29.09.2025

1. Тақырыбы: Контентті басқару ұғымы. Цифрлық дәуірдегі мазмұнды тиімді басқару

Контентті басқару — бұл цифрлық ақпаратты жасау, сақтау, өңдеу және таратуды жүйелі түрде ұйымдастыру үдерісі. Қазіргі веб-даму кезеңінде контент негізгі құндылыққа айналды, өйткені сайттар мен қосымшалардың сапасы көбіне мазмұнның өзектілігі мен қолжетімділігіне байланысты. Контентті басқару жүйелері (CMS) — веб-сайттағы мәтіндер, суреттер, бейнематериалдар және басқа файлдарды реттеуге арналған арнайы бағдарламалық құралдар. Мысалы, WordPress, Joomla немесе Drupal секілді CMS платформалары мазмұнды оңай жаңартып, оны көпшілікке қолайлы форматта ұсынуға мүмкіндік береді. Цифрлық дәуірде тиімді контентті басқару ұйымның беделін арттырып, аудиториямен тұрақты байланыс орнатуға және бизнес процестерін оңтайландыруға ықпал етеді.

1. Тақырыбы: Мазмұнын анықтау

Мазмұнын анықтау — веб-жоба немесе цифрлық платформа үшін қандай ақпарат пайдаланушыға ұсынылатынын айқындау үдерісі. Бұл кезең контент стратегиясының негізгі бөлігі болып саналады, себебі дұрыс таңдалған мазмұн сайттың мақсатына, аудиторияның қажеттілігіне және ұйымның миссиясына сәйкес болуы тиіс. Алдымен мақсатты аудитория талданады: олардың жас ерекшелігі, мүддесі, ақпарат іздеу әдеттері анықталады. Кейін контент түрлері — мәтін, бейне, сурет, инфографика немесе аудио — таңдалады. Мазмұнның тілі түсінікті, құрылымы логикалық және іздеу жүйелеріне (SEO) бейімделген болуы маңызды. Мазмұнын нақты анықтау веб-ресурстың бірегейлігін арттырып, пайдаланушылардың қызығушылығын сақтауға және сапалы коммуникация орнатуға мүмкіндік береді.

30.09.2025

### 1) Мазмұны басқару жүйесі (CMS)

Мазмұны басқару жүйесі (Content Management System – CMS) — веб-сайттағы ақпаратты жасау, өңдеу және жариялау процесін жеңілдететін бағдарламалық платформа. CMS көмегімен пайдаланушылар техникалық білімсіз-ақ контентті өзгертіп, жаңарта алады. Негізгі компоненттері — әкімшілік панелі, дерекқор және қолданушы интерфейсі. Әдетте CMS жүйелері модульдік құрылымға ие, яғни жаңа функциялар плагиндер арқылы оңай қосылады. Мысалы, WordPress — блогтар мен шағын сайттарға ыңғайлы болса, Drupal мен Joomla күрделі корпоративтік жобаларға жиі қолданылады. CMS қолдану веб-ресурстарды жылдам іске қосуға, дизайн мен құрылымды икемді басқаруға және командалық жұмысты тиімді ұйымдастыруға мүмкіндік береді.

### 2) Контентті басқару жүйесі: Бағдарламалық жасақтама салыстыру

Контентті басқару жүйелерін таңдауда олардың функционалдық мүмкіндіктерін салыстыру маңызды. WordPress — ең танымал CMS, қолдануға жеңіл әрі көптеген тегін плагиндер ұсынады. Joomla — көптілді және күрделі құрылымдағы сайттар үшін тиімді шешім. Drupal — жоғары қауіпсіздік пен кеңейтілген баптауларды қажет ететін корпоративтік және үкіметтік жобаларға лайық. Ал Shopify және Magento — электрондық коммерцияға бағытталған жүйелер. Әр CMS-тің артықшылығы мен шектеуі бар: WordPress — икемді және қарапайым, Drupal — кәсіби және тұрақты, ал Magento — сатылымға бейімделген. Таңдау ұйымның мақсаттарына, бюджетіне және техникалық талаптарына сай жасалуы тиіс.

01.10.2025

### 1) Bootstrap

Bootstrap — веб-әзірлеуде ең танымал CSS фреймворктердің бірі. Ол веб-сайттар мен қосымшаларды тез әрі тиімді стильдеуге арналған құралдар жиынтығын ұсынады. Bootstrap құрамында алдын ала дайындалған стильдер, торлық (grid) жүйесі, компоненттер (батырмалар, навигация, модаль терезелер) және JavaScript плагиндері бар. Фреймворк мобильді құрылғыларға бейімделген, яғни сайт экран өлшеміне қарай автоматты түрде өзгеріп отырады (responsive design). Bootstrap қолдану веб-әзірлеушілерге дизайн мен құрылымды нөлден жасауға уақыт жоғалтпай, стандартты және үйлесімді интерфейстер құруға мүмкіндік береді. Сонымен қатар, оның ашық коды бар болғандықтан, әлемдік қауымдастық арқылы тұрақты жаңартылып отырады.

### 2) Мазмұнды басқару жүйесі (CMS)

Мазмұнды басқару жүйесі (CMS) — веб-сайттағы ақпаратты жасау, өңдеу, сақтау және жариялауды автоматтандыратын бағдарламалық платформа. CMS көмегімен пайдаланушылар техникалық білімі шектеулі болса да контентті оңай жаңарта алады. Негізгі элементтері — әкімшілік панель, дерекқор және қолданушы интерфейсі. Танымал CMS мысалдары: WordPress — блог және шағын сайттар үшін, Joomla — көптілді сайттар үшін, Drupal — корпоративтік жобалар үшін. CMS модульдік құрылымға ие, жаңа функциялар плагиндер арқылы қосылады. Ол веб-ресурсты жылдам іске қосуға, контентті оңай басқаруға және командалық жұмысты үйлестіруге мүмкіндік береді.

### 3) Bootstrap жүктеу (онлайн)

Bootstrap-ті қолдану үшін оны онлайн түрде қосуға болады. Ең қарапайым әдіс — ресми CDN (Content Delivery Network) арқылы сілтеме беру. HTML файлының <head> бөліміне CSS стилінің сілтемесін, ал <body> соңына JavaScript пен Popper.js сілтемелерін қою жеткілікті. Онлайн жүктеу артықшылықтары: файлдарды локалды сақтау қажет емес, соңғы жаңартылған нұсқаны қолдануға болады, сайт жылдам жүктеледі. Сонымен қатар, көптеген хостинг және веб-әзірлеу платформалары Bootstrap CDN-ін қолдайды, сондықтан интеграция өте жеңіл. Бұл әдіс әсіресе прототип жасау немесе шағын жобалар үшін тиімді болып саналады.

02.10.2025

### 1) Bootstrap жүйесі және адаптивтілігі

Bootstrap — веб-әзірлеуде қолданылатын ең танымал CSS және JavaScript фреймворктердің бірі. Оның негізгі ерекшелігі — адаптивті (responsive) дизайнды қолдау. Бұл дегеніміз, сайт немесе қосымша әртүрлі экран өлшемдеріне — компьютер, планшет, смартфон — автоматты түрде бейімделеді. Bootstrap торлық жүйесі (grid system) арқылы контентті жолдар мен бағандарға бөліп орналастыруға мүмкіндік береді. Сонымен қатар, алдын ала дайындалған компоненттер мен стильдер веб-әзірлеушілерге уақытты үнемдеуге және бірізді дизайн жасауға көмектеседі. Адаптивтілік арқасында пайдаланушы тәжірибесі жақсарып, сайт барлық құрылғыларда ыңғайлы көрінеді.

### 2) Bootstrap мәтінге арналған стиль

Bootstrap мәтін элементтерін стильдеуді жеңілдетеді. Ол заголовоктар, абзацтар, сілтемелер, тізімдер және арнайы мәтіндік белгілер үшін дайын CSS кластарын ұсынады. Мысалы, .text-center — мәтінді ортасына орналастырады, .text-primary — басты түспен бояйды, .lead — басты абзац үшін қолдануға арналған. Сонымен қатар, Bootstrap мәтінді түстермен, өлшеммен, қалыңдықпен, курсивпен және көлеңкемен оңай безендіруге мүмкіндік береді. Бұл әдіс контентті визуалды түрде тартымды және үйлесімді етіп көрсетеді, сонымен қатар веб-сайт дизайнын бірізді ұстауға көмектеседі.

### 3) Bootstrap кестелермен жұмыс (онлайн)

Bootstrap кестелерді стильдеу және форматтауды жеңілдетеді. Ол арнайы CSS кластарын қолдану арқылы кестелерге көркемдік және интерактивті эффект береді. Мысалы, .table — негізгі стиль, .table-striped — жолдарды алма-кезек бояу, .table-bordered — шекара қосу, .table-hover — тышқанмен көрсеткенде жолдарды бөлектеу. Кестелердің адаптивтілігін қамтамасыз ету үшін .table-responsive класы қолданылады, бұл мобильді құрылғыларда көлденең айналдыруға мүмкіндік береді. Онлайн қолдану үшін Bootstrap CDN арқылы сілтеме беріп, HTML ішінде кесте элементтеріне қажетті кластарды қосу жеткілікті. Бұл әдіс локалды файлдарсыз веб-сайтты тез әрі әдемі жасауға мүмкіндік береді.

03.10.2025

### 1) Bootstrap түймелері

Bootstrap түймелері (buttons) веб-формалар мен интерфейс элементтерін интерактивті етуге арналған. Олар алдын ала дайындалған CSS кластарымен стильденеді, мысалы, .btn негізгі стиль, .btn-primary — басты түсті батырма, .btn-success — оң нәтиже көрсететін түсті батырма. Сонымен қатар, түймелер өлшемі бойынша (.btn-sm, .btn-lg), түсі мен күйі бойынша (hover, disabled) бапталады. Bootstrap түймелері адаптивті, яғни экран өлшеміне қарай оңай бейімделеді. JavaScript плагиндері арқылы кейбір батырмалар модаль терезелерді шақыру, dropdown мәзірін ашу немесе басқа интерактивті әрекеттерге жауап бере алады. Бұл әдіс веб-дизайнды бірізді әрі визуалды тартымды етеді.

### 2) Bootstrap форма

Bootstrap формалары веб-сайттағы ақпаратты жинау процесін жеңілдетеді. Ол мәтіндік өрістер, чекбокстар, радио батырмалар, таңдаулы тізімдер, textarea және батырмаларды стандартты және үйлесімді стильде көрсетеді. CSS кластар арқылы формаларды вертикаль, горизонталь немесе inline форматта орналастыруға болады. Мысалы, .form-control — енгізу өрістерін бірізді стильдеуге, .form-check — чекбокстарды дұрыс орналастыруға арналған. Bootstrap формалары адаптивті, яғни мобильді құрылғыларда да ыңғайлы көрінеді. JavaScript плагиндері арқылы формаларды валидациялау, интерактивті элементтер қосу мүмкіндігі бар.

### 3) Bootstrap шәрифтер мен суреттер

Bootstrap мәтіннің визуалды көрінісін және суреттерді басқаруды жеңілдетеді. Шәрифтерге арналған кластар: .text-primary, .text-muted, .font-weight-bold, .text-center және басқалары мәтіннің түсін, қалыңдығын, орнын анықтайды. Суреттерді стильдеу үшін .img-fluid класы қолданылады — сурет контейнерге бейімделіп, адаптивті болады. Сонымен қатар, .rounded — бұрыштарды жұмсақ етеді, .img-thumbnail — рамка қосады, .img-circle — шеңберге айналдырады. Бұл әдіс контентті визуалды үйлесімді әрі эстетикалық тартымды етіп көрсетеді, веб-сайт дизайнының бірізділігін сақтайды.

### 4) Bootstrap мәтінді панельдер (онлайн)

Bootstrap мәтінді панельдер (cards) — ақпаратты блоктарда көрсетуге арналған заманауи шешім. Олар тақырып, мәтін, сурет және батырмаларды біріктіре алады. Панельдерді жасау үшін .card, .card-body, .card-title, .card-text сияқты CSS кластар қолданылады. Адаптивтілік үшін .card-deck немесе .card-group арқылы бірнеше панельді қатар орналастыруға болады. Онлайн қолдану үшін Bootstrap CDN арқылы сілтеме беріп, HTML элементтеріне сәйкес кластарды қосу жеткілікті. Бұл әдіс веб-ресурстардағы контентті құрылымды, көрнекі және оқуға ыңғайлы етуге мүмкіндік береді.

06.10.2025

### JavaScript кіріспе

JavaScript — веб-әзірлеудің негізгі бағдарламалау тілдерінің бірі, ол веб-беттерге интерактивтілік пен динамикалық функционал қосуға арналған. HTML және CSS статикалық құрылымды және стильді қамтамасыз етсе, JavaScript пайдаланушы әрекеттеріне жауап беріп, контентті өзгертуге мүмкіндік береді. Мысалы, формаларды тексеру, батырмаға басқанда анимация жасау немесе динамикалық мазмұн көрсету сияқты әрекеттер JavaScript арқылы жүзеге асады. Ол браузерде тікелей орындалады және барлық негізгі веб-шолғыштарда қолданылады. JavaScript синтаксисі қарапайым, бірақ мүмкіндіктері кең: айнымалылар, функциялар, оқиғалар (events), циклдер және шартты операторлар сияқты негізгі құрылымдардан тұрады. Кіріспе деңгейінде оның негізгі идеясы — веб-бетпен өзара әрекеттесу мүмкіндігін беру.

07.10.2025

### 1) JS-те ақпаратты шығару

JavaScript бағдарламалау тілінде ақпаратты шығару веб-бетпен байланыс орнатудың негізгі тәсілдерінің бірі болып табылады. Ақпаратты шығару бірнеше әдіспен жүзеге асады: alert() — хабарлама терезесін көрсету, document.write() — HTML құжатына мәтін жазу, және innerHTML арқылы элементтің ішкі мазмұнын өзгерту. Мысалы, пайдаланушыға хабарлама көрсету немесе динамикалық контент қосу кезінде бұл әдістер қолданылады. Әр тәсілдің ерекшелігі бар: alert() интерактивті хабарлама береді, бірақ интерфейсті блоктайды; document.write() бет жүктелу кезінде қолайлы, ал innerHTML — қазіргі уақытта бет мазмұнын жаңарту үшін ең ыңғайлы тәсіл. Ақпаратты шығару негіздерін меңгеру — JavaScript-пен жұмыс істеудің алғашқы қадамы.

### 2) JS консольмен жұмыс

JavaScript консольі — бағдарламаны тексеру, қателерді анықтау және ақпаратты шығару үшін браузерде қолжетімді құрал. Консольді ашу үшін Chrome, Firefox немесе Edge шолғыштарында F12 пернесін басу немесе «Developer Tools» бөлімін қолдануға болады. console.log() функциясы ақпаратты консольге шығару үшін жиі қолданылады, ал console.error() немесе console.warn() — қателер мен ескерту хабарламаларын көрсетуге арналған. Консоль арқылы айнымалылардың мәнін тексеруге, функциялардың жұмысын бақылауға және кодты қадамдап орындауға болады. Бұл әдіс веб-әзірлеушіге бағдарламаның логикасын түсінуге және қателерді жылдам түзетуге мүмкіндік береді.

08.10.2025

### 1) JS-те қолданылатын кітапханалар

JavaScript кітапханалары — дайын функциялар мен модульдер жиынтығы, олар веб-әзірлеушіге кодты жылдам және тиімді жазуға мүмкіндік береді. Ең танымал кітапханалардың бірі — jQuery, ол DOM элементтерін таңдау, оқиғаларды өңдеу және анимация жасау процесін жеңілдетеді. Сондай-ақ Lodash массивтер мен объектілерді басқаруға, Moment.js уақыт пен даталарды өңдеуге, ал Chart.js графиктер мен диаграммаларды визуалдауға арналған. Кітапханаларды қолдану әзірлеушінің уақытын үнемдеп, кодты қысқартып, стандартты тапсырмаларды оңай орындауға мүмкіндік береді. Жаңадан үйренушілер үшін кітапханалар JavaScript-тің негізгі мүмкіндіктерін практикалық түрде қолдануды жеңілдетеді.

### 2) JS-тегі дерек түрлері

JavaScript тілінде әртүрлі деректер түрлері (data types) бар, олар бағдарламаның логикасын құруға қажет. Негізгі түрлері: Number — сандық мәндер, String — мәтіндік деректер, Boolean — true немесе false мәні, Null — бос мәнді көрсету, Undefined — анықталмаған айнымалы, Object — күрделі құрылымдар (объектілер мен массивтер). Әр дерек түрінің ерекшеліктерін білу бағдарламаны дұрыс жобалауға және қателіктерден сақтануға мүмкіндік береді. Мысалы, мәтіндік деректерді сандық есептеулерде қолдану қате нәтижелерге әкеледі. Дерек түрлерін дұрыс таңдау — JavaScript-те сенімді және тиімді код жазудың негізі.

### 3) JS айнымалылардың қолданылуы (онлайн)

JavaScript-те айнымалылар — деректерді уақытша сақтау және өңдеу үшін қолданылатын негізгі құрылым. Айнымалыларды жариялау үшін үш негізгі кілт сөз қолданылады: var, let және const. var — ескі стильдегі айнымалы, оның мәні функция деңгейінде сақталады; let — блок ішінде шектеулі және қазіргі заманғы практикада жиі қолданылады; const — тұрақты мәнді сақтау үшін, оны өзгерту мүмкін емес. Айнымалылар арқылы деректерді есептеу, функцияларға беру және веб-беттегі интерактивті әрекеттерді орындау мүмкін болады. Онлайн қолдануда айнымалылар браузерде тікелей тестілеу және динамикалық контентті көрсету үшін қолайлы.

09.10.2025

### 1) JavaScript-те айнымалылардың қолданылуы

JavaScript-те айнымалылар — деректерді уақытша сақтау және өңдеу үшін қолданылатын негізгі құрылым. Айнымалыларды жариялау үшін үш негізгі кілт сөз қолданылады: var, let және const. var — ескі стильдегі айнымалы, оның мәні функция деңгейінде сақталады; let — блок ішінде шектеулі және қазіргі заманғы практикада жиі қолданылады; const — тұрақты мәнді сақтау үшін, оны өзгерту мүмкін емес. Айнымалылар арқылы деректерді есептеу, функцияларға беру және веб-беттегі интерактивті әрекеттерді орындауға болады. Дұрыс қолданылған айнымалылар бағдарламаның логикасын түсінікті және басқаруға жеңіл етеді.

### 2) JavaScript математикалық операциялар

JavaScript-тағы математикалық операциялар сандармен жұмыс жасау үшін қолданылады. Негізгі операторлар: + (қосу), - (азайту), \* (косу), / (бөлу), % (қалдық алу). Сондай-ақ ++ және -- операторлары арқылы айнымалыны бірлікке арттыру немесе азайту мүмкіндігі бар. Математикалық функциялар Math объектісінде сақталған: Math.round() — дөңгелектеу, Math.floor() — төменгі бүтінге, Math.ceil() — жоғарғы бүтінге, Math.random() — кездейсоқ сан генерациялау. Математикалық операцияларды дұрыс қолдану веб-беттегі есептеулер мен динамикалық интерактивтілік үшін маңызды.

### 3) JavaScript бір өлшемді массивтер (онлайн)

JavaScript-та бір өлшемді массивтер (arrays) бір қатардағы бірнеше мәнді сақтау үшін қолданылады. Массив элементтері индекс бойынша нөлден басталып сақталады: [элемент1, элемент2, ...]. Массивтерді құру үшін let arr = [1, 2, 3]; синтаксисі қолданылады. Массив элементтеріне қол жеткізу үшін индекс қолданылады: arr[0] — бірінші элемент. Массивтерге жаңа элемент қосу үшін push() немесе unshift(), элементті шығару үшін pop() немесе shift() әдістері қолданылады. Онлайн тестілеуде массивтер динамикалық деректерді басқару, формаларды өңдеу және веб-бетте ақпаратты ұйымдастыру үшін ыңғайлы болып табылады.

10.10.2025

### 1) JS-тегі көп өлшемдегі массивтер

JavaScript-та көп өлшемді массивтер — бұл массивтің ішінде массивтер сақталатын құрылым. Олар матрицалар, кестелер немесе көп деңгейлі деректерді сақтау үшін қолданылады. Мысалы, екі өлшемді массивты құру: let arr = [[1, 2], [3, 4]];. Бір элементке қол жеткізу үшін екі индекс қолданылады: arr[0][1] — бірінші жолдағы екінші элемент. Көп өлшемді массивтер циклдер арқылы өңделеді, мысалы, for немесе forEach арқылы жолдар мен бағандарды өтуге болады. Бұл құрылым динамикалық деректерді ұйымдастыруға, веб-қосымшаларда кестелер мен графиктерді басқаруға мүмкіндік береді.

### 2) Контентті басқару ұғымы және мазмұн анықтау

Контентті басқару — бұл веб-ресурстардағы ақпаратты жасау, сақтау, өңдеу және таратуды ұйымдастыру үдерісі. Бұл процестің негізгі мақсаты — мазмұнның аудитория үшін қолжетімді және түсінікті болуын қамтамасыз ету. Мазмұн анықтау кезеңінде мақсатты аудиторияны, контент түрлерін (мәтін, сурет, бейне) және олардың жариялану уақытын анықтау маңызды. Дұрыс ұйымдастырылған контент аудиторияның қызығушылығын арттырып, сайттың немесе қосымшаның тиімділігін жоғарылатады. Контентті басқару жүйелері (CMS) осы үдерісті автоматтандырып, ақпаратты оңай жаңартуға мүмкіндік береді.

### 3) CMS және олардың түрлері

Мазмұнды басқару жүйесі (CMS) — веб-сайттағы контентті жасау, өңдеу және жариялауды автоматтандыратын бағдарламалық платформа. CMS түрлері әртүрлі: Open-source (WordPress, Joomla, Drupal) — тегін және кеңейтілетін, Proprietary (Adobe Experience Manager, Sitecore) — лицензиялық, корпоративтік шешімдер, E-commerce (Magento, Shopify) — онлайн дүкендерге бағытталған. CMS таңдау ұйымның мақсаттарына, бюджеттің көлеміне және техникалық талаптарға байланысты. Әр жүйенің ерекшелігі — функционалдығы, қауіпсіздігі және интерфейсінің ыңғайлылығы.

### 4) CMS қауіпсіздік және қорғау баптаулары (онлайн)

CMS қауіпсіздігі веб-сайтты хакерлік шабуылдан, деректерді ұрлаудан және зиянды бағдарламалардан қорғауға бағытталған. Негізгі шаралар: жүйені үнемі жаңарту, күрделі парольдер пайдалану, қосымша қауіпсіздік плагиндерін қосу, HTTPS протоколын қолдану. Сонымен қатар, файлдарға рұқсаттарды дұрыс орнату және резервтік көшірмелерді жасау маңызды. Онлайн конфигурация кезінде көптеген CMS платформалары — WordPress, Joomla, Drupal — қауіпсіздік баптауларын әкімшілік панель арқылы оңай орнатуға мүмкіндік береді. Бұл әдіс веб-ресурстың тұрақтылығын сақтап, пайдаланушылардың деректерін қорғауға көмектеседі.

13.10.2025

### 1) CMS-те қауіпсіздікті қорғау баптау

CMS қауіпсіздігін қамтамасыз ету — веб-сайтты хакерлік шабуылдардан, зиянды бағдарламалардан және деректерді ұрлаудан қорғаудың негізгі шарасы. Мұнда бірнеше маңызды қадам бар: жүйені үнемі жаңартып отыру, күрделі және бірегей парольдерді қолдану, қосымша қауіпсіздік плагиндерін орнату, HTTPS протоколын пайдалану. Сондай-ақ файлдарға рұқсаттарды дұрыс орнату, пайдаланушыларға шектеулі құқық беру және резервтік көшірмелер жасау ұсынылады. CMS платформалары көбінесе әкімшілік панель арқылы қауіпсіздік баптауларын оңай конфигурациялауға мүмкіндік береді. Дұрыс бапталған қауіпсіздік жүйесі веб-ресурстың тұрақтылығын арттырып, пайдаланушылардың деректерін сенімді қорғайды.

### 2) CMS үшін жеке шаблондарды жасау

Жеке шаблондар жасау CMS платформасында веб-сайттың бірегей дизайнын және құрылымын қамтамасыз етеді. Шаблондар HTML, CSS және кейде JavaScript кодынан тұрады. Жеке шаблон арқылы сайттың беттерін, меню құрылымын, виджеттер мен компоненттерді өз қалауыңыз бойынша реттеуге болады. Мысалы, WordPress-та бұл үшін child theme жасау ұсынылады, ол негізгі тақырыпты сақтай отырып, жеке стильдер мен функцияларды қосуға мүмкіндік береді. Жеке шаблон қолдану веб-сайтты визуалды ерекше етіп қана қоймай, функционалдық қажеттіліктерге толық бейімдеуге мүмкіндік береді.

14.10.2025

1) CMS үшін плагиндер жасау

CMS үшін плагиндер — веб-сайтқа қосымша функционал қосуға арналған бағдарламалық модульдер. Олар негізгі CMS ядросын өзгертпей, жаңа мүмкіндіктерді енгізуге мүмкіндік береді. Плагин жасау кезінде алдымен қажетті функцияларды анықтау, файл құрылымын дайындау және кодты дұрыс ұйымдастыру маңызды. Мысалы, WordPress-та плагин PHP файлынан басталып, арнайы комментарий арқылы анықталады, ал кейін әкімшілік панельден іске қосылады. Плагиндер формаларды кеңейту, жаңа виджеттер қосу, деректерді өңдеу немесе визуалды эффектілер жасау үшін қолданылады. Дұрыс жасалған плагин сайттың қауіпсіздігіне зиян келтірмей, функционалдығын арттырады.

### 2) PHP синтаксисі және негізгі құрылымы

PHP — серверлік бағдарлама тілі, ол динамикалық веб-беттер жасауға арналған. PHP коды HTML ішіне енгізіліп, серверде орындалады. PHP синтаксисі қарапайым: код <?php ... ?> тегтері арасында жазылады. Негізгі құрылымға айнымалылар ($variable), операторлар (+, -, \*, /), шартты конструкциялар (if, else), циклдер (for, while) және функциялар кіреді. PHP массивтер, объектілер және деректер базасымен жұмыс жасауға мүмкіндік береді. Ол MySQL сияқты деректер қорымен интеграцияланып, динамикалық контентті басқаруға арналған қуатты құрал болып саналады. PHP үйрену веб-әзірлеушіге серверлік логиканы тиімді ұйымдастыруға мүмкіндік береді.

15.10.2025

### 1) (PHP) Айнымалылар және деректер түрлері

PHP тілінде айнымалылар — деректерді уақытша сақтау және өңдеуге арналған негізгі құрылым. Айнымалыны $ белгісі арқылы жариялайды: $variableName = value;. PHP-де негізгі деректер түрлері бар: Integer — бүтін сандар, Float — ондық сандар, String — мәтіндік деректер, Boolean — true/false, Array — бірнеше мәндерді сақтау үшін, Object — объектілер, Null — бос мән. Айнымалыны дұрыс жариялау және дерек түрін таңдау бағдарламаның логикасын түсінікті және сенімді етеді. Деректер түрлерін өзгерту мүмкіндігі (type juggling) PHP-ді икемді етеді, бірақ қателіктерді болдырмау үшін мұқият қолдану қажет.

### 2) (PHP) Арифметикалық операторлар

PHP-де арифметикалық операторлар сандармен жұмыс жасау үшін қолданылады. Негізгі операторлар: + (қосу), - (азайту), \* (көбейту), / (бөлу), % (қалдық алу). Сонымен қатар ++ және -- операторлары арқылы айнымалыны бірлікке арттыру немесе азайтуға болады. PHP арифметикалық операциялар арқылы динамикалық есептеулер, формулаларды орындау және деректерді өңдеу жүзеге асырылады. Мысалы, қолданушыдан алынған мәндермен математикалық есептерді шығару немесе баға мен салықты есептеу сияқты әрекеттер осылай іске асады.

### 3) (PHP) Салыстыру және логикалық операторлар (онлайн)

PHP-де салыстыру және логикалық операторлар бағдарламаның логикасын басқаруға арналған. Салыстыру операторлары: == (тең), != (тең емес), > (үлкен), < (кіші), >=, <=. Логикалық операторлар: && (және), || (немесе), ! (емес) арқылы бірнеше шарттарды біріктіруге болады. Бұл операторлар if, elseif, else сияқты шартты конструкциялармен бірге қолданылады. Онлайн тестілеу кезінде операторларды пайдаланып, формаларды тексеру, деректерді сүзу немесе интерактивті әрекеттерді орындау жеңілдейді. Олар PHP-де бағдарламалық шешімдердің дұрыс жұмыс істеуін қамтамасыз етеді.

16.10.2025

### 1) (PHP) Тағайындау операторлары

PHP-де тағайындау операторлары айнымалыға мәнді беру үшін қолданылады. Негізгі оператор — = (тең), мысалы: $a = 5;. Сонымен қатар қосымша тағайындау операторлары бар: +=, -=, \*=, /=, %=. Олар айнымалының ағымдағы мәнін өзгертуді жеңілдетеді. Мысалы, $a += 3; — айнымалыға бұрынғы мәніне 3 қосу. Тағайындау операторларын қолдану есептеулерді қысқа және оқуға жеңіл кодпен жазуға мүмкіндік береді. PHP-дің икемді синтаксисі арқылы әртүрлі арифметикалық және мәтіндік операцияларды бір жолда орындауға болады.

### 2) (PHP) Шартты операторлар: if, else, else if

PHP-де шартты операторлар бағдарламаның орындалуын белгілі бір шартқа байланысты басқаруға мүмкіндік береді. if операторы шарт дұрыс болса, код блогын орындайды. else операторы шарт орындалмаған жағдайда орындалады. else if бірнеше шарттарды тексеруге арналған, оларды тізбектеп тексеруге болады. Мысалы:

if ($a > 10) {

echo "a 10-нан үлкен";

} elseif ($a == 10) {

echo "a 10-ға тең";

} else {

echo "a 10-нан кіші";

}

Бұл операторлар логикалық шешім қабылдауға, формаларды тексеруге және деректерді сүзуге мүмкіндік береді.

### 3) (PHP) Шартты операторлар: switch (онлайн)

PHP-де switch операторы бірнеше мәнді бірден тексеруге арналған. Ол әсіресе көптеген шарттарды if-else тізбектерімен салыстырғанда ыңғайлы. Синтаксис:

switch ($variable) {

case 1:

echo "Мәні 1";

break;

case 2:

echo "Мәні 2";

break;

default:

echo "Басқа мән";

}

case әрбір мүмкін мәнді тексереді, ал default ешбір шарт орындалмаса орындалады. Онлайн қолдануда switch формаларды өңдеуде, мәзірлерді таңдау немесе деректерді категориялау үшін ыңғайлы. Бұл оператор кодты қысқа және түсінікті етеді.

17.10.2025

### 1) (PHP) Циклдер: for, while, do…while

PHP-де циклдер бірдей операцияларды бірнеше рет орындау үшін қолданылады. for циклі алдын ала белгілі болған қадамдар саны үшін қолайлы:

for ($i = 0; $i < 5; $i++) {

echo $i;

}

while циклі шарт дұрыс болғанша орындалады:

while ($i < 5) {

echo $i;

$i++;

}

do…while циклі бір рет міндетті түрде орындалып, содан кейін шарт тексеріледі:

do {

echo $i;

$i++;

} while ($i < 5);

Циклдер массивтерді өңдеу, есептеулер жасау және веб-қосымшаларда интерактивті әрекеттер үшін маңызды.

### 2) (PHP) Break және continue операторлар

break және continue операторлары циклдерді басқаруға арналған. break циклді толық тоқтатады және орындалу циклдан кейінгі кодқа өтеді. continue ағымдағы итерацияны өткізіп, циклдің келесі қадамына өтеді. Мысалы, массивтегі кейбір мәндерді өткізіп немесе белгілі бір шартқа жеткенде циклді тоқтату қажет болғанда пайдаланылады. Бұл операторлар кодты ықшам әрі оқуға жеңіл етіп, логиканы басқаруды жеңілдетеді.

### 3) (PHP) Функция құру

PHP-де функция — кодты қайта қолдануға арналған блок. Функцияны function кілт сөзі арқылы жариялайды:

function myFunction() {

echo "Сәлем!";

}

myFunction();

Функциялар кодты құрылымды, оқуға жеңіл және тиімді етеді. Олар күрделі есептеулерді, деректерді өңдеуді және интерактивті әрекеттерді бірнеше рет қайталанбай орындауға мүмкіндік береді. Функциялар веб-қосымшалардың логикасын модульдерге бөліп ұйымдастыру үшін маңызды құрал болып табылады.

### 4) (PHP) Функция параметрлері және қайтару мәндері (онлайн)

Функция параметрлері функцияға сырттан деректер беру үшін қолданылады. Мысалы:

function greet($name) {

return "Сәлем, $name!";

}

echo greet("Айдар");

$name — параметр, ал return арқылы функциядан мән қайтарылады. Параметрлерді қолдану кодты икемді және қайта қолдануға ыңғайлы етеді. Қайтару мәндері функцияны есептеулердің нәтижесін алу немесе басқа код блоктарына беру үшін пайдаланылады. Онлайн тестілеу кезінде функция параметрлері мен қайтару мәндері динамикалық әрекеттерді оңай орындауға мүмкіндік береді.

20.10.2025

### 1) (PHP) Массив құру

PHP-де массивтер — бірнеше мәнді бір айнымалыда сақтау үшін қолданылатын құрылым. Массив екі түрге бөлінеді: бір өлшемді (single-dimensional) және көп өлшемді (multi-dimensional). Бір өлшемді массивті құру синтаксисі:

$fruits = array("алма", "банан", "апельсин");

немесе қысқаша:

$fruits = ["алма", "банан", "апельсин"];

Элементтерге индекстер арқылы қол жеткізуге болады: $fruits[0] — "алма". Көп өлшемді массивтер — массив ішінде массивтер, мысалы, кестелер немесе матрицалар үшін қолайлы. Массивтерге жаңа элемент қосу үшін array\_push() немесе индекс арқылы мән тағайындауға болады. PHP массивтері веб-қосымшаларда деректерді ұйымдастыру, формаларды өңдеу және динамикалық мазмұн жасау үшін маңызды құрал болып табылады.

21.10.2024

### 1) (PHP) Ассоциативті массив

PHP-де ассоциативті массив — элементтерді индекс арқылы емес, кілт (key) арқылы сақтау үшін қолданылатын массив түрі. Әрбір элементтің жеке атауы болады, мысалы:

$person = array(

"аты" => "Айдар",

"жасы" => 25,

"қаласы" => "Алматы"

);

Элементке қол жеткізу үшін кілтті қолданамыз: $person["аты"] — "Айдар". Ассоциативті массивтер веб-қосымшаларда деректерді кілт бойынша ұйымдастыруға, формалар мен базадан алынған ақпаратты өңдеуге ыңғайлы. Олар массив элементтерін логикалық түрде топтауға мүмкіндік береді және кодты оқуға жеңіл етеді.

### 2) (PHP) Көпөлшемді массив

Көпөлшемді массив — бұл массивтің ішінде массивтер сақталатын құрылым, яғни деректер матрицасы. Мысалы, екі өлшемді массив:

$matrix = array(

array(1, 2, 3),

array(4, 5, 6),

array(7, 8, 9)

);

Элементке қол жеткізу үшін екі индекс қолданылады: $matrix[0][1] — 2. Көпөлшемді массивтер кестелер, графиктер немесе күрделі деректер құрылымдарын сақтау және өңдеу үшін қолайлы. Циклдер арқылы оларды оңай өтуге және өңдеуге болады. PHP-де көпөлшемді массивтер динамикалық веб-қосымшалар үшін өте маңызды құрал болып саналады.

22.10.2025

### 1) Массивтерге қолданылатын функциялардың негізі

PHP-де массивтерді өңдеу үшін көптеген алдын ала анықталған функциялар бар. Негізгі функциялар:

* count() — массивтегі элементтер санын есептеу;
* array\_push() — массив соңына жаңа элемент қосу;
* array\_pop() — соңғы элементті шығару;
* array\_shift() — массивтің бірінші элементін шығару;
* array\_unshift() — массивтің басына элемент қосу;
* in\_array() — белгілі бір элементтің массивте бар-жоғын тексеру.

Бұл функциялар массивтерді басқаруды жеңілдетіп, деректерді динамикалық өңдеуге мүмкіндік береді. Массив функцияларын дұрыс қолдану веб-қосымшаларда деректерді тез және тиімді ұйымдастыруға септігін тигізеді.

### 2) ДҚБЖ ұғымы және түрлері

Деректер қорын басқару жүйесі (ДҚБЖ, Database Management System — DBMS) — деректерді ұйымдастыру, сақтау, басқару және іздеуді жеңілдететін бағдарламалық платформа. Негізгі түрлері:

* Реляциялық ДҚБЖ (MySQL, PostgreSQL, Oracle) — деректер кестелерде сақталады, кілттер арқылы байланысады;
* Құжаттық ДҚБЖ (MongoDB) — JSON немесе XML форматындағы құжаттарды басқарады;
* Кілт-мәліметтер ДҚБЖ (Redis) — кілт-мән жұптарын сақтайды;
* Графтық ДҚБЖ (Neo4j) — объектілер арасындағы байланысты граф түрінде көрсетеді.

ДҚБЖ веб-қосымшалар мен бағдарламалық жобаларда деректерді тиімді және қауіпсіз басқаруға мүмкіндік береді.

### 3) Жаңа кесте жасау CREATE TABLE (онлайн)

SQL-де жаңа кесте жасау үшін CREATE TABLE командасы қолданылады. Мысалы:

CREATE TABLE Students (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(50),

age INT,

city VARCHAR(50)

);

Бұл команда жаңа кестені анықтап, бағандарының атын, типін және негізгі кілтін көрсетеді. Онлайн тестілеу немесе деректер қорымен тәжірибе жасау кезінде бұл команда деректерді ұйымдастырудың алғашқы қадамы болып табылады. Кесте құрылымы дұрыс анықталған жағдайда, деректерді енгізу, өзгерту және сұрыптау оңай жүзеге асады.

23.10.2025

### 1) Кесте бағандарының деректер түрлері

SQL-де кесте бағандарының деректер түрлері әрбір бағанда сақталатын ақпараттың табиғатын анықтайды. Негізгі деректер түрлері:

* INT — бүтін сандар;
* VARCHAR(n) — мәтіндік деректер, n — символ саны;
* TEXT — ұзын мәтіндер;
* DATE — күн;
* DATETIME — күн мен уақыт;
* DECIMAL(p, s) — ондық сандар, p — жалпы разряд, s — ондық бөлшек разряды;
* BOOLEAN — true/false мәндер.

Деректер түрін дұрыс таңдау деректер қорын тиімді және қауіпсіз басқаруға, сұрауларды жылдам орындауға мүмкіндік береді.

### 2) Деректерді таңдау және шығару (SELECT)

SQL-де SELECT операторы кестеден деректерді шығару үшін қолданылады. Мысалы:

SELECT name, age FROM Students;

Бұл сұрау Students кестесінен name және age бағандарының мәндерін қайтарады. SELECT \* FROM Students; барлық бағандарды таңдайды. SELECT операторы деректерді сүзу, сұрыптау және қажетті ақпаратты алу үшін негіз болып табылады. Бұл команда веб-қосымшалар мен деректер қорында ақпаратты тиімді көрсетуге мүмкіндік береді.

### 3) WHERE арқылы шарт қою

SQL-де WHERE клаузы деректерді сұрауда шарт қою үшін қолданылады. Мысалы:

SELECT \* FROM Students WHERE age > 18;

Бұл сұрау 18 жасынан үлкен студенттердің жазбаларын қайтарады. WHERE-де әртүрлі салыстыру операторлары (=, >, <, >=, <=, !=) және логикалық операторлар (AND, OR, NOT) қолданылуы мүмкін. Шарттарды дұрыс қолдану деректерді сүзу, қажетсіз ақпаратты шығарып тастау және веб-қосымшаларда нақты нәтиже алу үшін маңызды.

24.10.2025

### 1) INSERT INTO арқылы деректерді енгізу

SQL-де INSERT INTO командасы кестеге жаңа жазбаларды қосу үшін қолданылады. Синтаксисі:

INSERT INTO Students (name, age, city) VALUES ('Айдар', 25, 'Алматы');

Бұл мысал Students кестесіне жаңа студент жазбасын қосады. INSERT INTO арқылы бір немесе бірнеше жолды бірден қосуға болады. Команда деректер қорында ақпаратты жаңарту және веб-қосымшаларға жаңа мәліметтер енгізу үшін негізгі құрал болып табылады.

### 2) JOIN, FULL JOIN кестелерді байланыстыру

SQL-де JOIN операторлары бірнеше кестені біріктіру үшін қолданылады. INNER JOIN — екі кестедегі сәйкес жазбаларды ғана шығарады. LEFT JOIN — сол жақтағы кестедегі барлық жазбаларды, сәйкес келгендерді оң жақ кестеден қосады. RIGHT JOIN керісінше оң жақ кестеге бағдарланады. FULL JOIN екі кестенің барлық жазбаларын қосады, сәйкес келмейтіндерге NULL мәнін тағайындайды. JOIN кестелер арасындағы байланысты көрсету, күрделі сұраулар жасау және деректерді біріктіру үшін қолданылады.

### 3) UPDATE деректерді жаңарту

SQL-де UPDATE командасы кестедегі бар жазбаларды өзгерту үшін қолданылады. Синтаксис:

UPDATE Students SET age = 26 WHERE name = 'Айдар';

Бұл мысалда Айдар есімді студенттің жасын 26 деп жаңартады. WHERE клаузы арқылы шарт қойылса, тек сәйкес жазбалар өзгереді, әйтпесе барлық жазбалар жаңартылады. UPDATE кестедегі деректерді түзету, түзетілген ақпаратты сақтау және веб-қосымшалардағы динамикалық мәліметтерді басқару үшін маңызды.

### 4) DELETE деректерді жою (онлайн)

SQL-де DELETE командасы кестеден жазбаларды жою үшін қолданылады. Мысалы:

DELETE FROM Students WHERE age < 18;

Бұл сұрау 18 жасынан кіші студенттердің жазбаларын жояды. WHERE клаузы көрсетілмесе, кестедегі барлық жазбалар өшіріледі, сондықтан сақтық қажет. DELETE кестені тазалау, қажетсіз деректерді жою және веб-қосымшалардағы ақпаратты басқаруға арналған негізгі құрал болып табылады. Онлайн практикада бұл команда деректер қорымен жұмыс істеудің негізгі операцияларының бірі болып есептеледі.

27.10.2025

*(25-Қазан демалыс)*

*28.10.2025*

### *1) PHP кодын HTML-ге қолдану*

PHP кодын HTML бетіне енгізу арқылы веб-беттерге динамикалық функционал қосуға болады. PHP коды <?php ... ?> тегтері арасында жазылады. Мысалы, HTML бетте пайдаланушыға сәлемдесу хабарламасын көрсету:

*<h1>Қош келдіңіз!</h1>*

*<?php*

*$name = "Айдар";*

*echo "<p>Сәлем, $name!</p>";*

*?>*

PHP HTML элементтерімен бірге жұмыс істеп, серверде орындалады. Бұл мүмкіндік формалардан деректерді алу, деректер қорымен байланысу, шартты логиканы орындау және динамикалық контентті көрсету үшін қолданылады. PHP кодын HTML бетіне дұрыс енгізу веб-сайттың интерактивтілігін арттырады және пайдаланушы тәжірибесін жақсартады.

### *2) HTML форма арқылы деректерді алу*

HTML формалар веб-беттерде пайдаланушыдан деректер жинау үшін қолданылады. Форманы құру үшін <form> тегі қолданылады, оның негізгі атрибуттары: action — деректерді жіберетін файл, method — деректерді жіберу тәсілі (GET немесе POST). Мысалы:

***<form action="process.php" method="post">***

***Атыңыз: <input type="text" name="username">***

***<input type="submit" value="Жіберу">***

***</form>***

PHP арқылы форма деректерін алу үшін $\_GET немесе $\_POST суперглобалды массивтерін қолданамыз:

*$name = $\_POST['username'];*

*echo "Сәлем, $name!";*

Бұл әдіс веб-қосымшаларда деректерді өңдеуге, есептеулер жасауға және пайдаланушымен интерактивті байланыс орнатуға мүмкіндік береді.