

**Коммерциялық емес  
акционерлік қоғамы**



**АЛМАТЫ ЭНЕРГЕТИКА  
ЖӘНЕ БАЙЛАНЫС  
УНИВЕРСИТЕТИ**

«Ақпараттық жүйелер»  
кафедрасы

## **WEB-БАҒДАРЛАМАЛАУ**

5B060200 – Информатика мамандығының студенттері үшін  
курстық жұмыстарды орындау бойынша әдістемелік нұсқаулықтар

Алматы 2016

ҚҰРАСТЫРУШЫЛАР: Альмуратова К.Б., Тұрысбек Ж., Иманбекова У.Н. Web-бағдарламалау. 5B060200 – Информатика мамандығының студенттері үшін курстық жұмыстарды орындау бойынша әдістемелік нұсқаулықтар. – Алматы: АЭЖБУ, 2015. – 23 б.

Әдістемелік нұсқаулықтар курстық жұмыстың негізгі есептерінен, оларды орындауға қойылатын талаптардан, курстық жұмыстарды орындауға арналған ұсыныстардан және тапсырма нұсқаларынан тұрады.

Әдістемелік нұсқаулықтар 5B060200-Информатика мамандығының студенттеріне «Web-бағдарламалау» пәні бойынша курстық жұмыстарды орындауға арналған. Берілген әдістемелік нұсқаулықта РНР тілінде web-жүйелерді құру технологиясы қарастырылған.

Әдеб.көрсеткіші – 9 атау.

Пікір беруші: техн.ғыл.канд., доц. К.С. Чезимбаева

«Алматы энергетика және байланыс университетінің» коммерциялық емес акционерлік қоғамының 2015 ж. жоспары бойынша басылады.

© «Алматы энергетика және байланыс университеті» КЕАҚ, 2016 ж.

## Кіріспе

Web-бағдарламалау – бағдарламалаудың бір бөлімі, web - қосымшаларды (дүниежүзілік тораптың динамикалық сайттарының функционалдылығын қамтамасыз ететін бағдарламалар) құрастыруға бағытталған.

Web-бағдарламалау тілдері – бұл негізінен Web-технологиялармен жұмыс істеуге арналған тіл. Web-бағдарламалау тілдері бір-біріне жалғасатын екі топқа бөлуге болады: клиенттік және серверлік.

Аталуына сәйкес, клиенттік тілдеріндегі бағдарламалар клиент жағында өңделеді, әдетте ол браузерде орындалады. Бұл клиенттік тілдердің басты мәселесін құрайды – бағдарламаның (скрипттің) орындалу нәтижесі пайдаланушы браузеріне байланысты. Яғни, егер пайдаланушы клиенттік бағдарламаларды орындауға тиым салса, онда бағдарламалаушы қанша қаласа да олар орындалмайды. Сонымен қатар, әртүрлі браузерде немесе бір браузердің түрлі нұсқаларында бағдарлама әртүрлі орындалуы мүмкін. Екінші жағынан, егер бағдарламашы серверлік бағдарламаларға үміт артса, онда ол өз жұмысын жеңілдетумен қатар бағдарламала арқылы сервердегі жүктемені азайтуы мүмкін, өйткені клиент жағында, ешқашан бетті қайта қосуды (генерациялауды) талап етпейді.

Пайдаланушы кез келген бетке сұраныс жасаған кезде (сілтемеге өткенде немесе өз браузеріңізде мекен-жай жолағына мекен-жайды енгізген кезде), келтірілген бет бірінші серверде өңделеді, яғни барлық бағдарламаларды іске қосу бетке байланысты, содан кейін файл ретінде желіге қайтарылады. Бұл файлдың кеңейтілімдері: HTML, PHP, ASP, ASPX, Perl, SSI, XML, DHTML, XHTML.

Бағдарламалардың жұмыс істеуі сайт орналасқан серверге және сол немесе басқа тілдің қайсыбір қолдайтын версияларына толығымен тәуелді. Бағдарламалаудың серверлік тілдеріне келесілер жатады: PHP, Perl, Python, Ruby, кез келген .NET бағдарламалау тілі, (ASP.NET технологиясы) Java, Groovy.

Серверлік тілдердің жұмыс істеуінің маңызды жақтары деректер қорын басқару жүйесімен (немесе ДҚБЖ) - кез келген мезетте шақырыла алатын ақпараттың ретімен орналасқан сервермен тікелей өзара әрекетімен ұйымдастырылуы мүмкіндігі болып табылады.

Курстық жұмыстың негізгі мақсаты нақты ДҚБЖ (My SQL, HTML, JavaScript, PHP және т.б.) негізінде анықталған пәндік аймақ үшін ДҚ қосымшасы мен ДҚ қорғау тәсілдерін бағдарламалық түрде жүзеге асыратын деректер қорын құру бойынша тәжірибелік дағдыларды оқып-үйрену болып табылады.

Курстық жұмысты орындау үшін деректер қорын жобалаудың принциптері мен әдістерін, ДҚБЖ-нің тілдік құралдарын, деректер қорының

қосымшаларын жүзеге асыратын принциптерін және нақты ДҚБЖ ортасында жұмыс істеуді білуі керек.

## **1 Курстық жұмыстың негізгі есептері мен оларды орындауға қойылатын талаптар**

### **1.1 Курстық жұмыстың мазмұны**

«Web-бағдарламалау» пәні бойынша курстық жұмыстарды жеке тапсырмаға сәйкес, студент белгілі бір пәндік облыс үшін есепке алу деректері бойынша ақпараттық жүйені дамыту керек.

Клиент-сервер архитектурасының 3-деңгейдегі дамыған жүйесі құралуы қажет.

Қазіргі заманғы кез келген браузер клиенттік қосымша ретінде пайдаланылуы тиіс.

HTTP- Apache сервері бағдарламалар сервері ретінде пайдаланылуы тиіс. ДҚБЖ туралы ақпаратты MySQL деректер базасында сақталуы тиіс,

Жүйенің қалыптасқан модульдік құрылымы болуы тиіс.

Ол келесі тапсырмаларды орындай алуы керек:

- мәліметтерді қосу;
- деректерді редакциялау;
- мәліметтерді жою;
- мәліметтерді беру.

Жүйенің қалыптасқан деректердің редакциялау және жазбаны жеңілдету керек.

### **1.2 Курстық жобаны рәсімдеу және қорғау**

Курстық жұмыстың тақырыбы, үлгі бойынша құрылған студент құжатында көрсетіледі: «Есепке алудың Web-жүйесін дамыту [Пәндік саладағы]».

Мысалы: «Театр қойылымдарын есепке алудың web-жүйесін дамыту».

Курстық жұмыстың түсіндірме жазба мәтінінің көлемі 20-50 бет шамасында болуы тиіс. Кафедрада қабылдаған нормативтік бақылау ережелеріне сәйкес әзірленген.

«Есептің қойылымы» бөлімі нөмірлеу бөлімінде басталады.

Файлдардың бастапқы коды және деректер қорының sql-скрипт файлы қосымшада келтіріледі. Курстық жұмыстың негізгі және графикалық бөлігі мемлекеттік стандарттар талаптарына сәйкес жасалуы керек.

Тұтас мәтін үшін парақтың төрт жағынан өрістер (сол жағы - 25 мм, оң жағы - 18 мм, жоғарғы жағы - 20 мм, төменгі жағы - 25 мм), бір жарым жол аралық аралықпен Times New Roman №12 шрифтiсi (Times New Roman №14 бір жол аралық аралықпен) пайдаланылады.

Негізгі бөлімнің мәтіні тарауларға бөлінеді, ал тараулар ішкі тарауларға бөлінеді. Тараулар барлық жұмыс аралығында араб цифрларымен белгіленген реттік нөмірі болуы керек, тарау нөмірінен кейін нүкте қойылмайды. Тараудың атауы азат жолдан үлкен әріптермен жазылады, тарау атауынан кейін нүкте қойылмайды. Әрбір тарау жаңа беттен басталған жөн.

Ішкі тараулар әрбір тарау аралығында нөмірленуі керек. Ішкі тараудың нөмірі нүктемен ажыратылған тарау нөмірі мен ішкі тарау нөмірінен тұрады, ішкі тараудың нөмірінен кейін нүкте қойылмайды. Ішкі тараудың атауы тақырып сияқты әріптермен жазылады, бірінші әрпі – үлкен. Ішкі тараудың атауынан кейін нүкте қойылмайды.

Түсіндірме жазбаның мазмұны:

- титулдық беті (міндетті түрде тақырып атауымен бірге стандарт бойынша рәсімделген);
- түсіндірме жазба;
- курстық жобаға тапсырма;
- мазмұны;
- есептің қойылымы (қойылған мәселенің шарты, өз нұсқасына қатысты жүйенің функциясы);
- деректер қорының құрылымын сипаттау (кестелер сипаттамасы және олардың арасындағы қарым-қатынастар);
- жоба құрылымының сипаттамасы (файлдар, каталогтары мен функциялары сипаттамалары);
- қолданушыға нұсқаулық (әзірленген жүйенің толық сипаттамасы, пайдалану, орнату және т. б.);
- дерек көздері мен әдебиеттер тізімі;
- қосымшалар.

Барлық беттер (парақтар) титулдық беттен бастап, сонымен қатар қосымшаны қосқанда ретпен нөмірленуі керек. Бет нөмірі нүктесіз жоғарғы оң жақ бұрышта (титулды бетте бет нөмірі қойылмайды) жазылады. Мазмұнына кіретін әзірленген жүйенің толық сипаттамасы, пайдалану, атаулар (бірінші әріпінен басқасы) кіші әріптермен жазылады. «МАЗМҰНЫ», «КІРІСПЕ», «ҚОРЫТЫНДЫ», «ӘДЕБИЕТТЕР ТІЗІМІ» деген сөздер тақырып ретінде үлкен әріптермен жазылады және оларға нөмір қойылмайды. Әрбір қосымша жаңа беттен басталады. Егер олар бірден көп болса, онда қосымшаларды қазақ алфавитінің бас әріптерімен белгілейді. «Қосымша» сөзінің алдына оның тізбегін көрсететін әріп болады. «Қосымша» сөзі біріншісі үлкен одан кейін кіші әріптермен жазылады және жазба беттің жоғарғы ортасында орналасады. Егер қосымша бірнеше беттен тұрса, онда әрбір келесі беттеріне «А қосымшасының жалғасы» деп жазу керек.

Курстық жұмыс құжатта ретпен нөмірленетін (егер олар мәтінде бірден көп болса) кестелер мен суреттерден (сызбалар, сұлбалар, диаграммалар және т.б.) тұрады.

«Кесте» сөзі сол жақ бұрышта азат жолдан үлкен әріппен жазылады, одан кейін дефис қойылады да, әрі қарай кестенің атауы үлкен әріппен бастап жазылады. «Сурет» сөзі сурет астына азат жолдан үлкен әріппен жазылады, одан кейін дефис қойылады да, әрі қарай суреттің атауы үлкен әріппен бастап жазылады. Кестенің (суреттің) атауынан кейін нүкте қойылмайды.

Кестелердің (суреттердің) нөмірленуі тарау аралығында болуы мүмкін. Мұндай жағдайда кестенің (суреттің) нөмірі нүктемен ажыратылған тарау нөмірі мен кестенің (суреттің) реттік нөмірінен тұрады. Барлық кестелерге (суреттерге) құжат мәтінінде сілтемелер келтірілуі керек.

Жұмыстар қорғауға жоғарыда келтірілген талаптарға сәйкес оқытушының тексеруінен кейін жіберіледі. Қорғауда студент жұмыста пайдаланылған негізгі анықтамалар мен түсініктерді білуі керек, қабылданған шешімді таңдағанын түсіндіре алуы керек, сонымен қатар, барлық жасалған мәселелерді білуі керек.

Курстық жұмысты қорғау компьютерде жасалған қолданба жұмысының демонстрациясымен келтірілуі керек.

## **2 Курстық жұмыстарды орындауға ұсыныстар**

### **2.1 Ақпараттық модельді құру және деректер қорын жобалау**

Жеке тапсырма бойынша топтың тізімінің нөмірі немесе жұмыстың бағыты бойынша таңдалған пәндік облысқа талдау жасалынады. Соңғы қолданушыны қызықтыра алатын нысандар анықталады.

Жеке тапсырманы оқытушы студентпен ақылдаса келе өзі тағайындайды. Деректер қорының логикалық моделін оқытушы белгілі бір нысанға келтіреді.

Пәндік облыстың үлгі ретіндегі тізімінің негізінде жасалған тапсырманың жеке нұсқасы:

- телефондық желінің абоненттерін тіркеу (Аты-жөні, мекен-жайы, телефон нөмері, тариф, ақысы);
- студенттік конференцияның қатысушыларын тіркеу (Аты-жөні, факультет, топ, курс, ғылыми секцияның тақырыбы, жетекшісі, баяндаманың тақырыбы);
- «Ауа-райының күнтізбесі» (мерзімі, ауаның температурасы, желдің бағыты, жауын-шашындар);
- қалалық кеңестің депутаттарын сайлауалды компаниясының қатысушыларын тіркеу (Аты-жөні, жасы, саясаттық партия, лауазымы, отбасылық жағдайы);
- үй кітапханасындағы кітаптарды тіркеу (атауы, авторы, әдеби жанры, баспа, бағасы);
- ақылы тұрақтағы автокөліктерді тіркеу (автокөліктің нөмірі, иесінің аты-жөні, тұрақта тұрған уақыты, құны).

- баскетбол турнирінің нәтижесін, турнирге қатысатын командалардың мәліметтерін тіркеу (атауы, қала, бапкердің тегі, өткізілген ойындардың саны, ұтыстар саны);
- оқу сыныбындағы компьютерлік жабдықтарды, компьютер туралы мәліметтерді тіркеу (есептік нөмірі, модельдің коды, процессордың типі, дисктің өлшемі, жадының көлемі);
- бір топтың студенттерінің бір семестрдегі емтихан тапсыруының нәтижесін тіркеу (студенттің аты-жөні, сынақ кітапшасының нөмірі, информатика пәнінен бағасы, ағылшын тілінен бағасы, математика пәнінен бағасы, физика пәнінен бағасы);
- Кафедраның қызметкерлерінің жаздық демалысқа шығу кестесін құру, қызметкерлер туралы мәліметтер (Аты-жөні, лауазымы, табельдік нөмірі, жаздық демалыстың ұзақтылығы);
- «Оқытушының электронды журналы», студенттердің сабаққа қатысуы мен үлгерімін тіркеу (Аты-жөні, пән, сабаққа қатысуы, бағасы);
- Ухта бекеті арқылы жүретін поездердың кестесі. Поездер туралы мәліметтер (нөмірі, жөнелту пункті, бекет бағыты, жолдағы уақыты, келу уақыты);
- кітапхананың абоненттерін тіркеу, абоненттер туралы мәліметтер (Аты-жөні, жеке нөмірі, кітаптың аты, авторы, кітаптың инвентарлы нөмірі, тапсыру уақыты);
- ауру парағын тіркеу, ауру парағы туралы мәліметтер (Аты-жөні, диагнозы, ауру парағының нөмірі, дәрігердің аты-жөні, берілу мерзімі, аяқталу мерзімі, күндер саны);
- көшіру аппаратының жұмысын тіркеу (мерзімі, аты-жөні, бөлім, беттер саны, көшірмелер саны, бір көшірменің бағасы, әр жүз беттің пайыздық қосымша үстемесі, барлығы);
- Қоймадағы аяқ-киімдер тобын тіркеу (артикул (аяқ-киім типі – ерлер, әйелдер, балалар), атауы, 1 жұптың бағасы, сатып алушы, жіберілгені, жұптар көлемінің жеңілдік пайыздары, жұптар бағасы);
- тұрғын ауданының төлемақысы (тұрғын, мекен-жайы, тұрғын бөлменің ауданы, бағасы (бір шаршы аудан үшін), артық тұрғын ауданы үшін қосымша ақы төлеу, тұрғындар саны, барлығы);
- спорт клубындағы клубтық жарна (спортсменнің аты-жөні, спорт түрі, жарыс, орын, клубтық жарна, клубтық жарнаға жеңілдіктер (иемденген орнына байланысты есептелінеді), жарна сомасы);
- сұлулық салонындағы клиенттерді тіркеу (клиент, шебер, шебердің біліктілігі, қызметі, мерзімі, тұрақты клиенттерге жеңілдіктер, шебердің біліктілігі үшін қосымша жалақы, қызмет құны);
- театр билеттерін сатуды тіркеу (аты, жанр, режиссер, билет құны, сатылған билеттер, топтық сұраныстарғы жеңілдіктер, мерзімі, қалған билеттер, спектакльге берілгені);

- кадрлар бөліміндегі қызметкерлердің орындалған сағат саны мен есептелген еңбекақылары (қызметкер, бөлім, орындалған жыл саны, орындалған сағат, тариф, қосымша ақы төлеу, есептелгені);
- клиенттерге азық-түліктерді жеткізу (клиент, азық-түлік, саны, жеткізу ұзақтылығы, жеткізу ұзақтылығы үшін қосымша ақша төлеу, тұрақты клиенттерге жеңілділіктер, сомасы);
- видео/аудио дискілерді жалға алу (тапсырыс беруші, атауы, типі, көшіру форматы, берілген уақыты, қайтару уақыты, күндер саны, тәулік бойынша жалға алу уақыты, кідіріс үшін қосымша ақы төлеу, барлығы);
- құрылыс-бөлімдік ұйымдастыру жұмыстарын тіркеу (сұраныс беруші, қызмет түрі, фирманың орындау уақыты, сұраныс берушінің тапсырмасын жедел орындау үшін қосымша ақы төлеу, сұраныс берушінің материалының құны, жалпы құны);
- қалаларалық телефондық сөйлесулерді тіркеу (абонент, байланысу қаласы, мерзімі, коэффициент зонасы, минуттар саны, түнгі уақыттағы жеңілдіктер, құны);
- Интернет клубтың жұмысын тіркеу (қолданушы, компьютер нөмірі, сеанстың басталуы, қолдану уақыты, тариф (компьютердің жылдам жұмыс істеуі), кеңес берушінің қызметінің құны, түнгі уақыттағы жеңілдіктер, құны);
- Интернет-магазиннің клиенттерін тіркеу (фамилия, телефон, тауар коды, сатып алу мерзімі, тауар бірлігінің құны, саны, жалпы сомасы);
- автокөліктерді сату салонының клиенттерін тіркеу (аты-жөні, паспорт №, машина маркасы, шығарылу жылы, сомасы, жеңілдіктер, жалпы сомасы);
- құрылыс материалдарын жалпылама сатуды тіркеу (тапсырыс №, сату мерзімі, тауар түрі, бірлік құны, тауар саны, сома, жеңілдіктер, жалпы сомасы);
- ЖОО талапкерлер, талапкерлер туралы мәліметтер (аты-жөні, мекен-жайы, паспорт нөмірі, мамандық, орташа балл, аралық балл, есептелу).

## **2.2 Деректер қорымен жұмыс істеуге арналған нұсқауларды құру**

ДҚ программист пен администраторды сүйемелдейтін пайдаланушының нұсқаулары құрастырылады.

Пайдаланушының нұсқаулары интерфейс формаларының түрлерінің иллюстрацияларымен бірге соңғы пайдаланушы үшін қолданбамен жұмыс істеуге арналған нұсқаулар болуы керек.

Жасайтын программистке арналған нұсқаулар қажетті техникалық және бағдарламалық құралдардың сипатталуы, жүйе файлдарының сипаттамасы, оларды сақтау құрылымы болуы керек.

ДҚ администраторына арналған нұсқаулар жүйеге кіруге арналған парольді сақтау мен өзгерту тәсілдері туралы, ДҚ әкімшілік ету әдістері мен құралдары туралы нұсқаулар болуы керек.



## **2.3 Жұмысты тапсыру мен қорғау уақыты**

Жұмыс екі этапта қорғалады

Бірінші этапта студент жобаның алғашқы файлы мен деректер қорының sql-скрипт файлының жұмыс өнімділігін тексерту үшін оқытушыға көрсетуі тиіс.

Жобаны қорғағаннан соң студент құжаттаманы (түсінік хат) көрсетуі тиіс. Екінші этапта қорғау процесінде курстық жұмысты орындауда қолданылған технологияға байланысты сұрақтарға студент жауап беруі тиіс.

### **Қорытынды**

Деректер қорының технологияларын оқытумен және web-бағдарламалау пәндерімен байланысты пәндер бойынша курстық жұмысты орындау нақты ДҚБЖ негізінде анықталған пәндік аймақ үшін қорғау мен қауіпсіздік әдістерін пайдалана отырып, ДҚ құру және ДҚ қолданбаларын құру бойынша практикалық дағдыларды қамтамасыз етеді.

Курстық жұмысты орындау үрдісінде студенттер деректер қорын жобалау әдістері мен принциптерін білуге, деректер қорының қолданбаларын жобалау принциптері мен әдістерін, ДҚБЖ тілдік құралдарын, нақты ДҚБЖ ортасында жұмыс істеуге қажетті практикалық білім алуға иеленеді.

Курстық жұмысты орындау үрдісінде алынған білімдер мен дағдылар келешекте деректер қорының және web-бағдарламалау технологияларының негізіндегі түрлі жүйелерді құрастырумен байланысты пәндерді оқу, сонымен қатар дипломдық жобалауды орындау кезінде пайдаланылуы мүмкін.

## А қосымшасы

### Жобаның орындалу үлгісі

Студенттер туралы негізгі ақпаратты есепке алуды ақпараттандыруға арналған «Студенттер туралы ақпараттар» ақпараттық жүйесі.

Жүйе модульдік архитектура бойынша құрылған.

Топтағы студенттер туралы ақпаратты енгізу мен өңдеу тізім өрісі арқылы іске асырылады.

Жүйе келесі файлдардан құралған:

- Index.php – жобаның негізгі файлы;
- Function.php – қажетті функциялар тізімінен құралған файл;
- Grupa.php – топ туралы ақпаратты шығару;
- Grupaform.php – «Топ» кестесімен жұмыс жасауға арналған форманы сипаттау;
- Grupapost.php – «Топ» кестесіндегі ақпаратты өңдеу мен жою үшін арналған скрипттен құралған;
- Student.php – студенттер туралы ақпаратты шығарады;
- Studentform.php – «Студент» кестесімен жұмыс жасауға арналған форманы сипаттау;
- Studentpost.php – «Студент» кестесіндегі ақпаратты өңдеу мен жою үшін арналған скрипттен құралған.

«Index.php» файл мазмұны

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title> Студенттер туралы ақпарат </title>
</head>
<body>
<h1> Студенттер туралы ақпарат </h1>
<p><a href='index.php?p=1'> Студенттер </a> &nbsp;&nbsp;&nbsp;<a
href='index.php?p=2'>Группы</a><p>
```

```
<?php
```

```
$p=$_REQUEST['p'];
```

```
if (($p==0)||($p==1))
```

```
include_once 'student.php';
```

```
if ($p==11)
```

```
include_once 'studentform.php';
```

```

if ($p==12)
    include_once 'studentpost.php';

if ($p==2)
    include_once 'grupa.php';

if ($p==21)
    include_once 'grupaform.php';

if ($p==22)
    include_once 'grupapost.php';

// echo "<p>$fiostudent<p>";
?>

<p>&copy; Төлегенов Сәкен Маратұлы <p>
</body></html>

```

«Function.php» файл мазмұны  
<?php  
error\_reporting( E\_ERROR );

```

function dbconnect()
{
    $db = new mysqli('localhost', 'mysql', 'mysql', 'student');

    if (mysqli_connect_errno())
    {
        $error=mysqli_connect_error();
        $errorno=mysqli_connect_errno();
        exit ("<p> қосылу мүмкін емес: $errorno - $error<p>");
    }
    else
    {
        return $db;
    }
}

function student()
{
    $query =
        "SELECT  grupa.grupa,  student.idstudent,  student.idgrupa,  student.
fiostudent, student.nshifr

```

```

FROM student INNER JOIN grupa ON student.idgrupa = grupa.idgrupa
ORDER BY grupa.grupa, student.fiostudent";
$handle=dbconnect();
$result = $handle->query($query);
return $result;
}

```

```

function studentid($id)
{
    $query =
    "SELECT idstudent, fiostudent, idgrupa, nshifr
    FROM student
    WHERE idstudent=$id";

    $handle=dbconnect();
    $result = $handle->query($query);
    return $result;
}

```

```

function studentins($fiostudent, $idgrupa, $nshifr)
{
    $query =
    "INSERT INTO student (fiostudent, idgrupa, nshifr)
    VALUES ('$fiostudent', $idgrupa, '$nshifr')";

    $handle=dbconnect();
    $result = $handle->query($query);
    return $result;
}

```

```

function studentupd($idstudent, $fiostudent, $idgrupa, $nshifr)
{
    $query =
    "UPDATE student SET fiostudent='$fiostudent', idgrupa=$idgrupa,
nshifr='$nshifr'
    WHERE idstudent=$idstudent";

    $handle=dbconnect();
    $result = $handle->query($query);
    return $result;
}

```

```

function studentdel($idstudent)
{

```

```

    $query =
    "DELETE FROM student WHERE idstudent=$idstudent";
    $handle=dbconnect();
    $result = $handle->query($query);
    return $result;
}

function grupa()
{
    $query =
    "SELECT idgrupa, grupa
    FROM grupa
    ORDER BY grupa ASC";

    $handle=dbconnect();
    $result = $handle->query($query);
    return $result;
}

function grupaid($id)
{
    $query =
    "SELECT idgrupa, grupa
    FROM grupa
    WHERE idgrupa=$id";

    $handle=dbconnect();
    $result = $handle->query($query);
    return $result;
}

function grupains($grupa)
{
    $query =
    "INSERT INTO grupa (grupa) VALUES ('$grupa')";
    $handle=dbconnect();
    $result = $handle->query($query);
}

function grupaupd($idgrupa, $grupa)
{
    $query =
    "UPDATE grupa SET grupa='$grupa' WHERE idgrupa=$idgrupa";

```

```

    $handle=dbconnect();
    $result = $handle->query($query);
}

function grupadel($idgrupa)
{
    $query =
    "DELETE FROM grupa WHERE idgrupa=$idgrupa";

    $handle=dbconnect();
    $result = $handle->query($query);
}

```

?>

«Grupa.php» файл мазмұны

```

<?php
include_once 'function.php';

$result=grupa();
$numresult= $result->num_rows;

echo
"<table border=1>
<th>Группы</th>
<td colspan=2><a href='index.php?p=21&m=1'>Қосы</a></td>";

for ($i=0;$i<$numresult;$i++)
{
    $row=$result->fetch_assoc();
    $idgrupa=$row['idgrupa'];
    $grupa=$row['grupa'];
    $i1=$i+1;

    echo
    "<tr><td>$grupa</td>
<td><a
href='index.php?p=21&m=2&id=$idgrupa'>Өзгертю</a></td>
<td><a
href='index.php?p=21&m=3&id=$idgrupa'>Жою </a></td>";

}
echo

```

```
"</table>
<p> Жазбалар саны - $numresult<p>";
?>
```

«Grupaform.php» файл мазмұны

```
<?php
include_once 'function.php';

$m=$_REQUEST['m'];

if ($m==1)
{
echo "<h2> Топты қосу </h2>

<form action='index.php' method=post>
<input type='hidden' name='m' value=1>
<input type='hidden' name='p' value=22>
<table border=0>
<tr>
<th>Тобы</th>
<td><input type='text' name='grupa'></td>
</tr>
<tr>
<td colspan=2 align=center><input type='submit' value='
Сақмай'></td>
</tr>
</table>
</form>";
}
if ($m==2)
{

$idgrupa=$_REQUEST['id'];

$result=grupaid($idgrupa);

$row=$result->fetch_assoc();

$grupa=$row['grupa'];

echo "<h2> Топты өзгерту </h2>

<form action='index.php' method=post>
```

```

<input type='hidden' name='m' value=2>
<input type='hidden' name='id' value=$idgrupa>
<input type='hidden' name='p' value=22>
<table border=0>
<tr>
  <th>Тон </th>
  <td><input type='text' name='grupa' value='$grupa'></td>
</tr>
<tr>
  <td colspan=2 align=center><input type='submit' value='
Сақмай'></td>
</tr>
</table>
</form>";
}
if ($m==3)
{
  $idgrupa=$_REQUEST['id'];

echo "<h2>Тонты өшіру </h2>
<form action='index.php' method=post>
<input type='hidden' name='m' value=3>
<input type='hidden' name='id' value=$idgrupa>
<input type='hidden' name='p' value=22>
<table border=0>
<tr>
  <th> Тон туралы деректерді жою?</th>
</tr>
<tr>
  <td align=center><input type='submit' value='Иә'></td>
</tr>
</table>
</form>";
}
?>

```

«Grupapost.php» файл мазмұны

```

<?php
include_once 'function.php';

$m=$_REQUEST['m'];

if ($m==1)

```



```

{
    $grupa=$_REQUEST['grupa'];

    if (empty($grupa))
    {
        echo "<h3> Барлық деректер енгізілген жоқ</h3>";
    }
    else
    {
        grupains($grupa);
        echo "<script>location.href='index.php?p=2'</script>";
    }
}

if ($m==2)
{
    $idgrupa=$_REQUEST['id'];
    $grupa=$_REQUEST['grupa'];

    if (empty($grupa))
    {
        echo "<h3> Барлық деректер енгізілген жоқ </h3>";
    }
    else
    {
        grupaupd($idgrupa, $grupa);
        echo "<script>location.href='index.php?p=2'</script>";
    }
}

if ($m==3)
{
    $idgrupa=$_REQUEST['id'];
    grupadel($idgrupa);
    echo "<script>location.href='index.php?p=2'</script>";
}
?>

```

«Student.php» файл мазмұны

```

<?php
include_once 'function.php';

```

```

$result=student();
$numresult= $result->num_rows;
echo
"<table border=1>

<th>Топ</th>
<th>Студенттің аты-жөні </th>
<th>Сынақ кітапша № </th>
<td colspan=2><a href='index.php?p=11&m=1'>Қосу </a></td>";

for ($i=0;$i<$numresult;$i++)
{
    $row=$result->fetch_assoc();
    $idstudent=$row['idstudent'];
    $fiostudent=$row['fiostudent'];
    $idgrupa=$row['idgrupa'];
    $grupa=$row['grupa'];
    $nshifr=$row['nshifr'];
    $i1=$i+1;
echo
"<tr><td>$grupa</td>
    <td>$fiostudent</td>
    <td>$nshifr</td>
<td><a
    href='index.php?p=11 &id=$idstudent&m=2'>Өзгертуді </a></td>
<td><a
    href='index.php?p=11 &id=$idstudent&m=3'>жою</a></td>";
}

echo "</table>
<p> Жазбалар саны - $numresult";
?>

```

«Studentform.php» файл мазмұны

```

<?php
include_once 'function.php';

$m=$_REQUEST['m'];

if ($m==1)
{
    echo "<h2> Студент қосу </h2>

```

```

<form action='index.php' method=post>
<input type='hidden' name='m' value=1>
<input type='hidden' name='p' value=12>
<table border=0>
<tr>
  <th>Студенттің аты-жөні </th>
  <td><input type='text' name='fiostudent'></td>
</tr>
<tr>
  <th>Сынақ кітапша № </th>
  <td><input type='text' name='nshifr'></td>
</tr>
<tr>
  <th>Тон </th><td>

<select name='idgrupa'>
<option selected value='0'> Көрсетілмеген </option>";

$result=grupa();
$numresult= grupa()->num_rows;

for ($i=0;$i<$numresult;$i++)
{
  $row=$result->fetch_assoc();
  $idgrupa=$row['idgrupa'];
  $grupa=$row['grupa'];

echo
"<option value='$idgrupa'>$grupa</option>";
}

echo
"</select></td></tr>
<tr>
  <td colspan=2 align=center><input type='submit' value='Сақмай
'></td>
</tr>
</table>
</form>";
}

if ($m==2)
{
  $idstudent=$_REQUEST['id'];
  $result=studentid($idstudent);

```

```

$row=$result->fetch_assoc();
$fiostudent=$row['fiostudent'];
$nshifr=$row['nshifr'];
$idgrupasel=$row['idgrupa'];

echo "<h2> студентті өзгерту </h2>
<form action='index.php' method=post>
<input type='hidden' name='m' value=2>
<input type='hidden' name='id' value=$idstudent>
<input type='hidden' name='p' value=12>

<table border=0>
<tr>
<th> Студенттің аты-жөні </th>
<td><input type='text' name='fiostudent' value='$fiostudent'></td>
</tr>
<tr>
<th> Сынақ кітапша № </th>
<td><input type='text' name='nshifr' value='$nshifr'></td>
</tr>
<th>Топ</th><td>

<select name='idgrupa'>
<option value='0'> Көрсетілмеген </option>";

$result=grupa();
$numresult= grupa()->num_rows;

for ($i=0;$i<$numresult;$i++)
{
    $row=$result->fetch_assoc();
    $idgrupa=$row['idgrupa'];
    $grupa=$row['grupa'];

    if ($idgrupasel==$idgrupa)

    echo "<option value=' $idgrupa' selected>$grupa</option>";
    else

    echo "<option value=' $idgrupa'>$grupa</option>";
}

echo
"</select></td></tr>

```

```

        <tr>
            <td colspan=2 align=center><input type='submit'
value='Сақмай'></td>
        </tr>
    </table>
</form>";
}

```

```

if ($m==3)
{
    $idstudent=$_REQUEST['id'];

```

```

echo "<h2> Студентті жою </h2>

```

```

<form action='index.php' method=post>
<input type='hidden' name='m' value=3>
<input type='hidden' name='id' value=$idstudent>
<input type='hidden' name='p' value=12>

```

```

<table border=0>
<tr>
    <th> Студент туралы деректерді жою?</th>
</tr>
<tr>
    <td align=center><input type='submit' value='Иә'></td>
</tr>
</table>
</form>";
}
?>

```

«Studentpost.php» файл мазмұны

```

<?php
include_once 'function.php';

$m=$_REQUEST['m'];
if ($m==1)
{
    $fiostudent=$_REQUEST['fiostudent'];
    $nshifr=$_REQUEST['nshifr'];
    $idgrupa=$_REQUEST['idgrupa'];

```

```

    if (empty($fiostudent)||empty($nshifr)||($idgrupa==0))
    {
        echo "<h3> Барлық деректер енгізілген жоқ</h3>";
    }

    else
    {
        studentins($fiostudent, $idgrupa, $nshifr);

        echo "<script>location.href='index.php?p=1'</script>";
    }
}

if ($m==2)
{
    $idstudent=$_REQUEST['id'];
    $fiostudent=$_REQUEST['fiostudent'];
    $nshifr=$_REQUEST['nshifr'];
    $idgrupa=$_REQUEST['idgrupa'];

    if (empty($fiostudent)||empty($nshifr)||($idgrupa==0))
    {
        echo "<h3>Барлық деректер енгізілген жоқ</h3><p>";
    }
    else
    {
        studentupd($idstudent,$fiostudent, $idgrupa, $nshifr);

        echo "<script>location.href='index.php?p=1'</script>";
    }
}

if ($m==3)
{
    $idstudent=$_REQUEST['id'];
    studentdel($idstudent);
    echo "<script>location.href='index.php?p=1'</script>";
}
?>

```

## Әдебиеттер тізімі

- 1 Веллинг, Люк. Разработка Web-приложений с помощью PHP и MySQL / Люк Веллинг, Лаура Томсон ; пер. с англ. – 3-е изд. – М.: Издательский дом «Вильямс», 2008. – 880 с.; ил.
- 2 Веллинг, Люк. Разработка Web-приложений с помощью PHP и MySQL /Люк Веллинг, Лаура Томсон; пер. с англ. – 4-е изд. – М. : Издательский дом «Вильямс», 2008. – 848 с.; ил.
- 3 Сыдықов Б.Д. ж/е т.б. Деректер қоры.- А.: «Нур-Принт», 2012.
- 4 Леонард Л. И др. Разработка приложений на основе Microsoft SQL Server 2008.-М., 2010.
- 5 Тернстрем Т. Microsoft SQL Server 2008 . Разработка баз данных.-М., 2010.
- 6 Макленнен Д. SQL Server 2008 Data mining интеллектуальный анализ данных.-СПб.: «БХВ-Петербург», 2009.
- 7 Вишневский А. SQL Server. Эффективная работа.- Спб.: «Питер», 2009.

## Мазмұны

Кіріспе .....	3
1 Курстық жұмыстың негізгі есептері мен оларды орындауға қойылатын талаптар .....	4
1.1 Курстық жұмыстың мазмұны.....	4
1.2 Курстық жобаны рәсімдеу және қорғау.....	4
2 Курстық жұмыстарды орындауға ұсыныстар.....	6
2.1 Ақпараттық модельді құру және деректер қорын жобалау.....	6
2.2 Деректер қорымен жұмыс істеуге арналған нұсқауларды құру.....	8
2.3 Жұмысты тапсыру мен қорғау уақыты.....	9
Қорытынды.....	9
А қосымшасы.....	10
Әдебиеттер тізімі.....	23

Камшат Бимуратовна Альмуратова

Жәнібек Тұрысбек

Улжан Нурхалиевна Иманбекова

## WEB-БАҒДАРЛАМАЛАУ

5B060200 – Информатика мамандығының студенттері үшін  
курстық жұмыстарды орындау бойынша әдістемелік нұсқаулықтар

Редакторы Ж.Н. Изтелеуова

Стандарттау бойынша маман Н.Қ. Молдабекова

Басуға \_\_\_ \_\_ \_\_\_ қол қойылды

Таралымы 40 дана.

Көлемі 1,4 есептік-баспа табақ

**Пішімі 60x84 1/16**

Баспаханалық қағаз №1

Тапсырыс \_\_\_\_\_ Бағасы 700 тг.

«Алматы энергетика және байланыс университеті»  
коммерциялық емес акционерлік қоғамының  
көшірмелі-көбейткіш бюросы  
050013, Алматы, Байтұрсынұлы көшесі, 126