbiya amaliyotida qo'llashga intilmoqdalar.

Yurtboshimiz I.A.Karimov "XXI asr axborot texnologiyalari asri" – deb ta'kidlaganlaridek, butun jahon taraqqiyoti tobora axborot texnologiyalari bilan tez suratda rivojlanib bormoqda. Xususan biz kabi yoshlarni zamon talabiga javob beruvchi mutaxassislar qilib tayyorlashga e'tibor katta.

Mavzuning dolzarbligi. Onlayn buyurtmalar qabul qilish tizimi hozirgi kundagi axborot texnologiyalari sohasida xizmat ko'rsatayotgan ko'plab jabhalarning muammolarini hal qilib berish imkoniyatini beradi. Deyarli barcha sohalar jadal ravishda avtomatlashish jarayonidan o'tib kelmoqda. Demak, har bir sohada onlayn xizmatlarni yo'lga qo'yish va zamon bilan hamnafas ildamlanish dolzarb vazifa sifatida qaralmoqda. Shundan kelib chiqqan holda, onlayn buyurtmalarni qabul qilish tizimini tashkil etish bizning rivojlanayotgan yurtimizda dolzarb mavzulardan hisoblanadi.

Bitiruv malakaviy ishimizning maqsadi. Ishlab chiqarish, xizmat ko'rsatish sohalarida avtomatlashtirilgan onlayn buyurtma qabul qilish tizimini ishlab chiqish ishimizning asosiy maqsadi hisoblanadi.

Bitiruv malakaviy ishimizning vazifalari. Bitiruv malakaviy ishining maqsadidan kelib chiqqan holda oldimizga quyidagi vazifalarni belgilab oldik:

1. Web texnologiyalarning yangi ko'rinishlarini o'rganish, bilimlar to'plash.
2. To'plangan nazariy bilimlarni yagona tizimga keltirish.
3. Chet elda qo'llanilayotgan nomdor onlayn buyurtma tizimlarini o'rganish, ularning yutuq va kamchiliklarini aniqlash.
4. Bizning respublikamizdagi islohotlar, mentalitetimiz va kichik biznesni tashkil etish strukturalariga asoslanib onlayn buyurtma qabul qilish tizimini tashkil etish.

Bitiruvmalakaviy ishimizning o'rganilganlik darajasi. Onlayn buyurtma qabul qilish hozirda endi rivoj topayotgan tizimlar sirasiga kiradi. Shuning uchun

uni tashkil etish va rivojlantirish doimiy yangilik va o'rganishni taqozo etadi. Bu esa o'z navbatida o'rganilganlik darajasini aniqlab beradi.

Bitiruvmalakaviy ishimizning predmeti. Onlayn buyurtma qabul qilish tizimlari alohida mavqega ega sanaladi. Ushbu tizimini yaratishda web texnologiyalardan foydalanish, MySQL ma'lumotlar omborini qo'llash bitiruv malakaviy ishimizning predmeti sanaladi.

Bitiruvmalakaviy ishimizning ob'ekti. Onlayn buyurtma qabul qilish tizimi zamonamizning jadal rivojlanayotgan tizimlaridan biri. Shu tizimni yanada rivojlantirish, undagi muammolarni qisman hal etish bitiruv malakaviy ishimizning ob'yekti hisoblanadi.

Bitiruv malakaviy ishimizning ilmiy farazi. Onlayn buyurtmalarni qabul qilish tizimi orqali aholini axborot texnologiyalari bo'yicha savodining oshishi, biror narsa yoki xizmat turini xarid qilishda vaqtning tejalishi imkonini beradi. Ushbu tizim orqali yangi ish o'rinlari yaratilishi mumkin.

Bitiruv malakaviy ishimizning yangiligi. Onlayn buyurtma tizimi yurtimizda yangi rivojlanayotgan sohalardan biri sanaladi. Ushbu sohada yetarli ishlar qilinmaganligi bitiruv malakaviy ishimizning yangiligini belgilab beradi.

Bitiruv malakaviy ishimizning metodologik asosi. O'zbekiston Respublikasi "Ta'lim to'g'risida"gi Qonuni, "Kadrlar tayyorlash milliy dasturi", Prezidentimiz I.A.Karimovning ta'lim, ilm-fan, axborot texnologiyalari to'g'risidagi fikrlari, internet tizimidagi yetakchi ilm-fan jurnallari, veb-saytlar bitiruv malakaviy ishimizning metodologik asosi sanaladi.

Bitiruv malakaviy ishimizning metodlari. Bitiruv malakaviy ishi ustida ishlash jarayonida muammoga oid adabiyotlarni o'rganish, o'quv-metodik ta'minotni tahlil qilish, ilg'or davlatlar tajribalarini o'rganib chiqish, mavjud onlayn buyurtma tizimlari tuzilishlarini aniqlash kabi **metodlardan** foydalanildi.

Bitiruv malakaviy ishimizning tarkibi va hajmi. Ushbu bitiruv malakaviy ishi _____ betdan iborat bo'lib, kirish, mundarija, 2 bob, 5 paragraf, xulosa, foydalanilgan adabiyotlar ro'yxati va ilovadan tashkil topgan.

I bobda zamonaviy web texnologiyalari, ma'lumotlar ombori haqida nazariy

bilimlar o'rganib chiqildi. HTMLdan boshlab JQuerygacha, ASP va PHP texnologiyalarining qo'llanilishi, ularning MySQL bilan bog'lanishi, SQL so'rovlarning turli xillari haqida nazariy bilimlar to'plandi. To'plangan bilimlar ketma-ketlik asosida tizimlashtirildi.

II bobda mavjud Onlayn buyurtma qabul qilish tizimlari o'rganildi. Ularning yutuq va kamchiliklari aniqlandi. Shu asosida Onlayn buyurtma qabul qilish tizimi uchun dasturiy spesifikasiya ishlab chiqildi. Tizimning server qismi, mijoz qismi, ma'lumotlar ombori shakllantirildi. Qulaylik uchun hisobotlar ishlab chiqildi.

I. BOB. ONLAYN TIZIMLAR BO'YICHA NAZARIY BILIMLARNI TO'PLASH VA O'RGANISH

1.1 PHP va MySQLda bog'lanishlar.

Internetning paydo bo'lishi o'zaro muloqotni amalga oshirish uchun yangi kommunikatsion texnologiyalar hamda ma'lumotlar bazasi bilan ishlovchi texnologiyalarni ishlab chiqishni taqozo etdi. Shu kungacha internetda sahifalar yaratuvchi va ular bilan ishlovchi oddiy texnologiyalar ishlab chiqilgan edi. PHP-MBBT (ma'lumotlar bazasini boshqarish tizimi) bilan amalda oson va tez bog'lana olish imkoniyatini beruvchi Web dasturlashning yangi texnologiyalaridan biri hisoblanadi.

PHP Webda ishlovchilar uchun html-sahifalarni tez va oson, dinamik holda ishlashi va o'zgartirish uchun imkoniyatlar yaratib beruvchi texnologiyadir. PHP texnologiyasining ajoyib xususiyatlaridan biri bu—ochiq dasturiy kodga egaligi va bepul tarqatilishidir. PHP platformaga bog'liq emasligi, deyarli barcha platformalarda ishlay olish imkoniyati bilan ajralib turadi. Shuning uchun dunyoda ko'plab Web sayt yaratuvchilar ushbu texnologiyani ma'qul ko'rishadi. Boisi PHP kuniga milliondan ortiq murojaatlarga xizmat ko'rsata oladi, unda skriptlarni yaratish va sozlash boshqa dasturlash tillariga nisbatan ancha osondir.

PHP tarixi va uning imkoniyatlari

PHP texnologiyasi Rasmus Lerdorf (Rasmus Lerdorf) tomonidan 1994-yilning kuz oylarida yaratilgan. PHP texnologiyasining birinchi versiyasi Perl'da yozilgan bo'lib, CGI obolochka sifatida bo'lgan. PHP texnologiyasi yaratilganiga uncha ko'p vaqt bo'lmasdan, 1995-yilning o'rtalariga kelib esa forma yordamida ma'lumotlar bilan ishlovchi Form Interpreter (FI) qo'shildi. Bu versiya ma'lumotlar bazasi bilan ishlay oldi, ya'ni MySQLni qo'llay oldi. 1998-yili shu texnologiyaning avvalgilariga qaraganda mukammalroq va qaytadan yozilgan PHP3 versiyasi yaratildi. Shundan so'ng PHP dunyo bo'yicha yanada ommalashib ketdi. Net Craftning hisob-kitoblariga ko'ra 1999-yilga kelib PHP dunyo bo'yicha 1 milliondan ortiq saytda ishlatilgan. 2001-yilning yanvarida esa bu son 5

millionga yetgan. Bugungi kunga kelib, dasturchilarning 100 mingdan ortig'i PHPdan foydalanishmoqda. PHP esa Apache serverida ishlovchi 14,5 mingta saytning taxminan 52 %ida ishlatilmoqda.

PHP texnologiyasining imkoniyatlariga batafsil to'xtalamiz.

Samaradorligi. PHP samaradorligi yuqori bo'lgan WEB sahifalar yaratishda yuqori natija beruvchi yagona texnologiyadir. U bir kunda milliondan ortiq murojaatlarga xizmat qila oladi. PHPda skriptlarni yaratish va sozlash boshqa dasturlash tillariga qaraganda osonroqdir.

Ma'lumotlar bazasi bilan integratsiyasi. PHP bir qancha ma'lumotlar bazasini boshqarish tizimlari bilan aloqa o'rnatadi. U MySQLga qo'shimcha ravishda PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, Interbase va Sybaselar bilan ham bog'lana oladi.

Biriktirilgan kutubxonalarining mavjudligi. Turli xil masalalarni yechishda ko'pgina kichkina CGI— ssenariylarini yozishga hojat yo'q, chunki bu sahifani yuklanishini sekinlashtiradi va sayt sahifalarini yaratishda xalaqit beradi. PHP texnologiyasi WEB bilan bog'lanish va bir qancha umumiy masalalarni hal qilish imkoniyatini yaratuvchi, qo'shimcha tashqi kutubxonalar yordamida kengaytirilgan turli foydali funksiyalar to'plamiga egadir.

Narxi. PHP paketi mutlaqo bepuldir. Uning oxirgi versiyalarini <http://www.php.net> saytidan bepul yozib olish mumkin.

PHP ni o'rganish. PHP sintaksisi boshqa dasturlash tillariga asoslangan, birinchi navbatda C va Perlga. Agar foydalanuvchi C yoki Perl bilan yoki C++ yoxud Java bilan tanish bo'lsa, PHPda dastur yozishda tezda yuqori samaradorlikka erishishi mumkin.

Ko'chuvchanligi. PHP paketini turli operatsion tizimlar boshqaruvida ishlatish mumkin. PHP kodini Linux va FreeBSD singari bepul, Solaris va IRIX singari kommersion UNIX operatsion tizimlar muhitida, Microsoft Windowsning turli

versiyalarida yaratish imkoniyati mavjud. Odatda, dasturlar turli xil PHP o'rnatilgan operatsion tizimlarda hech qanday o'zgarishlarsiz ishlaydi.

Hajmi. PHP texnologiyasida yaratilgan har qanday WEB sayt xotiradan uncha katta hajmni egallamaydi va u ochiq tizim sifatida ishlovchi ilg'ortexnologiya. PHP texnologiyasi Perl, Microsoft ning Active Server Pages(ASP), JavaServerPages(JSP) va Allaire Cold Fusion texnologiyalariga raqib bo'la oladi va MySQL MBBT bilan birgalikda funksional, dinamik WEBSaytlarni yarata oladi.

PHP deskriptorlarining ishlatilishi

PHP kodlari "<?" belgisi bilan boshlanib, "?>" belgisi bilan tugatiladi, xuddi HTML deskriptorlari "kichik" (<) belgisi bilan boshlanib va "katta" (>) belgisi bilan tugaganidek. Bu belgilar PHP deskriptorlari deb ataladi va ular WEB serverga PHP kodlari qayerda boshlanib, qayerda tugaganligi haqida xabar beradi. PHP kodlari bloki WEB serverda qayta ishlanib brauzerga uzatiladi. Bu **deskriptorlarning** turli xil ko'rinishlari mavjud bo'lib, ko'rsatib o'tilgani qisqartirilgan formasidir.

PHP deskriptorlarning to'rt xil turi mavjud:

Qisqartirilgan turi:

<? ?> Bu tur eng sodda tur bo'lib, SGML instruksiyasiga mos keladi.

XML turi:

<?php ?> Deskriptorning bu turidan XML hujjatlarda foydalanish mumkin.

SCRIPT turi:

<SCRIPT LANGUAGE='php'></SCRIPT>

Deskriptorning bu turi JavaScript va VBScript dasturlari bilan ishlaganlar uchun juda yaxshi tanish.

PHP operatorlari

PHP interpretatori bajarishi kerak bo'lgan biror amal PHP operatori yordamida ko'rsatiladi. Quyidagi misolda PHP ning ma'lumotlarni chiqazish operatorlarini ko'rib chiqamiz. PHP da ma'lumotlarni ekranga chiqazish operatori echo va print operatorlaridir.

echo konstruksiyasi judayam oddiy ishlaydi. U brauzer oynasiga qo'shtirnoq ichidagi yozuvni chiqazadi. Amal bajarilgandan so'ng albatta nuqta-vergul qo'yilishi kerak, agar nuqta-vergul unutilib qoldirilsa, u holda dasturda xatolik yuzaga keladi. Masalan, _____

```
<? echo "WEB sayt yaratamiz!"; ?> _____
```

Yoki _____

```
<htmlXhead><title>Misol</title></head>
```

```
<body><? echo "WEB sayt yaratamiz!"; ?></body>
```

```
</html>
```

O'zgaruvchi tiplari

O'zgaruvchilar haqida siz maktab kurslaridan tushunchaga egasiz. O'zgaruvchi bu o'zida ma'lumotni, ya'ni biror-bir so'zni, biror-bir sonni saqlovchi belgi. O'zgaruvchilar dastur bajarilayotgan vaqtda, ustida amallar bajarish mumkin bo'lgan ma'lumotlarni o'zida jamlagan xotira maydoni nomini ifodalaydi. PHP da o'zgaruvchilar \$(dollar) belgisi va o'zgaruvchi nomini yozish orqali yaratiladi. O'zgaruvchi nomi biror-bir harf yoki so'z bilan ifodalanadi, lekin son bilan ifodalanishi mumkin emas. Masalan: \$name, \$n, \$a2 va h. O'zgaruvchiga biror qiymat berish uchun uni o'sha qiymatga tenglashtirish lozim. Tenglashtirish "=" amali yordamida amalga oshiriladi. Bir narsani yoddan chiqarmang, PHP da "=" amali aynan teng ma'nosini emas, balki tenglashtirish ma'nosini bildiradi. Agar bu ikki tushuncha aralashtirilib yuborilsa, dastur kodida xatoliklar yuzaga keladi.

Endi o'zgaruvchini biror-bir qiymatga tenglashtiramiz. Masalan, belgili qiymatga, u holda belgili qiymat "" ichida yozilishi kerak. Va har bir satr " ; " belgisi bilan yakunlanishi kerak.

`<? $name="Lola"; ?>` _____

Agar o'zgaruvchiga son qiymat beradigan bo'lsak, u holda "" belgisi qo'yilmasa ham bo'ladi. _____

`<? ~`

`$name="Lola";`

`$age=22;`

`?>` _____

PHP quyidagi ma'lumot tiplarini qo'llaydi:

Skalyar tiplar:

- boolean (mantiqiy)
- integer, int (butun)
- float, real, double (haqiqiy)
- string (satri)

Aralash tiplar:

- array(massiv)
- object (obyekt)

Maxsus tiplar:

- resource(resurs)

Skalyar tipdagi o'zgaruvchilar

Skalyar tipdagi o'zgaruvchilarni ma'lumotlarni saqlovchi bir quti desak ham bo'ladi. U o'ziga sonli, belgili va matnli qiymatlarni qabul qiladi. Skalyar tipdagi o'zgaruvchilar haqida yanada aniqroq tasavvur hosil qilish uchun quyidagi sxemani ko'rib chiqamiz.

\$a =

\$b =

Boolean (bulev yoki mantiqiy) tipi Bu oddiy tip qiymatni rostligini aniqlaydi, bu tipdagi o'zgaruvchi faqatgina ikkita qiymat qabul qiladi - rost (TRUE) yoki yolg'on (FALSE).

Mantiqiy tipni aniqlash uchun, ikkita kalit so'z TRUE yoki FALSE ishlatiladi.

2.3-misol. Mantiqiy tip

<? \$test = True; ?>_____

integer yoki int (butun) tipi

Integer (butun) tipi butun sonlar to'plamida ($Z = \{ \dots, -2, -1, 0, 1, 2, \dots \}$) gi qiymatlar uchun ishlatiladi. Butun tiplar o'nlik, o'n oltilik va sakkizlik sanoq sistemalaridagi qiymatlarni yoki "+" belgilarini ishlatgan holda ham qabul qiladi. Agar sakkizlik sanoq sistemasi ishlatilgan bo'lsa, raqamdan avval 0(nol) sonini, o'n oltilik sanoq sistemasida esa raqamdan avval Ox qo'shib yozish kerak.

<?php

\$a = 1234; // o'nli sanoq sistemasidagi son

\$a = -123; // manfiy son

\$a = 0123; # sakkizlik sanoq sistemasidagi son

\$a — Ox1A; # o'n oltilik sanoq sistemasidagi son

?>_____

Butun tipning eng katta qiymati 2 milliardga yaqin (bu 32 bit belgi). Belgisiz butun qiymatlarni PHP qabul qilmaydi. Float, real, double (nuqta bilan ajratilgan son qiymati) tipi Nuqta bilan ajratilgan son qiymati (haqiqiy son qiymati) tipi quyidagicha sintaksislarga ega._____

```
<?php
```

```
$a = 1.234;
```

```
$b = 1.2e3;
```

```
$c = 7E-10;
```

```
?> _____
```

Double(ikkilik qiymat) float, real tiplari bir-biriga o‘xshash bo‘lib, haqiqiy son qiymatlari uchun ishlatiladi.

String (satrlı) tipi. Satr — bu belgilar to‘plami. PHP da boshqa dasturlardagi kabi bayt bu turli 256 ta belgini qabul qiladi. String tipi satrlı qiymatlar uchun ishlatiladi.

PHP da satr turli xil usullar bilan beriladi;

— bittalik qo‘shirnoq yordamida

— ikkitalik qo‘shirnoq yordamida

— heredoc — sintaksisi yordamida

MySQL:tarix va imkoniyatlar

SQL(Structured Query Language—Strukturalashgan so‘rovlar tili)—relyatsion MB bilan ishlashda qo‘llaniladigan so‘rovlar tili.

MySQL (MyStructured Query Language)—relyatsion ma‘lumotlar bazasini boshqarish tizimidir. MySQLning tarixiga nazar solsak, bu tizim Mixael Videnius

tomonidan yaratilgan va dasturiy ta'minot bozoriga 1996-yili chiqarilgan bo'lsa-da, aslida bu tizimdan 1979-yildan buyon keng foydalanilmoqda.

MySQLning imkoniyatlari bilan tanishib chiqamiz.

Samaradorligi. MySQL—foydalanish uchun juda tezkor, ishonchli, oson va oddiy tizimdir. MySQL client/server texnologiyasi asosida ishlaydi. O'zining yengil qo'llanilishi bilan MySQL tizimi Internet tarmog'ida juda keng tarqalgan.

Dastur kodining ochiqligi. MySQL paketi ochiq kodli dasturiy ta'minot hisoblanadi, uni bunday nomlanishining sababi bu—har bir foydalanuvchi dasturni ishlata olishi va kodini o'qib chiqqan holda ichki kodini o'rganishi va o'ziga moslashtirib o'zgartirishi mumkin. Xuddi PHP kabi MySQLning dasturiy kodini ochib ko'rish va o'zgartirish mumkin.

Sozlash va o'rganishda soddaligi. Ko'pgina zamonaviy MBBTlarida SQL qo'llanilgan. Agar foydalanuvchi birorta MBBTlari bilan tanish bo'lsa, MySQLga o'tishda u hech qanday qiyinchilikka duch kelmaydi. Xuddi ko'pgina dasturlar singari MySQLni sozlash va o'rnatish ham sodda.

Xavfsizligi. Ma'lumotlar xavfsizligini yuqori darajada ta'minlaydi. MySQL tizimi ma'lumotlarga ruxsat berishni boshqaradi, ma'lumotlar bilan ishlash uchun bir vaqtning o'zida bir nechta foydalanuvchiga tezkor ruxsatni ta'minlaydi va foydalanuvchilarga faqatgina ruxsat berilgan ma'lumotlar bilan ishlashiga kafolat beradi. Parollar va huquqlarning mukammal tizimi. Tizim parol va kafolatlariga (priviligiyalarga) asoslangan yuqori darajali, mukammal va xavfsiz bo'lib, xost vositalari orqali tekshiruvni tashkil qilishga imkon yaratadi. Parollar himoyalangan, bog'lanish vaqtida barcha trafik parollari shifrlanadi. Parollar jadvalida 16 tagacha kalit mavjud bo'lib, har bir kalit 15 tagacha maydondan foydalana olishi mumkin.

Parollar va huquqlarning mukammal tizimi. Tizim parol va kafolatlariga (priviligiyalarga) asoslangan yuqori darajali, mukammal va xavfsiz bo'lib, xost

vositalari orqali tekshiruvni tashkil qilishga imkon yaratadi. Parollar himoyalangan, bog‘lanish vaqtida barcha trafik parollari shifrlanadi. Parollar jadvalida 16 tagacha kalit mavjud bo‘lib, har bir kalit 15 tagacha maydondan foydalana olishi mumkin.

Turli operatsion tizimlarga mosligi. Bir qancha turli operatsion tizimlarda va platformalarda ishlaydi. MySQL dan UNIX, Microsoft Windows va boshqa turli muhitlarda ham foydalanish mumkin. MySQL ning ayni vaqtda juda ko‘p muxoliflari mavjud bo‘lib, ular qatoriga mSQL, dbm, Hyperware, Informix, Interbase, Sybase, PostgreSQL, Microsoft SQL Server va Oracle lar kiradi. 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 guruhlargaaajratilgan 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.

Har bir foydalanuvchi SQL ma‘lumotlar bazasida nima qilish mumkinligini ko‘rsatuvchi imtiyozlarga egadir. Bu imtiyozlar vaqt o‘tishi bilan o‘zgarishi, ya‘ni eskilari o‘chirilib yangilari qo‘shilishi mumkin. SQL imtiyozlari bu ob‘ekt imtiyozlaridir. Bu shuni bildiradiki, foydalanuvchi berilgan komandani ma‘lumotlar bazasining biror ob‘ekti ustida bajarishi mumkin. Ob‘ekt imtiyozlari bir vaqtning o‘zida foydalanuvchilar va jadvallar bilan bog‘liq. Ya‘ni imtiyoz ma‘lum foydalanuvchiga ko‘rsatilgan jadvalda, asos jadvalda yoki tasavvurda beriladi. Ixtiyoriy turdagi jadvalni yaratgan foydalanuvchi shu jadval egasidir. Bu shuni bildiradiki foydalanuvchi bu jadvalda hamma imtiyozlarga ega va imtiyozlarini shu jadvalning boshqa foydalanuvchilariga uzatishi mumkin. Foydalanuvchiga tayinlash mumkin bo‘lgan imtiyozlar:

SELECT - bu imtiyozga ega foydalanuvchi jadvallarda so‘rovlar bajarishi mumkin.

INSERT -bu imtiyozga ega foydalanuvchi jadvalda INSERT komandasini bajarishi mumkin.

UPDATE - bu imtiyozga ega foydalanuvchi jadvalda UPDATE komandasini bajarishi mumkin. Bu imtiyozni jadvalning ayrim ustunlari uchun cheklab qo'yish mumkin.

DELETE -bu imtiyozga ega foydalanuvchi jadvalda DELETE komandasini bajarishi mumkin.

REFERENCES -bu imtiyozga ega foydalanuvchi jadvalning ustunidan (yoki ustunlaridan) ajdod kalit sifatida foydalanuvchi tashqi kalit aniqlashi mumkin.

SQL tilida quyidagi asosiy ma'lumotlar turlari ishlatilib, ularning formatlari har xil MBBT lar uchun farq qilishi mumkin:

INTEGER - butun son (odatda 10 tagacha 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 yuborilishi mumkin).

FLOAT - haqiqiy son 15 ta qiymatli raqam va butun darajadan iborat. Daraja 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'zgaruvchi, n simvoldan oshmagan simvolli qator ($n > 0$ va har xil MBBT larda har xil lekin 4096 dan kam emas).

DATE - maxsus komanda orqali aniqlanuvchi formatdagi sana, 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.

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.

SQL Standarti ANSI tomonidan aniqlangan va hozirda ISO tomonidan qabul qilingan. Lekin kommertsial ma'lumotlar bazalari dasturlari ANSIning o'zgartirilmaganidan SQLni kengaytiradilar, ya'ni foydali hisoblagan har xil xossalarni qo'shadilar.

SQL tili standartlari SQL tilining rasmiy standarti Amerika milliy standartlar instituti (American National Standards Institute —ANSI) va standartlar bo'yicha xalqaro tashkilot (International Standards Organization — ISO) tomonidan 1986 yilda e'lon qilindi va 1992 yilda sezilarli darajada kengaytirildi. Bundan tashqari SQL AQSH ning axborotlarni qayta ishlash bo'yicha federal standarti (FIPS — Federal Information Processing Standard) bo'lib xisoblanadi va shu asosida xisoblash texnikasiga bog'liq katta davlat shartnomalarida asosiy talabni tashkil etadi. Evropadagi UNIX operatsion sistemasi asosida dasturlash muxitining standarti bo'lgan XOPEN o'ziga SQLni berilganlar bazasiga murojaat qilish standarti sifatida biriktirgan.

SQL tili, SQL server berilganlar bazasini boshqarish tizimlari

Ma'lumotlar bazasi dunyosi tobora yagona bolib bormoqda. Bu jarayon har xil kompyuter muhitlarida faoliyat ko'rsatuvchi axborot tizimlarini hosil qilishda qo'llanuvchi yagona standart til yaratishni talab qildi.

Buyruqlar to'plamini bilgan foydalanuvchilarga standart til, ular shaxsiy kompyuter tarmoq ishchi stansiyasida yoki katta EHM da ishlashidan qat'iy nazar, ma'lumot yaratish, izlash va uzatishga imkon beradi.

SQL (Structured Query Language, odatda “sikvel” deyiladi), ma’nosi — Tartiblangan so'rovlar tili. Bu relyatsion ma'lumotlar bazalarida ishlashga imkon beradigan tildir. Bu til ifodalarining xususiyati shundan iboratki, ular ma'lumotlarni qayta ishlash protseduralariga emas, natijalariga yo'naltirilgandir.

SQL o'zi ma'lumotlar qayerda joylashgani, indekslar qandayligini va, hatto, amallarni qanday izchillikda qo'llasa bo'lishini aniqlaydi, bu tafsilotlarni ma'lumotlar bazasiga so'rovlarda ko'rsatish kerak emas. SQL tili IBM kompaniyasida MBBT DB2 yaratish jarayonida ishlab chiqilgan va keng ko'lamda RISC protsessorli mashinalarda UNIX tizimlar asosida hamda meynfreymlarda, super kompyuterlar asosida qurilgan katta hisoblash tizimlarida qo'llanilgan. Shu bilan birga, u mustaqil bo'lmasdan, PL/SQL va Transact-SQL kabi ichki dasturlash tillariga inkapsulyatsiya qilinadi. 1986-yilda ANSI (American National Standart Institute) SQL tilining rasmiy standartini ishlab chiqdi, 1992- yilda bu standart kengaytirildi. Butun til 30 ga yaqin operatorlarga ega bo'lib, ba'zi versiyalarida sal ko'proq, ba'zilarida sal kamroqdir. Har qanday MB har xil ob'ektlarga, ya'ni jadvallar, protseduralar, funksiyalar, tasavvurlar, ketma-ketliklar va hokazolarga ega.

“Kliyent-Server” texnologiyasiga ko'ra, foydalanuvchi EHM (Kliyent) lar so'rovlari maxsus ma'lumotlar serverlarida qayta ishlanadi, foydalanuvchi EHM larga faqat so'rovni qayta ishlash natijalari qaytariladi. Tabiiyki, Server bilan muloqot qilish uchun yagona til kerak va bunday til sifatida SQL tanlangan.

Shuning uchun hamma zamonaviy relyatsion MBBT versiyalarida (DB2, Oracle, Ingres, Informix, Sybase, Progress, Rdb) va, hattoki, norelyatsion MBBT versiyalarida (masalan, Adabas) “Klient-Server” texnologiyasi va SQL tilidan foydalaniladi. SQL tilida ma'lumotlarni jadval ko'rinishda tasvirlashga yo'naltirilgan amallar konsepsiyasi ko'p bo'lmagan (30 dan kam) ifodalardan iborat kompakt til yaratishga imkon berdi. Ikki xil: interaktiv va joylashtirilgan SQL mavjud. Ko'p hollarda ikkala forma bir xil ishlaydi, lekin ikki xil foydalaniladi.

Interaktiv SQL ma'lumotlar bazasining o'zida faoliyat ko'rsatadi va buyurtmachi foydalanishi uchun chiqish hosil qilishda ishlatiladi. SQLning bu formasida buyruq kiritilsa, u darrov bajariladi va darhol natijani (agar u mavjud bo'lsa) ko'rish mumkin. Joylashtirilgan SQL boshqa tilda yaratilgan dasturga joylashtirilgan SQL buyruqlardan iborat.

SQLning interaktiv va joylashtirilgan formalarida ko'p sonli guruhlar yoki subbolimlar mavjud. Ular ANSI tomonidan e'tiborga olingan va konseptual darajada foydali, lekin ko'pchilik SQL dasturlar ularni alohida qayta ishlamaydi, shuning uchun ular aslida SQL buyruqlarining funksional kategoriyalaridir.

- DDL (Ma'lumotlarni Ta'riflash Tili) — ANSI da sxemani ta'riflash tili, obyektlarni (jadvallar, indekslar, tasavvurlar va hokazo) yaratuvchi buyruqlardan iborat.

- DSL (Ma'lumotlarni O'zgartirish Tili) — bu jadvallarda qanday qiymatlar saqlanishini istalgan daqiqada aniqlovchi buyruqlar majmuasi.

- DCL (Ma'lumotlarni Boshqarish Tili) - foydalanuvchiga ma'lum obyektlar ustida malum ta'sir o'tkazishga ruxsat berish yoki bermaslikni aniqlovchi vositalardan iborat. SQL Standarti ANSI tomonidan aniqlangan va hozir ISO (Standartlashtirish bo'yicha xalqaro tashkilot) tomonidan qabul qilingan. Lekin kommersial ma'lumotlar bazalari dasturlari ANSI ni ogohlantirmasdan SQL ni kengaytiradilar, ya'ni o'zlari foydali deb hisoblagan har xil xossalarni qo'shadilar.

SQL so'rov tili — ma'lum talablar asosida berilganlar bazasiga murojaat qilib, undan so'rov talablariga javob beradigan natijalarni olish tili. Hozirgi paytda turli berilganlar bazasi va ularning sistemalari yaratilgan. Ammo relyatsion tipdagi barcha berilganlar bazasiga murojaat qilib, ulardan tegishli ma'lumotlarni olish SQL tilida ishlab chiqilgan. Bu tilda so'rovlar qanday vositalar yordamida tashkil qilinishiga to'xtalib o'tamiz. Shuni aytish joizki, ba'zi DBMS larda (masalan, Accessda) uni ishlatmasdan tanlashga so'rov yoki maxsus namuna bo'yicha so'rov banki sifatida murojaat qilib tegishli ma'lumotlar olinadi, bunday so'rovlarni

tashkil qilish uchun esa Базаданных — Создать (Berilganlar bazasi yaratish) buyrug'idan foydalaniladi. So'rov natijasini javob sifatida jadval ko'rinishida olish mumkin.

SQL — interaktiv so'rovlar tili. Foydalanuvchilar SQL komandalari asosida ishlovchi ma'lumotlarni o'qish va ularni ekranga chiqarishga mo'ljallangan interaktiv dastur orqali berilganlar bazasi ma'lumotlaridan foydalanadilar. Bu esa maxsus so'rovlarni hosil qilishning qulay usulidir.

SQL — berilganlar bazasini dasturlash tili. Berilganlar bazasi bilan ishlashda dasturchilar o'zlarining dasturlarida SQL komandalaridan foydalanadilar. Bu uslub foydalanuvchi tomonidan yozilgan dasturda qo'llanilganidek berilganlar bazasining xizmatchi dasturlarida ham ishlatiladi.

SQL — berilganlar bazasida administratorlik qilish tilidir. Berilganlar bazasining administratori SQL tilidan berilganlar bazasi strukturasi aniqlashda va ulardan foydalanish huquqini boshqarishda foydalanadi.

SQL — klient — server texnologiyasi asosida dasturlarni tuzish tilidir. Shaxsiy kompyuterlar uchun tuzilgan dasturlarda SQL tili lokal tur orqali berilganlar bazasi serveri bilan aloqani tashkil etish uchun qo'llaniladi. Ko'pgina yangi dasturlarda klient — server arxitekturasi tarmoq trafigidan minimum darajada foydalanish hamda shaxsiy kompyuterlar singari berilganlar bazasi serverlari tezligini oshirish uchun ishlatiladi.

SQL — taqsimlangan berilganlar bazasi tili. Taqsimlangan berilganlar bazasi sistemalarini boshqarishda SQL bir necha o'zaro aloqa qiluvchi hisoblash sistemalari o'rtasida ma'lumotlarni taqsimlashga yordam beradi. Har bir sistemaning dasturiy ta'minoti SQL dan foydalanib boshqa sistemalar bilan aloqa qiladi, ularga so'rovlar uzatadi.

SQL — berilganlar bazasida shlyuzlar o'rnatish imkonini beruvchi tildir. Hisoblash tarmoqlarida SQL dan bir BBBTidan boshqa turdagi BBBT bilan bog'lash uchun foydalanadilar.

SQL tili birinchi navbatda so'rovlarni bajarishga mo'ljallangan. SQL so'rovlarini ko'rish uchun qo'llaniladigan SELECT operatori boshqa SQL operatorlariga nisbatan ancha keng imkoniyatli hisoblanadi. SELECT operatori berilganlar bazasidan ma'lumotlarni o'qiydi va so'rov natijalarini jadval ko'rinishida qaytaradi. SELECT operatori MB jadvallaridan 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. SELECT operatori oddiy so'rovlar uchun ikkita zaruriy predlojeniyadan iborat bo'ladi. SELECT predlojeniyasida talab qilingan ustun nomlari keltiriladi. FROM predlojeniyasida esa bu ustunlarni saqlovchi jadvallar ko'rsatiladi. So'rov mantiqiy darajada FROM predlojeniyasida ko'rsatilgan jadvallarni satrma - satr ko'rib chiqish orqali bajariladi. Jadvalning har bir satridan natija sifatida chiqariladigan ustunlarning qiymatlari olinadi va so'rov natijasining bitta satri tuziladi. Shunday qilib oddiy so'rov natijalari jadvali berilganlar bazasi jadvalining har bir yozuvi uchun bitta berilganlar satri hosil qiladi. 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 guruhlarga ajratilgan natijaviy so'rov yaratishga imkon beradi. HAVING ifodasidan guruhlarni qaytarish sharti yoziladi va GROUP BY bilan birga ishlatiladi. ORDER BY ifodasi ma'lumotlar natijaviy to'plamini tartiblash yo'nalishini aniqlaydi. OFFICES jadvalidagi hamma yozuvlarni qaytaruvchi sodda so'rov ko'ramiz:

I.2. JQuery va Ajax texnologiyalari

JQuery nima, asosiy tushuncha va imkoniyatlari

Siz o'z web sahifangizga slayd ko'rinishidagi rasmlarni o'rnatmoqchidsiz, ehtimol siz saytingizga tashrif buyuruvchilarga ma'lumotlarni yuklashlari uchun qulayliklar yaratmoqchidsiz, unda sizga jQuery yordam beradi. Shuningdek jQuery orqali interaktiv effektlarni web sahifaga tezlik va osonlik bilan joylashtirishingiz mumkin. Bu mavzuda jQueryni umumiy qanday ishlatilishi, web sahifada qay usulda qo'llanilishi va jQuery effektlarini qay usulda qo'llash haqida ma'lumotga ega bo'lishingiz mumkin.

JQueryni tushinish JavaScript haqida qisqacha ma'lumotga ega bo'lishga va web brauzeringizdagi dastur tilini tushinishga yordam beradi. JavaScript kodi web sahifangizdagi rasmlar va matnlar bilan o'zaro bog'liq. Bunga misol qilib rasmni yashirish, tekstni harakatga keltirish, ma'lum bir vaqtdan keyin web sahifadagi ma'lumotlarni o'zgartirish, foydalanuvchilar sahifada bo'lganda qandaydir ish bajaradi. (M: sichqoncha tugmasini yo'nalish ustiga yetkazadi, ma'lum bir animatsiya ishlarini bajaradi, xususan ma'lum bir tugmani bosish natijasida JavaScript kodi rasmni paydo qiladi va 30sekunt davomida rasmni ushlab turadi). JavaScript mukammal dastur bo'lib va odatda web sahifaga interaktiv va dinamik effektlarni qo'shish uchun foydalaniladi. Mukammal va samarali ishlash uchun butilni o'rganish juda katta ahamiyatga ega.

JQuery biz uchun nima qila oladi?

JQuery web sahifaga boshlang'ich ishlovchilar va dizaynerlarga ajoyib effektlarni deyarli hech qanday kodlarsiz o'rnatishga yordam beradi. Chunki jQueryning o'zi alohida kutubxona va uning mashhurligi tobora ortib bormoqda. Yuqorida ta'kidlanganidek JQueryni siz web sahifalarda uchratishingiz mumkin.

Ko'pgina jQuery effektlaridan foydalanuvchi saytlar Facebook, Twitterdir. Siz Twitterga yangi ma'lumot qo'shganingizda sahifa yangilanmaydi. Uning o'rniga web sahifa ortidagi kod, jQueryga sahifani yangilamasdan yangi ma'lumot bir qator pastga tushishga buyruq beradi. Yoinki Twitterda JQueryni boshqa bir elementlardan biri yangi Twit (xabar) kelganida sahifa yangilanmasdan xabar kelganligi haqida ma'lumot beradi.

Agar siz ko'zda tutilayotga xabarga bossangiz, yangi xabar sizning sahifangizda paydo bo'ladi va twitlar ro'yxatiga qo'shiladi. Web sahifa hech qachon yangilanmaydi. Uning o'rniga sahifadagi ma'lumotlar dinamik tarzda o'zgaradi. jQuery bu JavaScript uchun add-on kutubxonasidir. JQueryni go'yo JavaScript kodlari o'rnida ko'rishingiz kerak, qisqa qilib aytganda sahifadagi bir necha qator kod JQueryga buyruq beradi.

JQueryning vazifalari.

JQuery osonlik bilan sizning web sahifadagi ma'lumotlarning ko'rinishini, joylashuv tartibini yoki vazifasini o'zgartirishga ega. JQuery kodi web sahifadagi rasmlar va matnlar ustida turli xil effektlar bajarilishini ta'minlaydi:

hide: web sahifadagi elementlarni yashiradi

show: web sahifadagi yashirin elementni namoyish qiladi

slideDown: web sahifadagi elementni slayd ko'rinishida pastga tushiradi

slideUp: web sahifadagi elementni slayd ko'rinishida tepaga ko'taradi

fadeIn: web sahifadagi yashirin elementga sichqoncha belgisi

eltganda, element paydo bo'ladi

fadeOut: web sahifadagi elementdan sichqoncha belgisi olinganda, yashirin holatga o'tishi

animate: web sahifadagi elementni ma'lum bir yo'nalishga yo'naltiradi.

Siz yuqoridagi jQuery kodlaridan foydalangan holda sahifangizdagi elementlar ustida yashirish, slaydlar hosil qilish va animatsiyalar o'rnatish kabi bir qator amallarni bajarishingiz mumkin. Quyidagi harakatlar bajarilgandan so'ngamalga oshuvchi funksiyalarni ko'rib chiqamiz:

load: web brauzer yoki element yuklangandan so'ng sodir bo'ladigan effekt

mouseover: web sahifaning ma'lum bir qismiga sichqoncha belgisi eltganda sodir bo'ladigan effekt

mouseout: web sahifaning ma'lum bir qismidan sichqoncha belgisi ketgandasodir bo'ladigan effekt

change: web sahifadagi biror element o'zgarganda sodir bo'ladigan effekt.

Plaginlar(moslamalar)dan foydalanish:Haqiqiy JQueryning kuchi yuzdan ortiq plaginlarga bog'liq bo'lib, bu ajoyib effektlar yaratishga yordam beradi. Plaginlar-JavaScript dasturlar bo'lib, bu JQueryni kengaytirish va kuchaytirish vazifasini o'taydi. Mashhur va ajoyibplaginlardan birini misol taraqasida ko'rishingiz mumkin:

<http://leandrovieira.com/projects/jquery/lightbox>.

Bu plagin foydalanuvchi tomonidan rasm ustiga bosilganda obyekt hajmikattalashgan holda paydo bo'ladi.

JQueryni ko'chirish. JQuerydan foydalanishdan oldin nusxasini ko'chirib olishingiz va uni nima maqsadda foydalanishingiz haqida ma'lumotga ega bo'lishingiz kerak. Bu mavzuda JQueryni ko'chirish va shaxsiy kompyuterga yoki boshqa web serverga o'rnatish haqida ma'lumotlar beriladi.

JQueryni ko'chirish uchun quyidagi ketma ketliklar bajariladi:

Brouzerda jQuery.com saytiga kiriladi va Download tugmasi bosiladi:

Buni bosish orqali siz <http://code.google.com> saytida jQueryni oxirgi versiyasiga eltadi. Odatda fayl nomi jquery-1.4.min.js.(nomi versiya ko'chirib olingan jQuery bilan bog'liq) bo'ladi. Ehtimol sizning versiyangiz yangiroq yoki butunlay boshqa bo'lishi ham mumkin. Ammo bu, kitobdagi kodlar va misollarda hech qanday farq bo'lmaydi.

JQuery.com saytidan JQueryning oxirgi versiyasini ko'chirish.

min so'zi faylning kichikligini bildiradi. Bu - elementlar va yozuvlar, umuman olganda hech narsa o'chib ketmaganligini, shunchaki faylning umumiy hajmi kichikligini anglatadi. Shuni aytib o'tish mumkinki, JQueryning kichraytirilgan versiyasini ko'chirib olgan ma'qulroq.

jQuery yo'nalishi fayllariga tugmani bosish.Buni bosganda siz uni ochishingiz yoki saqlashingiz haqida so'rovchiqadi.Siz qanday brauzer ishlatayotganingizga qarab jQueryni.jsfaylida saqlashingizyoki shunchaki saqlash tugmasini bosishingiz mumkin. Uning o'rniga siz to'liq Java Script kodini ko'rasiz.Bunda siz faylni tanlaysiz va xotirada saqlaysiz. Kompyuterning esda qolarli biror joyda xotirada saqlang

O'rnatish va tekshirish. Endi sizda JQueryning nusxasi mavjud va kompyuterning ma'lum bir qismida JQuerydan foydalanish uchun joy tanlaymiz. Buni HTMLda quyidagichabajaramiz:Kompyuterda juda ham oddiy, esda qolarli papka yarating. Masalan

webtest. C:/webtest;

web-test papkada images va js papka yaratilgan ko'rinishi.

Webtest papka ichida yana 2ta papka yarating va ularning nomini jsvaimages deb nomlang. Rasm 2.1.4 da webtest papkasini ko'rishingiz mumkin:

jQuery faylidan nusxa olib js papka ichiga tashlang.

Yuqoridagi amallar bajarilgandan so'ng o'z web sahifangizni yaratishingiz mumkin. Kitob davomida sizga HTML (editor) tahrirlovchi kerak bo'ladi. Hech bo'lmaganda Blaknot(texteditor)dan ham foydalanishingiz mumkin. Hozir ko'rib o'tiladigan usulda HTML sahifasiga kodni qo'shish orqali JQuery kutubxonasiga bog'lanishni ko'rib o'tamiz. Bu sahifa hech qanday ajoyib natija bermasada, JQuery kodini web sahifaga o'rnatishni oddiy usul orqali tushinishga yordam beradi.

JQuery kutubxonasiga ulanish: Aslida ko'pchilik harakatlar, HTML-hujjat elementlari orqali bajariladi. Shuning uchun hujjat to'liqligicha yuklangandan keyin skript bajariladi. Shumaqsadda JQuery kutubxonasi onload qurilmawindowhodisasidan foydalaniladi:

```
window.onload = function() {  
alert("Hujjat to'liq yuklangan");}
```

Lekin bu amal faqatgina HTML- hujjat to'liq yuklab bo'lingandan keyin amalga oshadi, shuningdek boshqa barcha elementlar ham. Misol uchun biror bir ko'rinish. Biror ko'rinishni yuklashda ancha vaqtni olishi va buni bajarish qiyinlashishi mumkin. JQuery kutubxonasi bizni bunday vaqt yo'qotishlardan xolos qilib, boshqa elementlarning yuklanishini kutmasdan hujjatning tarkibi tezda tuzilganda bu amalni bajarishga imkon beradi. Bu hodisareadyusuli orqali amalga oshiriladi:

```
JQuery(document).ready(function() {  
alert("Hujjat bilan ishlashga ruxsat berilgan, script  
bajarilishiga");  
});
```

JQuery funksiyasi o'rniga \$ ni qo'llash mumkin. Bu quydagicha amalga oshiriladi:

```
$(document).ready(function() {
```

```
alert("Hujjat bilan ishlashga ruxsat berilgan, script  
bajarilishiga "); });
```

Agar parametr sifatida jQuery funksiyasi vazifasi ko'rsatilsa, hujjat tuzilmasi darhol bajariladi:

```
jQuery(function() {  
alert("Hujjat bilan ishlashga ruxsat berilgan, script  
bajarilishiga ");  
});
```

Ushbu kodni yanada qisqartirish mumkin:

```
$(function() {  
alert("Hujjat bilan ishlashga ruxsat berilgan, script  
bajarilishiga ");  
});
```

Shuni esda tutingki `dasturreadmetodiga` bir nechta buyruq berishi mumkin.

Bu holatda metod beriladigan buyruq asosida ketma-ketlik bilan bajariladi.

Quyidagi holatdagi ketma-ketlikga e'tiborizni qarating. HTML dastur yaratamiz va bir nechta **ready()** metodi, shuningdek **onload** jarayonini ko'rib chiqamiz.

Listing kod 2.1. Jarayonlar ketma-ketligi:

```
<html>  
  
<head>  
  
<title>Jarayonlar ketma-ketligi </title>
```

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

<script src="jquery.js" type="text/javascript"></script>

<script type="text/javascript">

window.onload = function() {

alert("Onload hodisasi");

}

$(document).ready(function() {

alert("ready()metodi. Chaqiruv 1");

});

$(document).ready(function() {

alert("ready()metodi. Chaqiruv 2");

});

$(function() {

alert("$.funksiyasi chaqiruvi");

});

</script>

</head>

<body>

</body>

</html>
```

Ushbu amal quyidagi ketma-ketlikda bajarilishiga e'tiboringizni qarating:

1. ready()metodi. Chaqiruv 1,
2. ready()metodi. Chaqiruv 2,
3. \$()funksiyasi chaqiruvi,
4. Onload hodisasi.

Ko'rganingizdek **onload** hodisasi eng oxirida sodir bo'ladi.

Ready()va\$()metodlari dasturichida ketma-ketlik orqali sodir bo'ladi.

JQueryda selektorlar va elementlarning takrorlanishi: Sizga ma'lumki funksiya parametrlari \$() belgisi orqali selektorlar ko'rsatilishi mumkin. Chunki bu CSS standartining bir bo'lagi hisoblanadi. Selektorning har xil holatlarini ko'rib chiqamiz.

Ko'p qo'llaniladigan selektorlar:

Ko'p qo'llanadigan selektorlarni ko'rib chiqamiz:

*- teglarning barcha yig'indisi;

```
var n = $("*").size () ;// teglar soni
```

Teg – nomi ko'rsatilgan teglar, barcha teglar yig'indisi;

```
$("#I").css("backgroundColor", "red");// barcha satr boshlarga qizil fon beramiz.
```

#Identifikator- elementga identifikatorning ko'rsatilishi;

```
$("#id1") .cssp("backgroundColor", "red") ;// id="id" elementga qizil
```

fon beramiz.(Agar elementlarning ID raqamlari bir xil bo'lsa, yo'nalish faqat 1-elementgaqaytadi.)

Teg#Identifikator— maxsus tegni ichidagi #id tegi aniqlangan idintifikatorga qaytadi.

```
$("#p#id1") .css("backgroundColor", "red") ;// satr boshi orqasiga qizil
```

rang beramiz.(Agar faqatgina satr boshida bir xil ID raqamlari bo'lsa, birinchi ko'rsatilgan satrboshiga qaytasiz, identifikator boshqa bir teg ichida bo'lsa uni tanimaydi.)

Sinf — sinfni toifalaydigan elementlar to'plami.

```
$(".cls2").css("backgroundColor", "red");// class="cls2" elementi bilan boshlanuvchi, elementlarini orqa fonini qizil qilish;
```

Teg. Sinf — maxsus bir sinfni ta'riflaydigan elementlar to'plami

```
$("#p.cls2").css("backgroundColor", "red") ;// class="cls2" elementi bilan boshlanuvchi, elementlarini orqa fonini qizil qilish;
```

Eslatma:Tanlovni tezlashtirish uchun teg nomini ID bilan belgilamang.

Masalan, "p#id1" o'rniga "#id1"ni yozing. Qidirish dizaynerlashtirilgan sinf ko'rinishda bo'lishi kerak, boshqacha qilib aytganda teg nomini o'z ichiga olishi kerak. Misol uchun "cls1"ni o'rniga "p.cls1"ni yozing.

Agar ID(yoki sinf) o'z ichiga maxsus belgilarni olsa(nuqta yoki qavslar),ularning vazifasini //(slesh) orqali cheklashimiz mumkin.

```
$("#div1\\.index\\[5\\]").html("Matn");
```

Bu misolda biz yo'nalishni keyingi elementga o'tkazamiz:

```
<div id="div1.index[5]"></div>
```

Selektorlarni guruhlash. Hujjat elementlarining bog'lanishi: Agar zarur bo'lganda ko'rinishni turli elementlariga o'rnatishga to'g'rikelganda, vergullar bilan ajratish orqali bu ishni amalga oshira olamiz.

```
$("#idl,div").addClass("newClass");
```

Identifikatori mavjud bo'lgan bu misolda id1 element, shuningdek <div>elementidagi barcha teglar **Newclass** orqali amalga oshiriladi. HTMLning

boshqa elementlardagi munosabatini quyidagicha amalga oshiramiz.

Element1 Element2—konteyner ichida joylashgan Element1 ichidan Element2 ni topamiz.

```
$("#div a").css("color", "red");
```

<div> tegi <and> tegining ichida bo'lganligi uchun, tekstning yo'nalishi qizildir.

Element1 >Element2—Element2 ni topamiz, qaysiki Element1 uchun Element2, uning bolasi.

```
$("#div > a").css("color", "red");
```

<div> tegi <and> tegining ichida bo'lganligi va boshqa teg elementlari ichida bo'lmaganligi uchun tekstning yo'nalishi qizildir.

```
<div>
```

```
<a href="link1 .html">Yo'nalish 1</a><br>
```

```
<span><a href="link2 .html">Yo'nalish 2</a></span>
```

```
</div>
```

Bu misolda faqatgina 2ta yo'nalish qizil bo'ladi.

Ajax bilan tanishuv. Qo'llanilish sohasi

Qisqartirilgan Ajax «Asynchronous JavaScript and XML» («JavaScript va XML asinxronkodi») so'zidan olingan. Bu umumiy termin yuqori interaktivli dasturni bildirib, foydalanuvchi harakatlariga tez javob qaytaruvchi, ko'p amallarni klient kompyuterida bajaruvchi va server bilan aloqani tashki kanal yordamida amalga oshiradi. Tashqi aloqa deb (out-of-band) serverga bo'lgan so'rovni sahifaning operativ yangilanishiga aytiladi. Ajax dasturidagi natijalar Microsoft Windows klassik dasturlariga o'xshab asinxron ammallari kabi foydalanuvchini zeriktirmaydi.

Ajax texnologiyasi JavaScript-ob'ektlaridan XML Http Request qo'llanilishidan boshlangan. Nomidan ko'rinib turibdiki klient tomonidan HTTP – so'rovlarni qayta ishlab va XML-server javobini tahlil qiladi. Ajaxni o'rganishda birinchi qadam – XML Http Request ob'ektini yaratishdan iborat. So'rovlarni (GET yoki POST) qo'llashda HTTP protokoli metodi XML Http Request ob'ekti bilan bog'lanadi.

Ajax termini birinchi bo'lib fanga Djis Djeys Garret (Jesse James Garrett) asinxron JavaScript + XML texnologiyalari uchun kiritgan. Qisqa qilib aytganda Ajax sahifani yangilab o'tirmasdan tezda foydalanuvchi harakatiga javob qaytaradi. Ajaxni barcha asosiy brouzerlar qo'llab quvvatlaydi. Ajaxning mashhur bo'lishi Google kompaniyasi o'z saytlarni Gmail, Google Maps, Google Suggest kabilarni yaratishda keng ko'lamda bu texnologiyadan foydalandi. Ajaxning bu saytlarda qo'llanilishi bu texnologiyaning samarali ekanligini ko'rsatdi.

Oddiy standart yo`nalish va Ajaxni taqqoslash

Web-dasturlarning klassik modellari:

- Foydalanuvchi saytga kiradiva qaysidir element ustiga bosadi.
- Brauzer so'rovlarni formallashtiradiva serverga so'rovni jo'natadi.
- So'rovga javob sifatida server to'liq yangi web-sahifani generatsiya qiladi va klient brauzeriga jo'natadi va h.k.

Ajax modeli

- Foydalanuvchi saytga kiradi va biror element ustiga bosadi.
- Skript (JavaScript dagi skript) sahifani yangilash uchun qaysi ma'lumot kerakligini aniqlaydi.
- Brouzer mos so'rovni serverga jo'natadi.
- Server dokumentning so'rov kelgan qisminigina qaytaradi.
- Skript olingan ma'lumotni sahifaga joylashtiradi (sahifani to'liq yangilamasdan).

Ko'pincha server tayyor HTML-kod yubormasdan faqatgina ma'lumot yuboradi; berilgan ma'lumotlardan kelib chiqib DOM modifikasiya metodi qo'llangan holda HTML – elementlari yaratiladi. Ma'lumotlarni yuborish ko'rinishi sifatida XML va JSON qo'llaniladi.

JSON shunisi yaxshiki klient tomonidan ma'lumotlarni qayta ishlash kerak emas faqatgina eval() funksiyasini chaqirilsa bo'ldi. Lekin xavfsizlik nuqtai nazaridan malumotni tahlil qilish kerak chunki eval() funksiyasi har qanday kodni bajaradi shuningdek xavf tug'diruvchi kodni ham.

Ajax imkoniyatlari: Birinchi navbatda AJAX uchra katta bo'lmagan elementlar uchun korzinaga qo'shish, ovoz berish, a'zo bo'lish va boshq. Masalan, bosganda ochiladigan daraxt tanalari. Masalan maqolani tahrirlashda har 10 minutda serverda avto saqlash amali bajariladi. Eng tipik misol – chat. Uzluksiz yuklanuvchi serverda chat oynasida paydo bo'luvchi yangi xabarlar. Yoki shunga o'xshash joriy vaqtda birjadagi kotirovkalarni hosil qilish.

Google – birinchilardan bo'lib “tirik qidirish” , livesearch taklif qilgan. Foydalanuvchi qidirilayotgan frazani kiritaveradi, tizim esa eng ko'p qidirilishi mumkin bo'lgan qo'shimchalarni avtomatik tarzda serverdan olib qo'shib boradi.

Ajax mustaqil texnologiya bo'lmasdan bir nechta texnologiyalarning qo'shilmasidir. Ajax 2 ta asosiy printsipga asoslanadi:

- Sahifani qayta yuklamasdan serverga dinamik bog'lanish texnologiyasini qo'llanishi
- XMLHttpRequest ob'ekti(asosiy ob'ekt) ni qo'llash

Interfeys bilan amallar DOM(ing. *Document Object Model*) elementlari ustida amalga almashtirilib, foydalanuvchiga tegishli bo'lgan ma'lumotlar qayta ishlanadi va natijada ko'rinish o'zgartiriladi. Bu erda sichqoncha bilan ko'chish va klaviaturani bosish qayta ishlanadi. Kaskad jadvallar stili CSS(ing. *Cascading Style Sheets*) DOM ob'ektlariga murojaat qiluvchi elementlarning tashqi

ko'rinishini ta'minlaydi. XMLHttpRequest ob'ekti (yoki shunga o'xshash mexanizmlar) foydalanuvchi so'rovlarini qayta ishlashda va jarayon uchun kerakli ma'lumotlarni yuklashda server bilan asinxron aloqani tashkil etadi.

Bu to'rtta texnologiyadan uchasi – CSS, DOM va JavaScript va DHTML (ing. *Dynamic HTML*) dan iborat.

Ma'lumotlarni uzatish formati sifatida oddiy matn fragmentlari HTML-kodi, JSON yoki XML qo'llanilishi mumkin.

Ajax va PHP da parametrlarni yuborish usullari

GET va POSTHTTPso'rovlari :Gipermatnlarni uzatish protokoli tarmoq protokollari ichida eng sodda va qulay protokollardan hisoblanadi. Uning asosiy vazifasi «giperbog'lanish»dan hosil bo'lgan URL adresli elektron hujjatlarni o'qishga oid so'rov (zapro) ni serverga jo'natish (xuddi shu vaqtda so'ralayotgan hujjat joylashgan server bilan aloqa o'rnatiladi) va so'ralayotgan hujjat olib bo'lingandan so'ng server bilan aloqani uzishdan iborat. Internetda elektron hujjatlar gipermatn yordamida bayon etiladi. Gipermatn bu matnni giperbog'lanishlar yordamida ifodalashdir. Giperbog'lanishlar ichki va tashqi bo'lishi mumkin. Agar giperbog'lanish boshqa bir serverda mavjud bo'lgan alohida o'zining URL adresiga ega bo'lgan elektron hujjatga nisbatan ishlaydigan bo'lsa, u holda bunday giperbog'lanish tashqi deb ataladi. Ba'zi hollarda Web hujjat avtorlari qulaylik nuqtai nazaridan bir serverning o'zida joylashgan elektron hujjatning o'zini xam bir necha bo'laklarga bo'lib, giperbog'lanishlar yordamida ifodalaydilar, bunday giperbog'lanishlar ichki deb ataladi.

HTML formadan ma'lumotlar PHP skriptga yuborilganda forma avtomatik ravishda skriptga bog'lanadi. Ma'lumotlarni olishni bir necha usullari mavjud:

Masalan oddiy html forma:<form action="fos.php" method="post">

Name: <input type="text" name="username" />

Location: <input type="text" name="location" />

<input type="submit" name="submit" value="Send info!" />

</form>

PHP 5.4.0 versiyasidan boshlab ma'lumotni olishning faqat 2 turi mavjud. Mavjud metodlarni quyida keltirib o'tamiz:

Oddiy html formadan ma'lumotlarni olish:

```
<?php
echo $_POST['username'];
echo $_REQUEST['username'];
?>
```

PHP oldingi versiyalarida ma'lumotlar olishning bir necha usullari mavjud bo'lgan ulardan biri quyidagi misolda ko'rish mumkin:

```
<?php
// Diqqat: bu metodlar hozir ishlamaydi.
// import_request_variables() – bu funksiya PHP 5.4.0 dan keyin o`chirib tashlandi
import_request_variables('p', 'p_');
echo $p_username;
// bu uzun massiv ham PHP 5.4.0 dan keyin o`chirib tashlandi
echo $_HTTP_POST_VARS['username'];

echo $username;
?>
```

GET metodi ma'lumotlarni URL tarkibida ? belgisidan keyin jo'natadi (ma'lumotlarni ajratish uchun ampersand & belgisi ishlatiladi).

POST va GET metodlarini ishlashini amalda ko'rish uchun ma'lumotlarni kiritish uchun kattaroq formani yasaymiz:

```
<html>

<head>

<title>Ўзгарувчиларни Post орқали юбориш формаси</title>

</head>

<body>

<form method="post" action="index.php">Маълумотларни юбориш
майдонларини тўлдирамиз:<br>

Исмингизни кiritинг: <input name="user_name" type="text" maxlength="20"
size="25" value="" />

<br><br>Ёшингиз: <input name="age" type="text" maxlength="2" size="3"
value="" />

<br><br><input type="submit" value="Маълумотни юбориш"></form>

</body>

</html>
```

AJAX bilan birgalikda so'rovda turli ma'lumotlar jo'natish mumkin.

Misol sifatida oddiy kalkulyator uchun ikkita sonni qo'shishni ko'ramiz.

```
var x=document.getElementById('tf1').value;
```

```
var y=document.getElementById('tf2').value;
```

```
xhttp=new XMLHttpRequest();
```

```
xhttp.onreadystatechange=function(){
```

```
if (xhttp.readyState==4 && xhttp.status==200)

document.getElementById("ajax").innerHTML=xhttp.responseText;

}

xhttp.open("GET","add.php?x="+x+"&y="+y,true);

xhttp.send();
```

OddiyPOST so'rovga misolsifatida test file.txt faylinichaqirishva uningichidagima'lumotlarni chiqarish:

```
xhttp=newXMLHttpRequest();
xhttp.onreadystatechange=function(){
if(xhttp.readyState==4&& xhttp.status==200)
document.getElementById("result").innerHTML=xhttp.responseText;
}
xhttp.open("POST","testfile.txt",true);
xhttp.send();
```

Ajax sinxron so'rovlarni PHP yordamida qayta ishlash

Ajax bilan PHP birgalikda ishlashi quyidagi sohalarda birga qo'llash mumkin:

1. Saytga yangi izoh qo'shish qismlarida
2. Ovoz berish
3. Sayt avtorizatsiya qismlarida
4. Saytda qidirish masalalarida
5. E-mail ni a'zo qilish
6. Albomlarni tomosha qilishda va boshqa qismlarda...

Ko`rib turganingizdek ajax + php ni qo`llash variantlari juda ko`p. Ya`ni server bilan ma`lumotni almashishda sahifani yuklash kerak bo`lmagan joylarda qo`llash mumkin.

Ma`lumotlarni serverga jo`natishda birinchi XMLHttpRequest ob`yektini yaratish kerak. Uning yordamida URL(php script) ni ochish, unga ma`lumotni jo`natish (POST yoki GET metodi bilan), javobni olish, va javascript tili yordamida serverdan olingan javobni ekranga chiqarish(javob ixtiyoriy fragment yoki sahifa elementi bo`lishi mumkin).

Xulosa

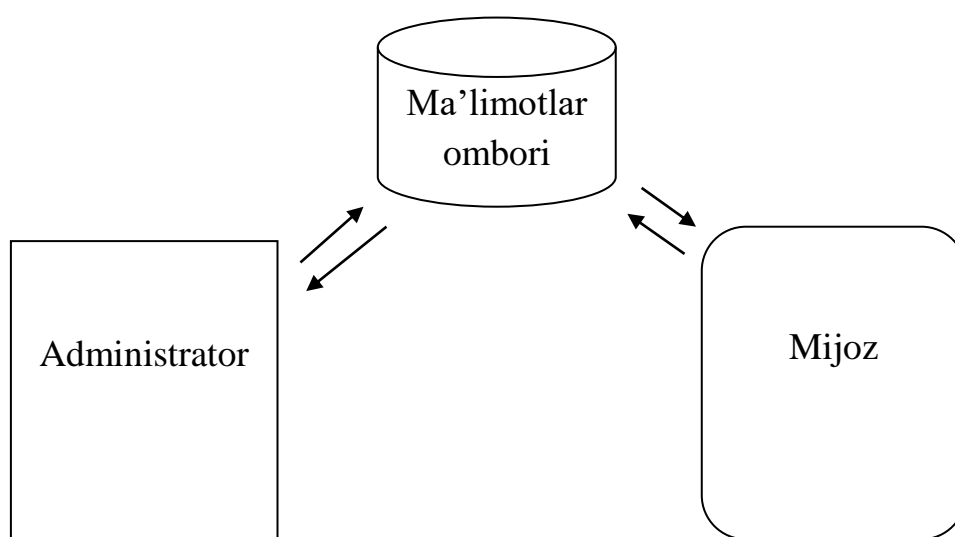
Ushbu bobda zamonaviy web texnologiyalarning turlari, ularning ma`lumotlar ombor bilan bog`lanishi kabi nazariy bilimlar to`plandi, o`rganildi va tizimlashtirildi. BMI uchun kerakli nazariy bilimlar, amaliy ko`rsatmalar tahlil qilindi. II bobni bajarish uchun poydevor tayyorlandi.

II BOB. ONLAYN BUYURTMA OLISH TIZIMINI TASHKIL ETISH JARAYONI

II.1. Onlayn buyurtma olish tizimining server qismi, ma'lumotlar omborini shakllantirish

Online buyurtma qabul qilish tizimi 3 qismdan tashkil topgan bo'ladi:

1. Server qismi (ma'lumotlar ombori)
2. Administrator qismi
3. Mijoz qismi



2.1.1-rasm. Online buyurtma qabul qilish tizimining sxemasi

Ushbu tizimning ma'lumotlar ombori MySQL da tayyorlanadi. Buning uchun bizdan PHP va MySQLni qo'llab quvvatlaydigan lokal server yoki Linux operatsion tizimli virtual server zarur bo'ladi.

Ma'lumotlar ombori (keying joylarda – baza)ning tarkibiy qismi quyidagicha bo'ladi.

1. Baza nomi – “tssystem_db6”
2. Foydalanuvchi nomi – “tssystem_db6”
3. Parol – “m2017”

Jadvallar:

1. Xizmatlar yoki mahsulotlar ro'yxati – sp_tovari

2. Kategoriyalar ro'yxati – sp_vid_tovari
3. Narxlar – sp_seni
4. Buyurtmalar – j_zakaz

Ushbu ma'lumotlar ombori orqali joylashgan hududdagi barcha sotuv shaxobchalarini, mahsulotlar nomlari, ularning miqdori, kattaliklari, narxlari, buyurtmalar, vaqti, yetkazib berish sanasi, holatlarini ko'rish mumkin.

Quyida ushbu bazani yaratish kodlari keltiriladi:

```
-- phpMyAdmin SQL Dump
-- version 4.6.5.2
-- https://www.phpmyadmin.net/
CREATE TABLE `j_detal_pereosenka` (
  `id` int(5) NOT NULL,
  `j_pereosenka_id` int(4) NOT NULL,
  `tovar_id` int(4) NOT NULL,
  `tip_seni` int(3) NOT NULL,
  `narx` int(8) NOT NULL,
  `primechanie` varchar(255) NOT NULL,
  `rozn` int(8) NOT NULL DEFAULT '0',
  `opt` int(8) NOT NULL DEFAULT '0',
  `eks` int(8) NOT NULL DEFAULT '0',
  `prixod` int(8) NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `j_invoice` (
  `id` int(6) NOT NULL,
  `data` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `sotrudnik_id` int(3) NOT NULL DEFAULT '1',
  `torgovaya_tochka_id` int(5) NOT NULL,
  `tip_seni_id` int(3) NOT NULL,
  `summa` int(11) NOT NULL,
```

```

`zakaz_vozvrat` tinyint(1) NOT NULL,
`primechanie` varchar(255) NOT NULL,
`pereosenka_id` int(5) NOT NULL,
`forma_oplati` int(3) NOT NULL,
`skidka` int(3) NOT NULL DEFAULT '0',
`v_invoice_id` int(5) NOT NULL,
`primechanie2` varchar(255) DEFAULT NULL,
`status` int(1) NOT NULL DEFAULT '1'
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `j_oplata` (
  `id` int(6) NOT NULL,
  `data` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `torgovaya_tochka_id` int(5) NOT NULL,
  `sotrudnik_id` int(3) NOT NULL,
  `forma_oplati_id` int(3) NOT NULL,
  `primechanie` varchar(255) NOT NULL,
  `summa` int(8) NOT NULL,
  `dostavshik_id` int(3) NOT NULL DEFAULT '0',
  `agentid_o_d` int(5) NOT NULL,
  `invoice_id` int(7) NOT NULL,
  `realdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `j_pereosenka` (
  `id` int(4) NOT NULL,
  `data` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `sotrudnik_id` int(3) NOT NULL,
  `primechanie` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```
CREATE TABLE `j_periki` (  
  `id` int(5) NOT NULL,  
  `torgovaya_tochka_id` int(5) NOT NULL,  
  `data` date NOT NULL,  
  `summa` int(11) NOT NULL,  
  `dostavshik_id` int(3) NOT NULL,  
  `invoice_id` int(5) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `j_plan` (  
  `id` int(5) NOT NULL,  
  `agent_id` int(5) NOT NULL,  
  `vid_tovara_id` varchar(255) NOT NULL,  
  `data` date NOT NULL,  
  `summa` int(12) NOT NULL,  
  `primechanie` varchar(255) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `j_postuplenie` (  
  `id` int(3) NOT NULL,  
  `data` date NOT NULL,  
  `postavshik_id` int(3) NOT NULL,  
  `sklad_id` int(3) DEFAULT NULL,  
  `summa` int(9) NOT NULL,  
  `primechanie` varchar(255) NOT NULL,  
  `pereosenka_id` int(5) NOT NULL DEFAULT '1'  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `j_prixod` (  

```

```
`id` int(3) NOT NULL,  
`postuplenie_id` int(3) NOT NULL,  
`tovar_id` int(3) NOT NULL,  
`kolichество` float NOT NULL,  
`primechanie` varchar(255) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `j_rasxodi` (  
`id` int(11) NOT NULL,  
`naimenovanie` varchar(255) NOT NULL,  
`summa` int(7) NOT NULL,  
`dostavshik_id` int(3) NOT NULL,  
`data` date NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `j_zakaz` (  
`id` int(11) NOT NULL,  
`invoice_id` int(6) NOT NULL,  
`tovari_id` int(3) NOT NULL,  
`kolichество` float NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `nastroyki` (  
`parametr` varchar(20) CHARACTER SET utf8 NOT NULL,  
`znachenie` varchar(255) CHARACTER SET utf8 DEFAULT NULL  
)  
ENGINE=InnoDB DEFAULT CHARSET=cp1251 COLLATE=cp1251_bin;
```

```
CREATE TABLE `n_marshrut` (  
`id` int(3) NOT NULL,
```

```
`agentid` int(3) NOT NULL,  
  `yakshanba` varchar(25) DEFAULT '0',  
  `dushanba` varchar(25) NOT NULL DEFAULT '0',  
  `seshanba` varchar(25) NOT NULL DEFAULT '0',  
  `chorshanba` varchar(25) NOT NULL DEFAULT '0',  
  `payshanba` varchar(25) NOT NULL DEFAULT '0',  
  `juma` varchar(25) NOT NULL DEFAULT '0',  
  `shanba` varchar(25) NOT NULL DEFAULT '0',  
  `data` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sp_doljnosti` (  
  `id` int(3) NOT NULL,  
  `naimenovanie` varchar(255) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sp_dostavshik` (  
  `id` int(3) NOT NULL,  
  `naimenovanie` varchar(50) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sp_edinisi` (  
  `id` int(3) NOT NULL,  
  `naimenovanie` varchar(255) NOT NULL,  
  `opisanie` varchar(255) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sp_forma_oplati` (  
  `id` int(3) NOT NULL,  
  `naimenovanie` varchar(255) NOT NULL
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sp_postavshiki` (  
  `id` int(3) NOT NULL,  
  `naimenovanie` varchar(255) NOT NULL,  
  `direktor` varchar(25) DEFAULT NULL,  
  `telefon` varchar(25) DEFAULT NULL,  
  `adres` varchar(255) DEFAULT NULL,  
  `rekvizit` varchar(255) DEFAULT NULL,  
  `nomer_dogovora` varchar(255) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sp_seni` (  
  `id` int(3) NOT NULL,  
  `naimenovanie` varchar(255) NOT NULL,  
  `prosent` int(3) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sp_sklad` (  
  `id` int(3) NOT NULL,  
  `naimenovanie` varchar(255) NOT NULL,  
  `adres` varchar(255) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sp_territoriya` (  
  `id` int(3) NOT NULL,  
  `naimenovanie` varchar(255) NOT NULL,  
  `primechanie` varchar(255) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```

CREATE TABLE `sp_torgovie_tochki` (
  `id` int(5) NOT NULL,
  `naimenovanie` varchar(255) NOT NULL,
  `klyuch_liso` varchar(255) DEFAULT NULL,
  `territoriya_id` int(3) NOT NULL,
  `adres` varchar(255) DEFAULT NULL,
  `orientir` varchar(255) DEFAULT NULL,
  `telefon` varchar(25) DEFAULT NULL,
  `status` varchar(3) NOT NULL DEFAULT '0',
  `tip_t_t_id` int(3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `sp_tovari` (
  `id` int(3) NOT NULL,
  `naimenovanie` varchar(255)
  CHARACTER SET koi8r COLLATE koi8r_bin NOT NULL,
  `vid_tovarov_id` int(3) NOT NULL,
  `Yedinisi_id` int(3) NOT NULL,
  `Kol_v_korob` float NOT NULL,
  `ostatok` float DEFAULT '0',
  `status` int(1) NOT NULL DEFAULT '1'
)ENGINE=InnoDB DEFAULT CHARSET=cp1251 COLLATE=cp1251_bin;

```

```

CREATE TABLE `sp_vid_tovarov` (
  `id` int(3) NOT NULL,
  `naimenovanie` varchar(255) NOT NULL,
  `postavshik_id` int(3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `tip_t_t` (

```

```

`id` int(3) NOT NULL,
`naimenovanie` varchar(25) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `users` (
  `id` int(11) NOT NULL,
  `naimenovanie` varchar(100) COLLATE cp1251_bin DEFAULT NULL,
  `username` varchar(64) COLLATE cp1251_bin DEFAULT NULL,
  `password` varchar(64) COLLATE cp1251_bin DEFAULT NULL,
  `sid` varchar(64) COLLATE cp1251_bin DEFAULT NULL,
  `doljnost_id` int(3) NOT NULL,
  `data_priyoma` date NOT NULL,
  `status` int(1) NOT NULL DEFAULT '1',
  `telefon` varchar(50) COLLATE cp1251_bin DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=cp1251
COLLATE=cp1251_bin;

```

Administratorlik qismining asosiy jadvallari JQGrid kutubxonasi orqali shakllantirilgan bo'lib, bunda jadvalning asosiy elementlari ko'rsatiladi. Masalan, Mahsulotlar nomli jadvalda quyidagi sozlanishlar keltirilgan. Jadvaldagi har bir ustun uchun ishlatiladigan sozlamalar:

```

$col = array(); // Massivni e'lon qilish
$col["dbname"] = "i.id"; //So'rovdagi ustun nomi
$col["title"] = "Id"; // Jadvaldagi ustun nomi
$col["name"] = "id"; //
$col["width"] = "20"; //Ustun kengligi
$col["align"] = "center"; //Tekislash
$col["editable"] = false; // Ustunni o'zgartirish imkoniyati
$cols[] = $col;

```

Jadval sozlamalari:

```

$g->set_actions(array(
    "add"=>true, // Qo'shishga ruxsat
    "edit"=>true, // O'zgartirishga ruxsat
    "delete"=>true, // O'chirishga ruxsat
    "view"=>true, //
    "rowactions"=>true, // O'chirish, qo'shish va o'zgartirish tugmalari
    "search" => "advance", // Qidiruv
    "showhidecolumns" => false,
    "export" => true,
    "inlineadd" => true,
    "export_excel"=>true, // jadvalni excelga eksport qilish
    "export_pdf"=>true, // jadvalni pdfga eksport qilish
    "export_csv"=>true // jadvalni csvga eksport qilish));

```

II.2. Onlayn buyurtma olish tizimining mijoz qismini shakllantirish

Ushbu tizimning mijoz qismi soddaroq ko'rinishda bo'lib, asosan mijozni tanlash va mijoz uchun mahsulotlarni belgilashdan iborat bo'ladi. Buyurtma qabul qilayotgan savdo xodimi eng avvalo o'z login va parolini terishi, so'ngra joriy kunda qabul qilingan buyurtmalar ro'yxatiga tushishi mumkin. Quyida mahsulotni tanlash sahifasi keltirilgan (detal_zakaz.php)

```

<?php include("blocks/connect.php");
include("functions.php");
include("login.php");
if (!isset($_GET['tsena'],$_GET['tt'])){die('parametrlar toliq emas');}
$tip_narx = $_GET['tsena'];
$tt = $_GET['tt'];
$pereosenkaq = mysql_query("SELECT id FROM j_pereosenka ORDER BY
id DESC LIMIT 1",$db);
$pereosenka = mysql_fetch_array($pereosenkaq);
$today = date('d.m.y');

```

```

$day = date('D');
?><form name="zakaz" id="zakaz" action="zakaz_send.php?tt=<?php echo
$tt;?>&n=<?php echo $_GET['tsena'];?>&p=<?php echo $pereosenka['id'];?>"
method="post">
<div align="center"><b>
<?php
$result=mysql_query("SELECT * FROM tsystem_db1.sp_torgovie_tochki
WHERE id=$tt",$db1);
$myrow=mysql_fetch_array($result);
print("<input name='tt' type='hidden' disabled value='". $tt." -
".$myrow['naimenovanie']."'>");
print("<div class='copyright' align=center>".$myrow['id']."
".$myrow['naimenovanie']."'</div>");
?>
</b></div>
<table class="table" border="1">
<tr><th width="50%">Mahsulot</th><th width="20%">Narx</th><th
style="display:none">Кол. в коп.</th></tr>
<br>
<?php
$vid_tovara_q = mysql_query("SELECT id,naimenovanie from
sp_vid_tovarov");
$vid_tovara = mysql_fetch_array($vid_tovara_q);
do{
$result=mysql_query("SELECT * FROM sp_tovari WHERE (
SELECT(SELECT IFNULL(SUM(j_prixod.kolichestvo),0) FROM j_prixod
WHERE j_prixod.tovar_id = sp_tovari.id AND j_prixod.postuplenie_id IN
(SELECT j_postuplenie.id FROM j_postuplenie)) - (SELECT
IFNULL(SUM(j_zakaz.kolichestvo),0) FROM j_zakaz WHERE j_zakaz.tovari_id
= sp_tovari.id AND j_zakaz.invoice_id IN (SELECT j_invoice.id FROM

```

```

j_invoice))
    as ostatok) >0 AND sp_tovari.vid_tovarov_id = ".$vid_tovara['id'],$db);
    if (mysql_num_rows($result)!=0){
        $myrow=mysql_fetch_array($result);
        echo "<tr><td colspan='2' style='font-size: 15px; color:#ffffff; background-
color:#393393'>".$vid_tovara['naimenovanie']."</td></tr>";
        do {
            $narx = GetNarx($myrow['id'],$pereosenka['id'],$stip_narx);
            $spisok_tovar="<tr style='border-top: solid 2px #000000' >
            <td><b>%s. %s</b></td>
            <td><input name='%s-narx' id='%s-narx' value='%s' type='hidden'>Цена:
            %s</td>
            <td style='display:none'>КОЛ-ВО В КОР: %s</td>
            </tr>
            <tr style='border-bottom:solid 2px #000000'>
            <td>ОСТАТОК: %s</td>
            <td>
            <label class='buttonmini' onClick=Plyus('%s',%s);>+</label>
            <input name='%s-kolichestvo' type='number' id='%s-kolichestvo' min='0'
            max='%s' step='%s' onChange='summa(%s)' style='width:50px; font-size = 15px;'
            value='0'>
            <label class='buttonmini' onClick=Minus('%s',%s);>-</label>
            </td>
            <td style='display:none'><input name='%s-summa' id='%s-summa' value='0'
            type='text' size='20' style='width:100px;' readonly></td>
            </tr>" ;
            printf($spisok_tovar,$myrow['id'],$myrow['naimenovanie'],
            $myrow['id'],$myrow['id'],$narx,$narx,
            $myrow['Kol_v_korob'],
            GetOstatok($myrow['id']),

```

```

$myrow['id'],$myrow['Kol_v_korob'],
$myrow['id'],$myrow['id'],GetOstatok($myrow['id'],$myrow['Kol_v_korob'],
$myrow['id'],
$myrow['id'],$myrow['Kol_v_korob'],
$myrow['id'],$myrow['id']);
}
while($myrow=mysql_fetch_array($result));
}}
while($vid_tovara = mysql_fetch_array($vid_tovara_q));
?>
<tr height="40"><td colspan=2><br><br></td></tr>
<input type="hidden" value="<?php echo $tip_narx;?>" name="tsena">
<tr class="fixirovka"><td>Итого: <input type="text" name="itog" id="itog"
value="0" style='width:100px; ' readonly></td><td><input type="submit"
value="ОТПРАВИТЬ"></td>
<td style="display:none"></td></tr>
</table>
</form>
<script>
function summa(id) {
    document.getElementById('itog').value =
document.getElementById('itog').value - document.getElementById(id+'-
summa').value;
    document.getElementById(id+'-summa').value =
document.getElementById(id+'-narx').value*document.getElementById(id+'-
kolichество').value;
    document.getElementById('itog').value =
parseInt(document.getElementById('itog').value,10) +
parseInt(document.getElementById(id+'-summa').value,10);
};

```

```

function Plyus(id,kol){
    if    (parseFloat(document.getElementById(id+'-kolichestvo').value)    +
parseFloat(kol)<=document.getElementById(id+'-kolichestvo').max)
        { document.getElementById(id+'-kolichestvo').value                =
parseFloat(document.getElementById(id+'-kolichestvo').value) + parseFloat(kol);
        summa(id);}
    };
function Minus(id,kol){
    if    (parseFloat(document.getElementById(id+'-kolichestvo').value)    -
parseFloat(kol)>=document.getElementById(id+'-kolichestvo').min)
        { document.getElementById(id+'-kolichestvo').value                =
parseFloat(document.getElementById(id+'-kolichestvo').value) - parseFloat(kol);
        summa(id);}
    };
</script>
<!doctype html>
<html>
<head>
<metacharset="utf-8">
<title>Mahsulotlarni tanlash oynasi</title>
</head>
<body>
</body>
</html>

```

E'tibor bersak, mahsulotlar sonini oshirish va kamaytirish uchun maxsus qo'shish va ayirish tugmalari qo'yilgan. Ular esa o'z navbatida JavaScriptda yozilgan Plyus() va Minus() funksiyalariga murojaat qilmoqda.

II.3. Onlayn buyurtma tizimida hisobotlarni shakllantirish

Onlayn buyurtma qabul qilishda hisoblar asosan quyidagilardan iborat bo'ladi:

1. Har bir mijoz bo'yicha buyurtma mahsulotlari ro'yxati
2. Har kungi omborxonadan chiqishi lozim bo'lgan mahsulotlar ro'yxati
3. Mijozlarning tizimdagi qarzdorligi

Ushbu hisobotlar MS Excel fayli formatida eksport qilinadi. Shu sababli php tilida PHPExcel nomli maxsus kutubxonadan foydalanamiz.

Mijoz bo'yicha buyurtma mahsulotlari ro'yxati
(obshiy_itog_period_report2.php)

```
<?php include("blocks/connect.php");
include("functions.php");
include ("Classes/PHPExcel.php");?>
<?php
if(isset($_POST['date1'])){ $data1 = $_POST['date1'];}
if(isset($_POST['date2'])){ $data2 = $_POST['date2'];}
if(isset($_POST['agent']))
{ if ($_POST['agent']=="all"){ $agentzapro = "";}else{ $agentzapro = " AND
(sotrudnik_id = '".$_POST['agent'].") ";} }
if(isset($_POST['zakaz_vozvrat'])){ $zakaz_vozvrat =
$_POST['zakaz_vozvrat'];}
if ($zakaz_vozvrat == -1){ $zakaz_vozvrat_q = " (zakaz_vozvrat > -1) ";} else
{ $zakaz_vozvrat_q = "(zakaz_vozvrat = '".$_POST['zakaz_vozvrat'].") ";}
$zapros = "SELECT * FROM j_invoice WHERE
".$_POST['zakaz_vozvrat_q'].$agentzapro." AND (DATE_FORMAT(data,'% Y-%m-%d')
BETWEEN '".$_POST['date1']."' AND '".$_POST['date2'].") ORDER BY data ASC";
$invoicequery = mysql_query($zapros);
$invoices = mysql_fetch_array($invoicequery);
if (!$invoices) {die(mysql_error().mysql_errno());}
$validLocale = PHPExcel_Settings::setLocale('ru');
```

```

$objPHPExcel = new PHPEXcel();
$objPHPExcel->setActiveSheetIndex(0);
//$objPHPExcel->createSheet();
$active_sheet = $objPHPExcel->getActiveSheet();
$active_sheet->getPageSetup()
    >setOrientation(PHPEXcel_Worksheet_PageSetup::ORIENTATION_PORTRAIT);
$active_sheet->getPageSetup()
->setPaperSize(PHPEXcel_Worksheet_PageSetup::PAPERSIZE_A4);
$active_sheet->getPageSetup()->setFitToPage(true);
$active_sheet->getPageSetup()->setFitToWidth(1);
$active_sheet->getPageSetup()->setFitToHeight(0);
$active_sheet->getPageMargins()->setTop(0.5);
$active_sheet->getPageMargins()->setRight(0.3);
$active_sheet->getPageMargins()->setLeft(0.3);
$active_sheet->getPageMargins()->setBottom(0.5);
$active_sheet->setTitle("Инвойс");
//$active_sheet->getHeaderFooter()->setOddHeader("&CШпка нашего прайс листа");
//$active_sheet->getHeaderFooter()->setOddFooter('&L&B'. $active_sheet->getTitle(). '&RСтраница &P из &N');
$objPHPExcel->getDefaultStyle()->getFont()->setName('Arial');
$objPHPExcel->getDefaultStyle()->getFont()->setSize(8);
$objPHPExcel->getDefaultStyle()->getAlignment()->setWrapText(true);
$active_sheet->getColumnDimension('A')->setWidth(10);
$active_sheet->getColumnDimension('B')->setWidth(35);
$active_sheet->getColumnDimension('C')->setWidth(7);
$active_sheet->getColumnDimension('D')->setWidth(12);
$active_sheet->getColumnDimension('E')->setWidth(11);
$active_sheet->getColumnDimension('F')->setWidth(13);

```

```

$active_sheet->getColumnDimension('G')->setWidth(13);
$active_sheet->getColumnDimension('H')->setWidth(17);
$active_sheet->getColumnDimension('I')->setWidth(2);
$active_sheet->getColumnDimension('J')->setWidth(10);
$active_sheet->getColumnDimension('K')->setWidth(35);
$active_sheet->getColumnDimension('L')->setWidth(7);
$active_sheet->getColumnDimension('M')->setWidth(12);
$active_sheet->getColumnDimension('N')->setWidth(11);
$active_sheet->getColumnDimension('O')->setWidth(13);
$active_sheet->getColumnDimension('P')->setWidth(13);
$active_sheet->getColumnDimension('Q')->setWidth(17);
$stop_bold = array('font'=>array('bold' => true,'name' => 'Arial','size' => 11),
    'alignment'=>array('horizontal'=>
PHPExcel_STYLE_ALIGNMENT::HORIZONTAL_RIGHT,
    'vertical'=>
PHPExcel_STYLE_ALIGNMENT::VERTICAL_CENTER
    )
); $stop_simple = array(
    'font'=>array(
        'bold' => false,
        'name' => 'Arial',
        'size' => 12),
    'alignment'=>array('horizontal'=>
PHPExcel_STYLE_ALIGNMENT::HORIZONTAL_LEFT,
    'vertical'=>
PHPExcel_STYLE_ALIGNMENT::VERTICAL_CENTER));
$inv_num = array(
    'font'=>array(
        'bold' => true,
        'name' => 'Calibri',

```

```

        'size' => 12),
    'alignment' => array(
'horizontal' => PHPExcel_STYLE_ALIGNMENT::HORIZONTAL_CENTER,
'vertical' => PHPExcel_STYLE_ALIGNMENT::VERTICAL_CENTER),
    'borders' => array(
        'allborders' => array(
            'style'=>PHPExcel_Style_Border::BORDER_MEDIUM
        )
    )
); $spisok_tovari_toq = array(
    'font'=>array(
        'bold' => false,
        'name' => 'Calibri',
        'size' => 14
    ),
    'alignment' => array(
        'horizontal'=>
PHPExcel_STYLE_ALIGNMENT::HORIZONTAL_CENTER,
        'vertical'=>
PHPExcel_STYLE_ALIGNMENT::VERTICAL_CENTER    ),
    'borders' => array(
        'vertical' => array(
            'style'=>PHPExcel_Style_Border::BORDER_MEDIUM
        ),
        'bottom' => array(
            'style'=>PHPExcel_Style_Border::BORDER_THIN
        )
    )
);

```

```

$spisok_tovari_juft = array(
    'font'=>array(
        'bold' => false,
        'name' => 'Calibri',
        'size' => 14
    ),
    'alignment' => array(
        'horizontal'=>
        PHPExcel_STYLE_ALIGNMENT::HORIZONTAL_CENTER,
        'vertical'=>
        PHPExcel_STYLE_ALIGNMENT::VERTICAL_CENTER
    ),
    'borders' => array(
        'vertical' => array(
            'style'=>PHPExcel_Style_Border::BORDER_MEDIUM
        ),
        'bottom' => array(
            'style'=>PHPExcel_Style_Border::BORDER_THIN
        )
    ),
    'fill'=>array(
        'type' => PHPExcel_Style_Fill::FILL_SOLID,
        'color' => array('rgb' => 'cccccc')
    )
);

$tovar_name_left = array(
    'alignment' => array(
        'horizontal'
        PHPExcel_STYLE_ALIGNMENT::HORIZONTAL_LEFT,
        'vertical'

```

```
PHPExcel_Style_Alignment::VERTICAL_CENTER
```

```
)  
);
```

```
$spisok_tovari_vertical = array(  
    'borders' => array(  
        'left' => array(  
            'style' => PHPExcel_Style_Border::BORDER_MEDIUM  
        ),  
        'right' => array(  
            'style' => PHPExcel_Style_Border::BORDER_MEDIUM  
        )  
    )  
);
```

```
$style_primechanie = array(  
    'font' => array(  
        'bold' => false,  
        'name' => 'Arial',  
        'size' => 14  
    ),  
    'alignment' => array(  
        'horizontal' => PHPExcel_Style_Alignment::HORIZONTAL_LEFT,  
        'vertical' => PHPExcel_Style_Alignment::VERTICAL_CENTER  
    )  
);
```

```
=>
```

```
PHPExcel_Style_Alignment::HORIZONTAL_LEFT,
```

```
=>
```

```
PHPExcel_Style_Alignment::VERTICAL_CENTER
```

```
),  
    'borders' => array(  
        'outline' => array(  
            'style' => PHPExcel_Style_Border::BORDER_MEDIUM  
        )  
    )  
);
```

```
)  
)  
);
```

```
$style_itog = array(  
    'font'=>array(  
        'bold' => true,  
        'name' => 'Calibri',  
        'size' => 16 ),  
    'alignment' => array(  
        'horizontal'=>  
PHPExcel_STYLE_ALIGNMENT::HORIZONTAL_CENTER,  
        'vertical'=>  
PHPExcel_STYLE_ALIGNMENT::VERTICAL_CENTER  
    ),  
    'borders' => array(  
        'outline' => array(  
            'style'=>PHPExcel_Style_Border::BORDER_MEDIUM  
        )  
    )  
);
```

```
$style_bonus = array(  
    'font'=>array(  
        'italic' => true  
    )  
);
```

```
$summa_k_oplate = array(  
    'font'=>array(  
        'bold' => true,
```

```

        'name' => 'Calibri',
        'size' => 20
    )
);

$border_thin_med = array(
    'borders' => array(
        'outline' => array(
            'style'=>PHPExcel_Style_Border::BORDER_MEDIUM
        ),
        'inside' => array(
            'style'=>PHPExcel_Style_Border::BORDER_THIN
        )
    )
);

//По каждой строке по циклу вставляется значение
$row_start = 1;
$massiv[][] = array();
do
{
    //Информация агента
    $agent_query = mysql_query("SELECT * FROM users WHERE uid
=".$invoices['sotrudnik_id']);
    $agentinfo = mysql_fetch_array($agent_query);
    if ($invoices['forma_oplati']==2){$forma_oplati = " Перечисление";}
    if ($invoices['forma_oplati']==1){$forma_oplati = " Н";}
    if ($invoices['forma_oplati']==3){$forma_oplati = " Терминал";}
    $tt_query = mysql_query("SELECT * FROM
tsystem_db1.sp_torgovie_tochki WHERE tsystem_db1.sp_torgovie_tochki.id
=".$invoices['torgovaya_tochka_id'],$db1);

```

```

$tt = mysql_fetch_array($tt_query);
$ster_query = mysql_query("SELECT * FROM tsystem_db1.sp_territoriya
WHERE tsystem_db1.sp_territoriya.id = ".$tt['territoriya_id'],$db1);
$ster = mysql_fetch_array($ster_query);
$active_sheet->getRowDimension($row_start)->setRowHeight(20);
$active_sheet->getRowDimension($row_start+1)->setRowHeight(20);
$active_sheet->getRowDimension($row_start+2)->setRowHeight(20);
$active_sheet->mergeCells('C'.$row_start.':D'.($row_start+2));
$active_sheet->mergeCells('F'.$row_start.':H'.$row_start);
$active_sheet->mergeCells('F'.($row_start+1.':H'.($row_start+1));
$active_sheet->mergeCells('F'.($row_start+2.':H'.($row_start+2));
$active_sheet->mergeCells('O'.$row_start.':Q'.$row_start);
$active_sheet->mergeCells('O'.($row_start+1.':Q'.($row_start+1));
$active_sheet->mergeCells('O'.($row_start+2.':Q'.($row_start+2));
//Первая часть шапки
$active_sheet->setCellValue('A'.$row_start,'Дата:');
$active_sheet->setCellValue('A'.($row_start+1),'Агент:');
$active_sheet->setCellValue('A'.($row_start+2),'Тел.:');
$active_sheet-
>setCellValue('B'.$row_start,date('d.m.Y',strtotime($invoices['data'])));
$active_sheet->setCellValue('B'.($row_start+1),$agentinfo['naimenovanie']);
$active_sheet->setCellValue('B'.($row_start+2),$agentinfo['telefon']);

$active_sheet->setCellValue('C'.$row_start,"Inv.
".$invoices['id'].$forma_oplati);

$active_sheet->setCellValue('E'.$row_start,'Клиент:');
$active_sheet->setCellValue('E'.($row_start+1),'Тел:');
$active_sheet->setCellValue('E'.($row_start+2),'Op-p.:');
$active_sheet->setCellValue('F'.$row_start,$tt['id']." ".$tt['naimenovanie']);

```

```

$active_sheet->setCellValue('F'.($row_start+1),$tt['telefon']);
$active_sheet->setCellValue('F'.($row_start+2),$tt['orientir']);

//Вторая часть шапки
$active_sheet->setCellValue('J'. $row_start,'Дата:');
$active_sheet->setCellValue('J'.($row_start+1),'Агент:');
$active_sheet->setCellValue('J'.($row_start+2),'Тел.:');
$active_sheet-
>setCellValue('K'. $row_start,date('d.m.y',strtotime($invoices['data'])));
$active_sheet->setCellValue('K'.($row_start+1),$agentinfo['naimenovanie']);
$active_sheet->setCellValue('K'.($row_start+2),$agentinfo['telefon']);

$active_sheet->setCellValue('N'. $row_start,'Клиент:');
$active_sheet->setCellValue('N'.($row_start+1),'Тел:');
$active_sheet->setCellValue('N'.($row_start+2),'Тер.:');
$active_sheet->setCellValue('O'. $row_start,$tt['id']. " ". $tt['naimenovanie']);
$active_sheet->setCellValue('O'.($row_start+1),$tt['telefon']);
$active_sheet->setCellValue('O'.($row_start+2),$ter['naimenovanie']);
$active_sheet->getStyle('A'. $row_start.':A'.($row_start+2))-
>applyFromArray($top_bold);
$active_sheet->getStyle('E'. $row_start.':E'.($row_start+2))-
>applyFromArray($top_bold);
$active_sheet->getStyle('J'. $row_start.':J'.($row_start+2))-
>applyFromArray($top_bold);
$active_sheet->getStyle('N'. $row_start.':N'.($row_start+2))-
>applyFromArray($top_bold);
$active_sheet->getStyle('B'. $row_start.':B'.($row_start+2))-
>applyFromArray($top_simple);
$active_sheet->getStyle('F'. $row_start.':F'.($row_start+2))-
>applyFromArray($top_simple);

```

```

    $active_sheet->getStyle('K'.$row_start.':K'.($row_start+2))-
>applyFromArray($top_simple);
    $active_sheet->getStyle('O'.$row_start.':O'.($row_start+2))-
>applyFromArray($top_simple);
    $active_sheet->getStyle('C'.$row_start.':D'.($row_start+2))-
>applyFromArray($inv_num);
    $row_start = $row_start + 2;

```

//Применяем стиль к первому строку

```

$style_first = array(
    'font'=>array(
        'bold' => true,
        'name' => 'Calibri',
        'size' => 12
    ),
    'alignment' => array(
        'horizontal'=>
PHPEXcel_Style_Alignment::HORIZONTAL_CENTER,
        'vertical'=>
PHPEXcel_Style_Alignment::VERTICAL_CENTER
    ),
    'borders' => array(
        'allborders' => array(
            'style'=>PHPEXcel_Style_Border::BORDER_THIN
        )
    )
);

$style_date = array(
    'font'=>array(

```

```

        'bold' => false,
        'name' => 'Calibri',
        'size' => 8
    ),
    'alignment' => array(
        'horizontal'=>
PHPExcel_Style_Alignment::HORIZONTAL_CENTER,
        'vertical'=>
PHPExcel_Style_Alignment::VERTICAL_CENTER
    ),
    'borders' => array(
        'allborders' => array(
            'style'=>PHPExcel_Style_Border::BORDER_THIN
        )
    )
);

//Перейдём на вторую строку инвойса
$row_start = $row_start + 1;
$active_sheet->setCellValue('A'.$row_start,'Код товара');
$active_sheet->setCellValue('B'.$row_start,'Наименование продукта');
$active_sheet->setCellValue('C'.$row_start,'Ед. изм. ');
$active_sheet->setCellValue('D'.$row_start,'в упаковке');
$active_sheet->setCellValue('E'.$row_start,'Цена за единицу');
$active_sheet->setCellValue('F'.$row_start,'Заказ, в ед. изм. ');
$active_sheet->setCellValue('G'.$row_start,'Заказ, в упаковке');
$active_sheet->setCellValue('H'.$row_start,'Сумма заказа');
$active_sheet->setCellValue('J'.$row_start,'Код товара');
$active_sheet->setCellValue('K'.$row_start,'Наименование продукта');
$active_sheet->setCellValue('L'.$row_start,'Ед. изм. ');

```

```

$active_sheet->setCellValue('M'.$row_start,'в упаковке');
$active_sheet->setCellValue('N'.$row_start,'Цена за единицу');
$active_sheet->setCellValue('O'.$row_start,'Заказ, в ед. изм. ');
$active_sheet->setCellValue('P'.$row_start,'Заказ, в упаковке');
$active_sheet->setCellValue('Q'.$row_start,'Сумма заказа');
$active_sheet->getStyle('A'.$row_start.':H'.$row_start)-
>applyFromArray($inv_num);
    $active_sheet->getStyle('J'.$row_start.':Q'.$row_start)-
>applyFromArray($inv_num);

    $active_sheet->getStyle('A'.$row_start.':H'.$row_start)->getAlignment()-
>setWrapText(true);//Перенос по словам
    $active_sheet->getStyle('J'.$row_start.':Q'.$row_start)->getAlignment()-
>setWrapText(true);//Перенос по словам

//Перейдём на третью строку инвойса, список продуктов
$row_start = $row_start + 1;
$bir = $row_start;
$zakaz_query = mysql_query("SELECT * FROM j_zakaz WHERE
invoice_id = ".$invoices['id']);
$zakaz = mysql_fetch_array($zakaz_query);
do // Цикл для ввода инфу о продуктах
{
    $tovar_query = mysql_query("SELECT * FROM sp_tovari WHERE id =
".$zakaz['tovari_id']);
    $tovar = mysql_fetch_array($tovar_query);
    $edinis_query = mysql_query("SELECT * FROM sp_edinisi WHERE id =
".$tovar['Yedinisi_id']);
    $edinis = mysql_fetch_array($edinis_query);
    $ts = $invoices['tip_seni_id'];

```

```

    $pereosenka = $invoices['pereosenka_id'];
    $narx_query = mysql_query("SELECT narx FROM j_detal_pereosenka
    WHERE j_pereosenka_id = ".$pereosenka." AND tovar_id = ".$stovar['id']." AND
    tip_seni = ".$sts);
    $narx = mysql_fetch_array($narx_query);

    $active_sheet->setCellValue('A'.$row_start,$zakaz['tovari_id']);
    $active_sheet->setCellValue('B'.$row_start,$stovar['naimenovanie']);
    $active_sheet->setCellValue('C'.$row_start,$dinis['naimenovanie']);
    $active_sheet->setCellValue('D'.$row_start,$stovar['Kol_v_korob']);
    $active_sheet->setCellValue('E'.$row_start,$narx['narx']);
    $active_sheet->getStyle('E'.$row_start)->getNumberFormat()-
    >setFormatCode('#,##0');
    $active_sheet->setCellValue('F'.$row_start,$zakaz['kolichestvo']);
    $active_sheet-
    >setCellValue('G'.$row_start,"=".$zakaz['kolichestvo']. "/" . $stovar['Kol_v_korob']);
    $active_sheet-
    >setCellValue('H'.$row_start,"=".$zakaz['kolichestvo']. "*" . $narx['narx']);
    $active_sheet->getStyle('H'.$row_start)->getNumberFormat()-
    >setFormatCode('#,##0');
    $active_sheet->setCellValue('J'.$row_start,$zakaz['tovari_id']);
    $active_sheet->setCellValue('K'.$row_start,$stovar['naimenovanie']);
    $active_sheet->setCellValue('L'.$row_start,$dinis['naimenovanie']);
    $active_sheet->setCellValue('M'.$row_start,$stovar['Kol_v_korob']);
    $active_sheet->setCellValue('N'.$row_start,$narx['narx']);
    $active_sheet->getStyle('N'.$row_start)->getNumberFormat()-
    >setFormatCode('#,##0');
    $active_sheet->setCellValue('O'.$row_start,$zakaz['kolichestvo']);
    $active_sheet-
    >setCellValue('P'.$row_start,"=".$zakaz['kolichestvo']. "/" . $stovar['Kol_v_korob']);

```

```

$active_sheet-
>setCellValue('Q'.$row_start,"=".$zakaz['kolichestvo']."*".$narx['narx']);
$active_sheet->getStyle('Q'.$row_start)->getNumberFormat()-
>setFormatCode('#,##0');
$massiv[$zakaz['tovari_id']]['id'] = $tovar['id'];
$massiv[$zakaz['tovari_id']]['edinis'] = $edinis['naimenovanie'];
$massiv[$zakaz['tovari_id']]['naimenovanie'] = $tovar['naimenovanie'];
$massiv[$zakaz['tovari_id']]['kol_v_korob'] = $tovar['Kol_v_korob'];
$massiv[$zakaz['tovari_id']]['sena'] = $narx['narx'];
$massiv[$zakaz['tovari_id']]['kolichestvo'] =
$massiv[$zakaz['tovari_id']]['kolichestvo'] + $zakaz['kolichestvo'];
$massiv[$zakaz['tovari_id']]['korobka'] =
$massiv[$zakaz['tovari_id']]['korobka'] +
$zakaz['kolichestvo']/$tovar['Kol_v_korob'];
$massiv[$zakaz['tovari_id']]['summa'] =
$massiv[$zakaz['tovari_id']]['summa'] + $zakaz['kolichestvo']*$narx['narx'];

```

Xulosa

II bobda onlayn buyurtma qabul qilish tizimini tashkil etish jarayoni ko'rsatib berildi. Bunda,

- onlayn buyurtma qabul qilish tizimi uchun MySQLda ma'lumotlar omborini tayyorlanish jarayoni;
- tizimning server qismidagi kerakli kutubxonalar asosiy sahifalarning ochib berilishi;
- tizimning mijoz qismidagi zarur sahifalar va ularning tavsiflari
- hisobotlarning tayyorlanish bosqichlari keltirib o'tildi.

Xotima

Ushbu bitiruv malakaviy ishimiz orqali qo'yilgan vazifalar to'liq bajarildi va maqsadga erishildi.

1. Zamonaviy web texnologiyalarning turlari, ularning ma'lumotlar ombori bilan bog'lanishi kabi nazariy bilimlar to'plandi, o'rganildi va tizimlashtirildi.

2. JQuery va Ajax texnologiyalari o'rganib chiqildi, ularning ish tamoyillari va qo'llanilish sohalari aniqlandi.

3. Onlayn buyurtma qabul qilish tizimi uchun MySQLda ma'lumotlar omborini tayyorlanish jarayoni taqdim etildi.

4. Tizimning server qismidagi kerakli kutubxonalar asosiy sahifalarning ochib berilishi ko'rsatildi.

5. Tizimning mijoz qismidagi zarur sahifalar va ularning tavsiflari keng ma'noda ochib berildi.

6. Hisobotlarning tayyorlanish bosqichlari keltirib o'tildi.

Foydalanilgan adabiyotlar

1. Karimov I.A. «Yuksakma'naviyatyengilmaskuch».
2. Каримов И.А. Баркамолавлодорзуси,- Ташкент «Шарк», 1999.
3. Ilm-fanyutuqlari – taraqqiyotningmuhimomili :
[O‘zbekistonRespublikasiPrezidentiShavkatMirziyoyevYangiyilarafasida –
2016 yil 30 dekabrkunimamlakatimizningyetakchiilm-
fannamoyondalaribilanuchrashdi] // Xalqso‘zi. – 2016. – 31 dek. – B.1. 2017
4. M.AripovS.DottayevM.Fayziyeva “webtexnologiyalar” Toshkent 2013
5. SayfiyevJ.F. “SQLtiligakirish” uslubiyqo’llanma.Buxoro 2005 yil.
6. Zakirova T.A., Ibragimov E.U., Masharipov A.K., Musaeva M.A.
“Webdasturlash” . O’quvqo’llanma – Toshkent 2010yil.
7. Молиново Э. SQL. Сборник рецептов. – Пер. с англ. – СПб: Символ-
Плюс, 2009. – 672 с., ил.
8. SH. P. Boboqulov, I. I. BakayevASPNETMVC 4 o’quvqo’llanmaBUXORO
– 2015
9. Дятлов В. А. Дистанционное профессиональное обучение – М. 1997.
- 10.<http://www.ziyonet.uz>
- 11.<http://www.uz.ref.uz>
- 12.<http://www.elkutubhona.narod.uz>
- 13.<http://uchebnikionline.ru>
- 14.<http://www.uz.ref.uz>
- 15.<http://www.ref.uz>