

Аманкулова Г.М., техникалық ғылым магистрі, негізгі автор,

<https://orcid.org/0000-0002-5566-5814>

«Жәңгір хан атындағы Батыс Қазақстан аграрлық-техникалық университеті» КеАҚ, 090009, Жәңгір хан көш., 51, Орал қ., Қазақстан Республикасы, gulfayrus.amankulova@bk.ru

Куангалиев Т.Г., техникалық ғылым магистрі, <https://orcid.org/0000-0002-0849-5875>

«Жәңгір хан атындағы Батыс Қазақстан аграрлық-техникалық университеті» КеАҚ, 090009, Жәңгір хан көш., 51, Орал қ., Қазақстан Республикасы, gaisiyevich@gmail.com

Мұхамбетов А.А., техникалық ғылым магистрі, <https://orcid.org/0000-0002-3022-4086>

«Жәңгір хан атындағы Батыс Қазақстан аграрлық-техникалық университеті» КеАҚ, 090009, Жәңгір хан көш., 51, Орал қ., Қазақстан Республикасы, amangeldy.abatovich@gmail.com

Amankulova G.M., Master, main author, <https://orcid.org/0000-0002-5566-5814>

«Zhangir Khan West Kazakhstan Agrarian-Technical University» NPJSC, 090009, 51 Zhangir Khan Str., Uralsk, Republic of Kazakhstan, gulfayrus.amankulova@bk.ru

Kuangaliyev T.G., Master, <https://orcid.org/0000-0002-0849-5875>

«Zhangir Khan West Kazakhstan Agrarian-Technical University» NPJSC, 090009, 51 Zhangir Khan Str., Uralsk, Republic of Kazakhstan, gaisiyevich@gmail.com

Mukhambetov A.A., Master, <https://orcid.org/0000-0002-3022-4086>

«Zhangir Khan West Kazakhstan Agrarian-Technical University» NPJSC, 090009, 51 Zhangir Khan Str., Uralsk, Republic of Kazakhstan, amangeldy.abatovich@gmail.com

JAVA-ДАҒЫ КЛИЕНТ-СЕРВЕРЛІК ҚОСЫМШАЛАР CLIENT SERVICE APPLICATIONS IN JAVA

Аннотация

Java тілі веб-қосымшалардың барлық түрлерінің негізі болып табылады және кіріктірілген және мобильді қосымшаларды, веб-мазмұнды, ойындар мен кәсіптік бағдарламалық қамтамасыз етуді дамытудың әмбебап стандарты болып табылады. Әлемде бағдарламалық қамтамасыз етуді тиімді әзірлеуге, сынауға, енгізуге және қолдануға мүмкіндік беретін Java -да қосымшалар жасайтын тоғыз миллионнан астам кәсіби мамандар бар. Қазіргі ақпараттық қоғамда ақпараттық технология құралдары үнемі жаңартылып отырады, бұл қазіргі таңда болашақ IT-мамандарды даярлауға жаңа талаптарын қояды. Жаңартылған бағдарламалық қамтамасыз ету, сондай -ақ жаңартылған немесе мүлде жаңа тілдер мен бағдарламалау ортасы жыл сайын дерлік пайда болады. Бірнеше жыл бұрын өзекті болған баспа оқулықтары қазір мағынасын жоғалтып отыр. Бұл жағдайда бағдарламалауды оқыу жаңа тәсілдерді іздеу қажет. Оқу -әдістемелік құралдар білім алушылардың оқытушымен тікелей қарым -қатынасы сияқты оқу процесінің құрамдас бөлігі болып табылады, сонымен қатар олар оқу орнының оқу -материалдық базасының элементі болып табылады.

ANNOTATION

The Java language is the basis for almost all kinds of web applications and is the universal standard for developing embedded and mobile applications, web content, games, and enterprise software. There are more than nine million professionals in the world who develop applications in Java, which allows you to effectively develop, test, implement and use software. In the modern information society, information technology tools are regularly updated, which today dictates new requirements for the training of future IT specialists. Updated software, as well as updated or completely new languages and programming environments, appear almost every year. Printed textbooks, which were relevant a few years ago, now lose their meaning and find no application. In these conditions, it is necessary to look for new approaches in teaching programming. Teaching aids are just as important a component in the educational process as the live communication of students

with a teacher, and they are also an element of the educational and material base of an educational institution.

Түйін сөздер: Java-ге бағытталған объектілі бағдарламалау тілі, сокеттер, клиент-сервер қосымшаларының архитектурасы, шифрлау алгоритмдері.

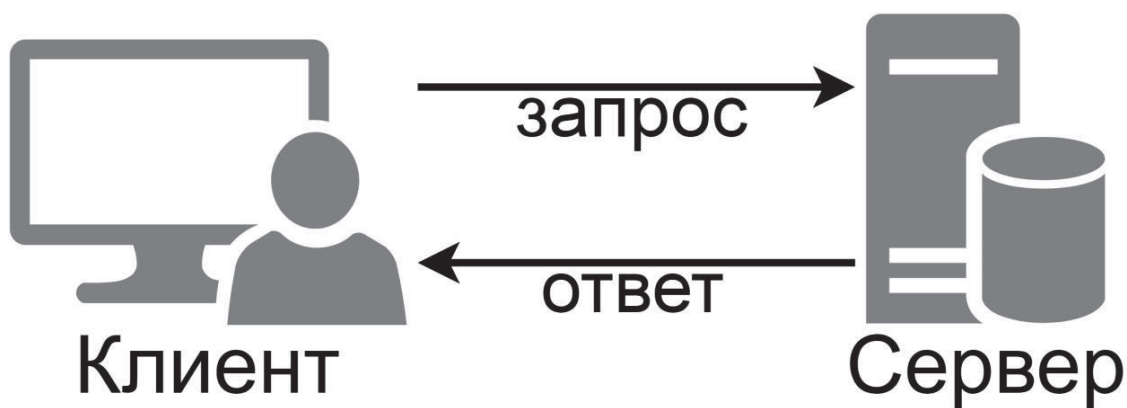
Keywords: object-oriented programming language Java, sockets, client-server application architectures, encryption algorithms.

Кіріспе: Адамдар қашанда ақпаратты қашықтықтан беру мүмкіндігіне қызығушылық танытты. Ақпарат алмасу үшін тек жеке ғана емес, алыс қашықтықта адамдар техникалық құралдарды ойлап тапты, бүкіл әлем бойынша кеңейтілген құрылымды кабельдік желілер, ұйымдастыру және пошта жүйелері, байланыс спутниктерін ұшырды. Ақпараттық технологиялардың дамуымен жаһандық байланыс одан да мүмкін болды. Бүгінде бүкіл әлем бойынша адамдар тек хабарламалармен ғана емес, сонымен қатар графикалық және дыбыстық және басқа да ақпарат түрлерімен оңай алмасады. Чаттардың «компьютерге дейінгі» тарихи предшественникі сөзсіз телефон болды. Телеграф та, пошта да нақты уақытта байланыс орната алмады және үйде болмады. Телефонның ойлап табылуы мен таралуы байланыс тәсілдері мен құралдарында нағыз революция тудырды. Басқа құрлықта нақты уақытта біреумен сөйлесе білу нағыз ғажайып сияқты көрінді [1-5].

Бағдарламалық қамтамасыз етуді құру процесі - бұл бағдарламалық қамтамасыз етуді құруға негізделген ережелер жиынтығы. Егер клиент-сервер архитектурасы болса, қосымшаны клиент-сервер қосымшасы деп атауға болады. Клиент-сервер қосымшасын әзірлеу клиент-сервер архитектурасын таңдаудан басталуы керек [5-10].

Клиент -сервер архитектурасы кем дегенде екі өзара әрекеттесетін, тәуелсіз процестердің болуымен сипатталады - клиент пен сервер.

Деректер базасы немесе файлдық жүйе қызметі сияқты қызметті орындайтын процестер серверлер деп аталады, ал серверлерден сұрау жіберу арқылы қызметтерді сұрайтын, содан кейін серверден жауап күтетін процестер клиенттер деп аталады. Шын мәнінде, бұл процестер әр түрлі компьютерлерге орнатылған және желілік протоколдарды қолдана отырып, компьютерлік желі арқылы өзара әрекеттесетін бағдарламалық қамтамасыз ету болып табылады, бірақ олар бір компьютерде орналасуы мүмкін [11-15].



Сурет 1 – Екі деңгейлі клиент-сервер архитектурасы

Кез келген клиент-сервер архитектурасы үш негізгі компоненттен тұруы керек:

- мәліметтерді енгізу және көрсету (қолданушы интерфейсі); - берілген пәндік аймаққа тән қолданбалы функциялар; - ақпараттық ресурстарды басқару (мәліметтер базасы немесе файлдық жүйе және т.б.).

Классикалық клиент-сервер архитектурасында қосымшаның үш негізгі бөлігін екі физикалық бағдарламалық қамтамасыз ету бойынша тарату қажет [16].

Ақпараттық технологияларды зерттейтін Gartner Group компаниясы екі деңгейлі (2 деңгейлі) клиент-сервер өзара әрекеттестігінің келесі бес моделін әзірлейді (екі деңгейлі деп аталады, себебі қосымшаның үш компоненті екі түйін арасында әр түрлі бөлінеді) [17].

Екі деңгейлі архитектураның артықшылықтары:

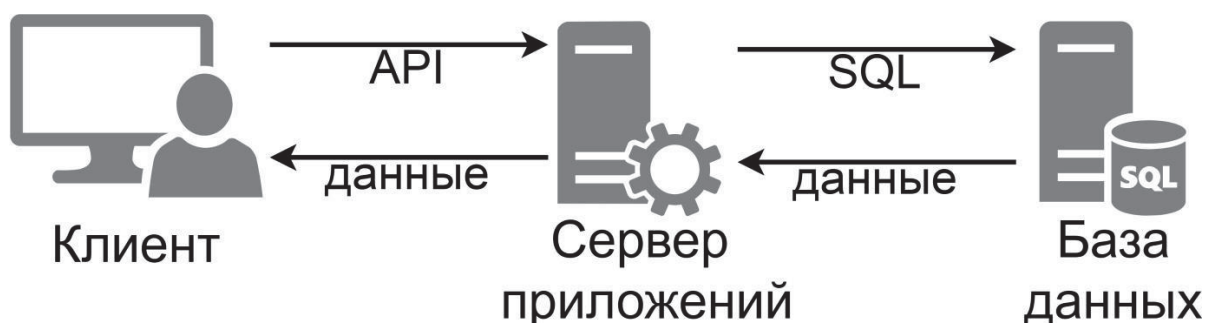
- Масштабтау мүмкіндігі бар; - Жабдық бойынша салыстырмалы үнемдеу.

Кемшіліктері:

- Масштабты тек тігінен (жүйе компоненттерінің өнімділігін арттыру); - Жаңа мүмкіндіктерді интеграциялаудың қиындығы (клиент пен сервер арасындағы үш компоненттің іске асуына байланысты).

Көп деңгейлі архитектура (N деңгейлі немесе көп деңгейлі, кейде оны үш деңгейлі архитектура немесе үш деңгейлі архитектура деп атайды, бірақ бұл ерекше жағдай)-бұл технологияны одан әрі жетілдіру [18-20].

«Клиент-сервер». Үш деңгейлі архитектурада бұрын клиенттік қосымшаларға енгізілген немесе толығымен серверде болған немесе осы екі сілтеме өз функцияларын өзара бөліскен барлық деректерді өңдеу қосымшалар сервері деп аталатын бөлек сілтемеге бөлінген.



Сурет 2 – Қатпарлы клиент-сервер архитектурасы

Деңгейлі архитектураның артықшылықтары:

- Масштабтылық (қолданба серверінің көлденең масштабталуы мен қосылымды мультиплекстеу есебінен);
- Тұтастық (деңгейлердің бір -бірінен оқшаулануына байланысты);
- Веб -браузерден немесе жұқа клиенттен қол жетімділік (өйткені клиент ақпаратты пайдаланушыға ыңғайлы түрде көрсетумен ғана айналысады);
- Жоғары қауіпсіздік; Кемшіліктері:
- Жоғары шығындар, қосымша әкімшілік шығындар;
- Мұндай жүйенің дамуы екі деңгейліге қарағанда әлдеқайда қиын.

Клиент-серверлік жүйе жүйенің сілтемелері арасында сақталатын және берілетін ақпараттың лайықты қауіпсіздігінің деңгейіне ие болуы керек. Мұндай жүйені жасаушы әр түрлі сілтемелердің қорғаныс құралдарының үйлестірілген жұмысына қол жеткізуі керек және қарапайым пайдаланушылардың өмірін қиындатпауы үшін, оларды қорғаныс жауаптарының көптеген қабаттары арқылы өтуге мәжбүр етпеуі үшін қажет. сұрақтардың бірдей түрі: «Күте тұрыңыз, кім барсыз ба?»

Ақпараттық жүйенің қауіпсіздігі деп жүйенің қалыпты жұмыс істеу үдерісіне кездейсоқ немесе қасақана араласудан, ақпаратты ұрлаудан (рұқсатсыз алу) және оны өзгертуден қорғаныс түсініледі. Ақпаратты байланыс арналары арқылы беру кезінде шифрлаудың криптографиялық алгоритмдері қолданылады. СА алгоритмдерінің бірнеше классификациясы бар, олардың біреуі белгілі бір алгоритмде қолданылатын кілттердің санына қарай бөлінеді:

- кілтсіз СА - есептеулерде ешқандай пернелерді қолданбаңыз; - бір пернелі СА - бір кілт параметрімен жұмыс (құпия кілт); - екі кілтті ОА - жұмыстың әр кезеңінде олар екі негізгі параметрді пайдаланады: құпия және ашық кілттер.

Хэшинг - бұл ақпаратты басқарудың түрлендіруі болып табылатын криптографиялық қорғау әдісі: шектеусіз мәліметтерден, криптографиялық түрлендірулерді орындау арқылы түпнұсқалық деректерге біркелкі сәйкес келетін ұзындықтың хэш мәні есептеледі.

Симметриялық шифрлауда бір кілт қолданылады, оның көмегімен шифрлау да, шифрлеу де бір симметриялы шифрлау алгоритмі арқылы орындалады. Бұл кілт шифрланған деректерді беру алдында екі жаққа қауіпсіз түрде беріледі.

Симметриялық шифрлауды екі түрге бөлуге болады: блок және ағын, кейбір классификацияларда олар бөлінбейді және ағынды шифрлау бірлік ұзындығының блоктарын шифрлау болып саналады.

Блоқты шифрлау ақпараттың белгіленген ұзындықтағы блоктарға алдын ала бөлінуімен сипатталады (мысалы, 64 немесе 128 бит). Сонымен қатар, әр түрлі криптографиялық алгоритмдерде немесе бір алгоритмнің әр түрлі жұмыс режимдерінде блоктарды бір -біріне тәуелсіз түрде де, «тізбектей» де шифрлауға болады, яғни ағымдағы мәліметтер блогының шифрлау нәтижесі тәуелді болғанда алдыңғы блоктың мәні немесе алдыңғы блоктың шифрлау нәтижесі бойынша.

Ағынды шифрлау, ең алдымен, ақпаратты блоктарға бөлуге болмайтын жағдайда қолданылады - мысалы, белгілі бір деректер ағыны бар, оның әрбір символы блок құруға жеткілікті қалған деректерді күтпей -ақ шифрленуі және жіберілуі тиіс. Ауысым жылдарында, электроника пайда болғанға дейін, ауысымдық регистрлерге негізделген ағынды шифрлар белсенді қолданылды. Оларды жобалау мен іске асыру оңай болды. Ағынды шифрлау алгоритмдері деректерді бит бойынша немесе символ бойынша шифрлайды. Мұндай алгоритмдердің беріктігі толығымен кілттер тізбегінің генераторының ішкі құрылымына байланысты. Егер генератор қысқа мерзімді тізбекті шығарса, онда жүйенің беріктігі төмен болады. Керісінше, егер генератор шынымен кездейсоқ биттердің шексіз тізбегін шығарса, онда сіз мінсіз беріктігі бар бір реттік төсеніш аласыз.

Java желісімен әрекеттесудің бірнеше әдісі бар. Ю пакеті бірінші және ұзақ уақыт бойы желіні қамтамасыз етудің жалғыз ықтимал әдісі болып қала берді. Оның мәні - розетканың сервер мен клиент жағынан жасалуында (клиент тарапынан, сервер жағынан - клиент тарапынан қосылу сұрауына жауап ретінде).

Әр розеткамен байланысты екі ағын бар (InputStream және OutputStream): біреуі хабарларды жіберу үшін, екіншісі хабарларды қабылдау үшін. Сонымен қатар, бұл ағындармен барлық операциялар бұғатталады, яғни. командалардың орындалуы осы операциялардың ұзақтығы үшін үзіледі. Бізге іздеу роботының бөлігі болып табылатын Web Crowler жасау керек делік. Бір сайтқа кіру үшін 300 мс қажет. Егер сізге 1000 сайтты индекстеу қажет болса, онда бұғаттау режимінде 300 x 1000 мс уақыт кетеді, яғни. 300 секунд.

Ю қолдану кезінде қосымшалардың жұмысын жылдамдату әдісі - бұл көп ағынды - тапсырма бөліктерге бөлінген, олардың әрқайсысы өзінің жеке бақылау жіпімен орындалады. Мысалы, 1000 торапты тексеру үшін оларды 250 тораптан тұратын төрт топқа бөлуге және әр топтан параллельді түрде тексеруге болады. Байланыстың ең танымал моделі - әр байланыс үшін жеке ағын жасау. Осылайша, барлық қосылымдар параллель қызмет көрсетеді және бір -біріне кедергі жасамайды. Бұл тәсілдің жетіспеушілігі - қызмет көрсетілетін қосылымдар санының өсуімен жіптер санының сызықтық өсуі және қажетті жад көлемі. Егер ағындар көп болса (ондаған мың), серверлік операциялық жүйе ағын контекстерін ауыстырумен айналысады және пайдалы жұмысқа уақыт қалмауы мүмкін.

Енгізу -шығаруды блоктау мәселелерін шешу үшін арналарды мультиплекстеуге негізделген механизм ойлап табылды. Бұл механизм Java 1.4.2 -де пайда болды және New IO (NIO) деп аталды. Механизмнің мәні мынада: мультиплексор (Java терминінде селектор, java.nio.channels.Selector деп аталатын) бар, ол бір тізбекте арналарды бірізділікпен сұрайды (егер желі өзара әрекеттескен жағдайда, java.nio.channels.SocketChannel және java.nio.channels.ServerSocketChannel сыныптары). Әр сауалнама нәтижесінде селектор енгізу - шығару операцияларын орындауға дайын арналардың идентификаторларын қайтарады (яғни арна қашықтағы жүйеге қосылған және енді оған сұрау жібере алады немесе керісінше қашықтағы жүйе арнаға бірдеңе жазды және енді бұл деректерді оқуға болады) одан). Мұндай идентификаторлар деп аталады. «Кілттер» (java.nio.channels.SelectionKey). Әр пернеде арна

қандай әрекетті орындауға дайын екендігі туралы ақпарат бар. Қолданбаның міндеті - барлық пернелерді айналдыру және сәйкес әрекеттерді орындау.

Бұл тәсілді қолданған кезде қосылымдар санының өсуімен ағындар санының сызықтық өсу мәселесінің шешімі барлық қосылған арналардың бір ағынмен қызмет етуі болып табылады. Алайда, мұндай шешім бағамен келеді, себебі желілік өзара әрекеттесу нәтижесінде алынған мәліметтер бойынша орындалатын операциялар өте қысқа болуы керек. Бір арнада кез келген қызметтің бұғатталуы басқаларына әсер етеді. Егер құлыптар ұзын болса, онда басқару ағыны соңғы дайын арналарға жеткенде, олар орнатқан байланыстар әрекетсіздіктен үзілуі мүмкін. Бұл мәселенің шешімі кейінге қалдырылған өңдеу деп аталуы мүмкін-бір NIO ағынында алынған мәліметтер негізінде бұғатталмайтын кезекке орналастырылған және бөлек ағынмен немесе бірнеше ағынмен орындалатын командалар құрылады. өз кезегінде енгізу - шығару операцияларына алаңдамайды.

Java NIO ағындардың ең аз санын пайдалана отырып, бірнеше құбырларды (желілік қосылымдар немесе файлдар) басқаруға мүмкіндік береді. Алайда, бұл тәсілдің құны блоктаушы ағындарды қолданғаннан гөрі деректерді талдаудың күрделірек.

Егер әзірлеушіге бір уақытта мыңдаған ашық қосылымдарды басқару қажет болса, олардың әрқайсысы аз ғана деректерді жібереді, Java NIO -ны таңдау артықшылық болады.

Қорытынды.

Java клиент-сервер қосымшаларын әзірлеудің ең жақсы құралдарының бірі деп қорытынды жасауға болады. Thread класын, Runnable интерфейсін, сондай-ақ клиент пен сервер розеткаларын қолдана отырып, сіз қарапайым клиент - сервер қосымшасын ғана емес, сонымен қатар файл серверін, дерекқор серверін, веб-серверді және пошта серверін жасай аласыз.

ӘДЕБИЕТТЕР ТІЗІМІ

1. Герберт Ш. "Java 8. Полное руководство 9-е издание" 2015
2. Дейтел Х. М., Дейтел П. Дж., Сантри С. И. Технологии программирования на Java 2. Распределенные приложения. 2011. –464 б.
3. Хорстманн К., Корнелл Г. "Java. Библиотека профессионала. Том 1 и 2". 9-е издание 2014г.
4. Яковлев А.В., Безбогов А.А., Родин В.В., Шамкин В.Н. Криптографическая защита информации: учебное пособие, – Тамбов:Изд-во Тамб. гос. техн. ун-та, 2006. – 140 с.
5. Анатольев А.Г., Компоненты сетевого приложения. Клиент-серверное взаимодействие и роли серверов. 06.12.2013.URL: <http://www.4stud.info/networking/lecture5.html>
6. Теория построения сетей – Технология клиент-сервер 13.1.2011г. URL: <http://xnets.ru/plugins/content/content.php?content.217.6> (дата обращения: 21.05.2017).
7. Компоненты сетевого приложения. Клиент-серверное взаимодействие и роли серверов. // Анатольев А.Г., 06.12.2013.URL: <http://www.4stud.info/networking/lecture5.html>
8. Иванова Г.С. Технология программирования: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2011. – 320 с. (Сер. Информатика в техническом университете). Электронная библиотека КазНУ им. Аль-Фараби: Ссылка: <http://elib.kaznu.kz>.
9. Программирование. В 2 т. Т. 1.: учебник для студ. вузов / Э. А. Нигматуллина и др.; ред. Н. И. Пак. – М.: Издательский центр "Академия", 2013. – 272 с. Ссылка: <https://b-ok.asia/book/2841574/c50dcc?regionChanged=&redirect=40958560>
10. Беркинбаев К.М., Нуруллаев А. Н. Практикум по программированию : учебное метод. пособие / отв. ред. Т. Б. Байтелиев. – Алматы: Юрид. лит., 2009. – 34 с. Республиканская межвузовская электронная библиотека (РМЭБ): Ссылка: <http://rmebrk.kz/epub>.
11. Барков И.А. Преподавание дисциплины «объектно-ориентированное программирование» // Образовательные технологии и общество. Том 12. 2009. 494 – 500 с
12. Java / Википедия – свободная энциклопедия. [Электронный ресурс]. Ссылка: <https://ru.wikipedia.org/wiki/Java>
13. Компьютерная программа / Википедия – свободная энциклопедия. [Электронный ресурс]. Ссылка: https://ru.wikipedia.org/wiki/Компьютерная_программа

14. Объектно-ориентированный язык программирования / Википедия – свободная энциклопедия. [Электронный ресурс]. Ссылка: https://ru.wikipedia.org/wiki/Объектно-ориентированный_язык_программирования
15. Язык программирования / Википедия – свободная энциклопедия. [Электронный ресурс]. Ссылка: https://ru.wikipedia.org/wiki/Язык_программирования
16. Дубаков А.А. Сетевое программирование: учебное пособие / А.А. Дубаков – СП: НИУ ИТМО, 2013. – С. 248.
17. СТ РК ISO/IEC/IEEE 26512-2012. Разработка программного обеспечения и проектирование систем. Требования к заказчикам и поставщикам документации пользователя. [Электронный ресурс]. Ссылка: <https://online.zakon.kz>
18. СТ РК ISO/IEC/IEEE 26511-2012. Разработка программного обеспечения и проектирование систем. Требования к менеджерам по документации пользователя. [Электронный ресурс]. Ссылка: <https://online.zakon.kz>.
19. Гурвиц Г. Разработка реального приложения в среде клиент-сервер –ДВГУПС, 2005, 206 с.
20. Алексей В. Самоучитель Java. – Издательство: Наука и Техника, 2017, 368 с.

ADEBIETTER TIZIMI

1. Gerbert SH. " Java 8. Polnoe rukovodstvo 9-e izdanie" 2015
2. Dejtel H. M., Dejtel P. Dzh., Santri S. I. Tekhnologii programmirovaniya na Java 2. Raspredelemnnye prilozheniya. 2011. – 464 b.
3. Horstmann K., Kornell G. "Java. Biblioteka professionala. Tom 1 i 2". 9-e izdanie 2014g.
4. YAKovlev A.V., Bezbogov A.A., Rodin V.V., SHamkin V.N. Kriptograficheskaya zashchita informacii: uchebnoe posobie, – Tambov: Izd-vo Tamb. gos. tekhn. un-ta, 2006. – 140 s.
5. Anatol'ev A.G., Komponenty setevogo prilozheniya. Klient-servernoe vzaimodejstvie i roli serverov. 06.12.2013. URL: <http://www.4stud.info/networking/lecture5.html>
6. Teoriya postroeniya setej – Tekhnologiya klient-server 13.1.2011g. URL: <http://xnets.ru/plugins/content/content.php?content.217.6> (data obrashcheniya: 21.05.2017).
7. Komponenty setevogo prilozheniya. Klient-servernoe vzaimodejstvie i roli serverov. // Anatol'ev A.G., 06.12.2013. URL: <http://www.4stud.info/networking/lecture5.html>
8. Ivanova G.S. Tekhnologiya programmirovaniya: Uchebnik dlya vuzov. – M.: Izd-vo MGTU im. N.E. Baumana, 2011. – 320 s. (Ser. Informatika v tekhnicheskom universitete). Elektronnaya biblioteka KazNU im. Al'-Farabi: Ssylka: <http://elib.kaznu.kz>
9. Programmirovaniye. V 2 t. T. 1. : uchebnik dlya stud. vuzov / E. A. Nigmatullina i dr. ; red. N. I. Pak. – M.: Izdatel'skij centr "Akademiya", 2013. – 272 s.
Ssylka:<https://b-ok.asia/book/2841574/c50dcc?regionChanged=&redirect=40958560>.
10. Berkinbaev K.M., Nurullaev A. N. Praktikum po programmirovaniyu : uchebnoe metod. posobie / otv. red. T. B. Bajteliev. – Almaty : YUrid. lit., 2009. – 34 s. Respublikanskaya mezhvuzovskaya elektronnaya biblioteka (RMEB): Ssylka: <http://rmebrk.kz/epub>
11. Barkov I.A. Prepodavaniye discipliny «obektno-orientirovannoe programmirovaniye» // Obrazovatel'nye tekhnologii i obshchestvo. Tom 12. 2009. 494 – 500 s
12. Java / Vikipediya – svobodnaya enciklopediya. [Elektronnyj resurs]. Ssylka: <https://ru.wikipedia.org/wiki/Java>
13. Komp'yuternaya programma / Vikipediya – svobodnaya enciklopediya. [Elektronnyj resurs]. Ssylka: https://ru.wikipedia.org/wiki/Komp'yuternaya_programma.
14. Obektno-orientirovannyy yazyk programmirovaniya / Vikipediya – svobodnaya enciklopediya. [Elektronnyj resurs]. Ssylka: https://ru.wikipedia.org/wiki/Ob'ektno-orientirovannyy_yazyk_programmirovaniya
15. YAzyk programmirovaniya / Vikipediya – svobodnaya enciklopediya. [Elektronnyj resurs]. Ssylka: https://ru.wikipedia.org/wiki/YAzyk_programmirovaniya
16. Dubakov A.A. Ssetvoe programmirovaniye: uchebnoe posobie / A.A. Dubakov – SP: NIU ITMO, 2013. – S. 248.

17. ST RK ISO/IEC/IEEE 26512-2012. Razrabotka programmnoгo obespecheniya i proektirovanie sistem. Trebovaniya k zakazchikam i postavshchikam dokumentacii pol'zovatelya. [Elektronnyj resurs]. Ssylka: <https://online.zakon.kz>
18. ST RK ISO/IEC/IEEE 26511-2012. Razrabotka programmnoгo obespecheniya i proektirovanie sistem. Trebovaniya k menedzheram po dokumentacii pol'zovatelya. [Elektronnyj resurs]. Ssylka: <https://online.zakon.kz>
19. Gurvic G. Razrabotka real'nogo prilozheniya v srede klient-server –DVGUPS, 2005, 206 s.
20. Aleksej V. Samouchitel' Java. – Izdatel'stvo: Nauka i Tekhnika, 2017, 368 s.

РЕЗЮМЕ

Язык Java является основой практически для всевозможных типов сетевых приложений и всеобщим стандартом для разработки встроенных и мобильных приложений, веб-контента, игр, и корпоративного программного обеспечения. В мире насчитывается более девяти миллионов специалистов, разрабатывающих приложения на Java, которая позволяет эффективно разрабатывать, тестировать, внедрять и использовать программное обеспечение. В современном информационном обществе регулярно обновляются средства информационных технологий, что диктует в наши дни новые требования к обучению будущих ИТ-специалистов. Обновленное программное обеспечение, а также обновленные или же совершенно новые языки и среды программирования появляются практически ежегодно. Печатные учебные пособия, которые были актуальны еще несколько лет назад, сейчас уже теряют 20 свой смысл и не находят применения. В данных условиях нужно искать новые подходы в обучении программированию. Средства обучения - такой же важный компонент в образовательном процессе, как и живое общение студентов с преподавателем, а также они являются элементом учебно-материальной базы образовательного учреждения.