

МИНОБРНАУКИ РОССИИ  
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ЭКОНОМИКИ И СЕРВИСА  
КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

Рабочая программа дисциплины (модуля)

## ПРОГРАММИРОВАНИЕ НА JAVA

Направление и направленность (профиль)

09.03.02 Информационные системы и технологии. Информационные системы и технологии

Год набора на ОПОП  
2017

Форма обучения  
заочная

Владивосток 2019

Рабочая программа дисциплины (модуля) «Программирование на Java» составлена в соответствии с требованиями ФГОС ВО по направлению(ям) подготовки 09.03.02 Информационные системы и технологии (утв. приказом Минобрнауки России от 12.03.2015г. №219) и Порядком организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры (утв. приказом Минобрнауки России от 05.04.2017 г. N301).

Составитель(и):

*Гриняк В.М., доктор технических наук, профессор, Кафедра информационных технологий и систем, Viktor.Grinyak@vvsu.ru*

Утверждена на заседании кафедры информационных технологий и систем от 29.05.2019 , протокол № 10

СОГЛАСОВАНО:

Заведующий кафедрой (разработчика) \_\_\_\_\_  
*подпись* *фамилия, инициалы*

Заведующий кафедрой (выпускающей) \_\_\_\_\_  
*подпись* *фамилия, инициалы*

## 1. Цель и задачи освоения дисциплины (модуля)

Целью изучения дисциплины «Программирование на Java» является усвоение и закрепление основных приемов, методов и принципов работы при создании кроссплатформенных программ, усвоение навыков использования языка Java, подготовка к профессиональной сертификации.

Основные задачи изучения дисциплины:

- развитие навыков объектно-ориентированного программирования;
- освоение подходов к созданию консольных и визуальных кроссплатформенных программ;
- знакомство с понятиями и языком предметной области, в том числе международной англоязычной терминологией;
- развитие навыков работы в коллективе;
- подготовка к профессиональной сертификации Java-программиста.

## 2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Планируемыми результатами обучения по дисциплине являются знания, умения, навыки, соотнесенные с компетенциями, которые формирует дисциплина, и обеспечивающие достижение планируемых результатов по образовательной программе в целом. Перечень компетенций, формируемых в результате изучения дисциплины (модуля), приведен в таблице 1.

Таблица 1 – Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля)

| Название ОПОП ВО, сокращенное                               | Код компетенции | Формулировка компетенции                                                                                                                                                                                                                                                                                                                               | Планируемые результаты обучения |                                                                                                      |
|-------------------------------------------------------------|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------|------------------------------------------------------------------------------------------------------|
| 09.03.02<br>«Информационные системы и технологии»<br>(Б-ИС) | ПК-2            | Способность проводить техническое проектирование                                                                                                                                                                                                                                                                                                       | Знания:                         | основных принципов и этапов технического проектирования                                              |
|                                                             |                 |                                                                                                                                                                                                                                                                                                                                                        | Умения:                         | проводить техническое проектирование                                                                 |
|                                                             |                 |                                                                                                                                                                                                                                                                                                                                                        | Навыки:                         | владения основными принципами технического проектирования                                            |
|                                                             | ПК-17           | Способность использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент, банковские системы, безопасность информационных систем, управление технологическими | Знания:                         | общих технологий разработки объектов профессиональной деятельности в различных областях              |
|                                                             |                 |                                                                                                                                                                                                                                                                                                                                                        | Умения:                         | использовать общие технологии разработки объектов профессиональной деятельности в различных областях |

|  |       |                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                       |                                                                                                                                                                                                                                                                                     |
|--|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  |       | процессами, механика,<br>техническая физика,<br>энергетика, ядерная<br>энергетика, силовая<br>электроника,<br>металлургия,<br>строительство,<br>транспорт,<br>железнодорожный<br>транспорт, связь,<br>телекоммуникации,<br>управление<br>инфокоммуникациями,<br>почтовая связь,<br>химическая<br>промышленность,<br>сельское хозяйство,<br>текстильная и легкая<br>промышленность,<br>пищевая<br>промышленность,<br>медицинские и би | Навыки:                               | владения общими технологиями<br>разработки объектов<br>профессиональной<br>деятельности в различных<br>областях                                                                                                                                                                     |
|  | ПК-32 | Способность<br>адаптировать<br>приложения к<br>изменяющимся<br>условиям<br>функционирования                                                                                                                                                                                                                                                                                                                                          | Знания:<br><br>Умения:<br><br>Навыки: | основных принципов<br>адаптации приложений к<br>изменяющимся условиям<br>функционирования<br><br>адаптировать приложения к<br>изменяющимся условиям<br>функционирования<br><br>владения основными<br>принципами адаптации<br>приложений к изменяющимся<br>условиям функционирования |

### 3. Место дисциплины (модуля) в структуре основной образовательной программы

Отнесение дисциплины к вариативной части учебного плана определяется спецификой и миссией ВГУЭС, а также особенностями взаимодействия ВГУЭС с рынком труда и региональными требованиями, выраженными в результатах образования и компетенциях.

Входными требованиями, необходимыми для освоения дисциплины, является наличие у обучающихся компетенций, сформированных при изучении дисциплин и/или прохождении практик «Алгоритмизация и программирование», «Инженерная и компьютерная графика модуль 1», «Инженерная и компьютерная графика модуль 2», «Информатика и основы программирования», «Методы и технологии продвижения информационных ресурсов», «Объектно-ориентированное программирование», «Программирование на С», «Технология программирования», «Учебная практика по

получению первичных профессиональных умений и навыков». На данную дисциплину опираются «Базы данных продвинутой курс», «Базы данных», «Введение в анализ больших данных», «Веб-программирование», «Геоинформационные системы», «Защита выпускной квалификационной работы, включая подготовку к процедуре защиты и процедуру защиты», «Интеллектуальные информационные системы», «Информационная безопасность и защита информации», «ИТ-инфраструктура и прикладные системы предприятий», «Курсовое проектирование», «Предметно-ориентированное программирование», «Программирование для мобильных устройств».

#### 4. Объем дисциплины (модуля)

Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу с обучающимися (по видам учебных занятий) и на самостоятельную работу, приведен в таблице 2.

Таблица 2 – Общая трудоемкость дисциплины

| Название ОПОП<br>ВО                                   | Форма<br>обуче-<br>ния | Часть<br>УП | Семестр<br>(ОФО)<br>или курс<br>(ЗФО,<br>ОЗФО) | Трудо-<br>емкость | Объем контактной работы (час) |            |       |      |                    |     | СРС | Форма<br>аттес-<br>тации |
|-------------------------------------------------------|------------------------|-------------|------------------------------------------------|-------------------|-------------------------------|------------|-------|------|--------------------|-----|-----|--------------------------|
|                                                       |                        |             |                                                | (З.Е.)            | Всего                         | Аудиторная |       |      | Внеауди-<br>торная |     |     |                          |
|                                                       |                        |             |                                                |                   |                               | лек.       | прак. | лаб. | ПА                 | КСР |     |                          |
| 09.03.02<br>Информационные<br>системы и<br>технологии | ЗФО                    | Бл1.В       | 3                                              | 3                 | 17                            | 6          | 10    | 0    | 1                  | 0   | 91  | ДЗ                       |

#### 5. Структура и содержание дисциплины (модуля)

##### 5.1 Структура дисциплины (модуля) для ЗФО

Тематический план, отражающий содержание дисциплины (перечень разделов и тем), структурированное по видам учебных занятий с указанием их объемов в соответствии с учебным планом, приведен в таблице 3.1

Таблица 3.1 – Разделы дисциплины (модуля), виды учебной деятельности и формы текущего контроля для ЗФО

| № | Название темы                                                                        | Кол-во часов, отведенное на |      |     |     | Форма<br>текущего контроля                            |
|---|--------------------------------------------------------------------------------------|-----------------------------|------|-----|-----|-------------------------------------------------------|
|   |                                                                                      | Лек                         | Прак | Лаб | СРС |                                                       |
| 1 | Синтаксис языка Java, классы в языке Java. Наследование и инкапсуляция в языке Java. | 1                           | 0    | 2   | 13  | текущий опрос, отчет о выполнении лабораторной работы |
| 2 | Разработка классов в языке Java. Наследование и интерфейсы в языке Java.             | 1                           | 0    | 1   | 14  | текущий опрос, отчет о выполнении лабораторной работы |
| 3 | Обобщённые типы и коллекции значений в языке Java. Работа со строками в языке Java.  | 1                           | 0    | 1   | 14  | текущий опрос, отчет о выполнении лабораторной работы |
| 4 | Обработка исключений. Ввод и вывод в Java программах. Файловый ввод и вывод.         | 1                           | 0    | 2   | 14  | текущий опрос, отчет о выполнении лабораторной работы |
| 5 | Многопоточные программы Java. Параллельное программирование Java.                    | 1                           | 0    | 2   | 14  | текущий опрос, отчет о выполнении лабораторной работы |

|                         |                                                                                        |          |          |           |           |                                                       |
|-------------------------|----------------------------------------------------------------------------------------|----------|----------|-----------|-----------|-------------------------------------------------------|
| 6                       | Построение приложений баз данных с использованием JDBC API. Локализация Java программ. | 1        | 0        | 2         | 14        | текущий опрос, отчет о выполнении лабораторной работы |
| <b>Итого по таблице</b> |                                                                                        | <b>6</b> | <b>0</b> | <b>10</b> | <b>83</b> |                                                       |

## 5.2 Содержание разделов и тем дисциплины (модуля) для ЗФО

*Тема 1 Синтаксис языка Java, классы в языке Java. Наследование и инкапсуляция в языке Java.*

Содержание темы: Структура классов Java. Программные блоки и комментарии. Переменные. Условные операторы и циклы. Инкапсуляция при разработке классов Java. Моделирование задачи с использованием классов Java. Неизменяемые классы. Подклассы: создание и использование. Перегрузка методов класса. Методы с переменным числом аргументов. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, лабораторная работа.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, лабораторным работам.

*Тема 2 Разработка классов в языке Java. Наследование и интерфейсы в языке Java.*

Содержание темы: Спецификаторы доступа private, protected, default и public. Перегрузка конструкторов и других методов. Использование оператора instanceof для определения типа объекта. Виртуальный вызов методов класса. Преобразование типов «вверх» (апкостинг) и «вниз» (даункостинг). Перегрузка методов класса Object. Использование абстрактных классов. Ключевые слова final и static. Шаблон проектирования singleton. Вложенные классы. Интерфейсы в Java, определение интерфейсов. Особенности использования интерфейсов и классов в программах. Расширение интерфейсов. Рефакторинг кода. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, лабораторная работа.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, лабораторным работам.

*Тема 3 Обобщённые типы и коллекции значений в языке Java. Работа со строками в языке Java.*

Содержание темы: Обобщённые типы как способ создания классов в Java. Создание объектов в рамках обобщённого типа. Создание коллекций без использования обобщённых типов и с их использованием. Работа со структурами данных ArrayList, Set, HashMap. Реализация стека и очереди. Перечислимые типы. Чтение данных из командной строки. Поиск строк. Парсинг строк. Создание строк с использованием класса StringBuilder. Поиск в строке, парсинг строки и удаление строк с использованием регулярных выражений. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, лабораторная работа.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, лабораторным работам.

*Тема 4 Обработка исключений. Ввод и вывод в Java программах. Файловый ввод и вывод.*

Содержание темы: Типы исключений в Java. Использование конструкций try и throw. Использование catch, единожды и многократно. Ключевое слово finally. Классы исключений. Создание выборочных исключений и автозакрываемых ресурсов. Использование assertions.

Основы ввода и вывода в Java программах. Чтение данных с консоли и вывод данных на консоль. Использование потоков для чтения и записи файлов. Чтение и запись объектов с использованием сериализации. Использование интерфейса Path для работы с файлами. Работа с классом Files для операций над файлами. Канальный и потоковый ввод-вывод в файлах. Работа с атрибутами файлов. Доступ к дереву каталогов. Поиск файлов с использованием класса PathMatcher. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, лабораторная работа.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, лабораторным работам.

#### *Тема 5 Многопоточные программы Java. Параллельное программирование Java.*

Содержание темы: Определение и создание потоков. Управление потоками. Синхронизация потоков. Проблемы многопоточного программирования. Атомарные переменные. Метод ReentrantReadWriteLock(). Работа с коллекцией java.util.concurrent. Синхронизирующие классы. Использование ExecutorService. Fork-Join фреймворк. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, лабораторная работа.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, лабораторным работам.

#### *Тема 6 Построение приложений баз данных с использованием JDBC API. Локализация Java программ.*

Содержание темы: Основные функции JDBC API. Подключение к базе данных с использованием драйвера JDBC. Подача запросов получение результатов из базы данных. Транзакции и JDBC. Использование паттерна Data Access Object. Особенности и задачи локализации программ. Определение и представление локализуемых данных. Чтение и установка локализуемых данных с помощью объекта Locale. Построение ресурсов. Вызов ресурсов из приложений. Форматирование текста и его локализация с использованием NumberFormat DateFormat. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, лабораторная работа.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, лабораторным работам.

## **6. Методические указания по организации изучения дисциплины (модуля)**

В ходе изучения дисциплины «Программирование на Java» студенты могут посещать аудиторные занятия (лекции, лабораторные занятия, консультации).

Особое место в овладении частью тем данной дисциплины может отводиться самостоятельной работе, при этом во время аудиторных занятий могут быть рассмотрены и проработаны наиболее важные и трудные вопросы по той или иной теме дисциплины, а второстепенные и более легкие вопросы, а также вопросы, специфичные для ОПОП, могут быть изучены студентами самостоятельно.

В соответствии с учебным планом процесс изучения дисциплины может предусматривать проведение лекций, лабораторных занятий, консультаций, а также самостоятельную работу студентов. Обязательным является проведение лабораторных занятий в специализированных компьютерных аудиториях, оснащенных подключенными к центральному серверу терминалами или персональными компьютерами.

В рамках общего объема часов, отведенных для изучения дисциплины, предусматривается выполнение следующих видов самостоятельных работ студентов:

подготовка к лабораторным занятиям, работа над рекомендованной литературой, самостоятельное решение задач.

Практикуется интерактивный способ работы над задачами: студенты размещают решение задачи на форуме, указанном преподавателем, проходит обсуждение задачи и её решения, после чего преподаватель даёт свою оценку всем участникам процесса.

По тематике дисциплины выпущено большое количество печатных изданий. Электронные ресурсы могут быть использованы как источники справочной и обзорной информации по тематике дисциплины.

- Информационные технологии: Microsoft Windows 7 Russian
- Материально-техническое обеспечение: Компьютеры

#### **Особенности организации обучения для лиц с ограниченными возможностями здоровья и инвалидов.**

При необходимости обучающимся из числа лиц с ограниченными возможностями здоровья и инвалидов (по заявлению обучающегося) предоставляется учебная информация в доступных формах с учетом их индивидуальных психофизических особенностей:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания, консультации и др.
- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания, консультации и др.
- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; индивидуальные задания, консультации и др.

### **7. Фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине (модулю)**

В соответствии с требованиями ФГОС ВО для аттестации обучающихся на соответствие их персональных достижений планируемым результатам обучения по дисциплине созданы фонды оценочных средств. Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в Приложении 1.

### **8. Учебно-методическое и информационное обеспечение дисциплины (модуля)**

#### **8.1 Основная литература**

1. Васильев, Алексей Николаевич. Java. Объектно-ориентированное программирование : учеб. пособие для магистров и бакалавров / А. Н. Васильев - СПб. : Питер , 2014 - 400 с.
2. Лафоре, Роберт. Структуры данных и алгоритмы JAVA / Р. Лафоре ; [пер. с англ. Е. Матвеева] - СПб. : Питер , 2015 - 704 с. : ил.
3. Эккель, Брюс. Философия JAVA / Б. Эккель - 4-е изд. - СПб. : Питер , 2017 - 1168 с. : ил.

#### **8.2 Дополнительная литература**

1. Беляев С.А. Разработка игр на языке JavaScript : Учебные пособия [Электронный ресурс] : Издательство "Лань" , 2018 - 128 - Режим доступа: <https://e.lanbook.com/book/102209>

### **8.3 Ресурсы информационно-телекоммуникационной сети "Интернет", включая профессиональные базы данных и информационно-справочные системы (при необходимости):**

1. IBM developerWorks Россия: Технология Java. Режим доступа: <http://www.ibm.com/developerworks/ru/java/>
2. Java™ Platform, Standard Edition 7 API Specification. Режим доступа: <http://docs.oracle.com/javase/7/docs/api/index.html>
3. The Java Tutorials. Режим доступа: <http://docs.oracle.com/javase/tutorial/index.html>
4. Баженова И.Ю. Язык программирования Java. – М.: Диалог-МИФИ, 2008. – 254 с. Режим доступа: [http://biblioclub.ru/index.php?page=book\\_view\\_red&book\\_id=54745](http://biblioclub.ru/index.php?page=book_view_red&book_id=54745)
5. Васюткина И. А. Технология разработки объектно-ориентированных программ на JAVA : учеб.- метод. пособие / И.А. Васюткина. – Новосибирск : Изд-во НГТУ, 2012. – 152 с. Режим доступа: <http://znanium.com/go.php?id=557111>
6. Электронно-библиотечная система издательства "Лань" - Режим доступа: <https://e.lanbook.com/>
7. Open Academic Journals Index (OAJI). Профессиональная база данных - Режим доступа: <http://oaji.net/>
8. Президентская библиотека им. Б.Н.Ельцина (база данных различных профессиональных областей) - Режим доступа: <https://www.prilib.ru/>
9. Информационно-справочная система "Консультант Плюс" - Режим доступа: <http://www.consultant.ru/>

## **9. Материально-техническое обеспечение дисциплины (модуля) и перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю) (при необходимости)**

### Основное оборудование:

- Компьютеры
- Проектор
- Экран Projecta 160\*160

### Программное обеспечение:

- Microsoft Windows 7 Russian

## **10. Словарь основных терминов**

### **100% Pure Java™**

Под термином "100% Pure Java™" понимают Java-программу, которая опирается только на специфицированное поведение Java-платформы, не использует платформно-зависимых (native) методов и не зависит ни от каких программных интерфейсов, кроме Java core API. Программа сертификации служит для того, чтобы определить, является ли данное приложение или апплет 100% Pure Java или нет.

### **Abstract Window Toolkit - AWT**

Стандартный пакет графических компонент, реализованных с использованием специфических платформенных методов. Данные компоненты поддерживают лишь то подмножество функциональных возможностей, которое присуще всем платформам.

### **abstract**

Ключевое слово языка программирования Java, используемое в определении класса для указания на невозможность создания его экземпляров, но при этом доступного для

наследования другими классами. Абстрактный класс может содержать нереализованные (абстрактные) методы, которые должны быть реализованы в его подклассах.

абстрактный класс (abstract class)

Класс, который содержит один или более абстрактных методов, вследствие чего нельзя создавать экземпляры данного класса. Абстрактные классы определены таким образом, чтобы другие классы могли расширять и конкретизировать их, реализуя абстрактные методы.

абстрактный метод (abstract method)

Метод, не имеющий реализации.

контроль доступа (access control)

Технические средства, посредством которых ограничиваются множества пользователей или программ, взаимодействующих с ресурсами, с целью повышения целостности, конфиденциальности и доступности.

АЦИН (ACID - Atomicity, Consistency, Isolation and Durability)

Акроним четырех свойств, обеспечиваемых транзакциями: атомарность, целостность, изоляция и надежность.

интерфейс прикладного программирования (API - Application Programming Interface)

Спецификация, предназначенная для пользователей и описывающая методы доступа к свойствам и состоянию объектов и классов.

апплет (applet)

Компонент, который обычно выполняется в Web-браузере или в любой другой программе просмотра апплетов.

аутентификация (authentication)

Процесс, посредством которого один объект показывает другому, что он действует от имени определенной идентификационной записи. Платформа J2EE нуждается в трех видах аутентификации: обычной (basic), связанной с формой (form-based) и взаимной (mutual).

bean-компонент (bean)

Программный компонент многократного использования. Bean-компоненты могут быть объединены для создания приложения.

методы обратной связи (callback methods)

Метод компонента, вызываемый контейнером для уведомления компонента о важных событиях во время его жизненного цикла.

класс (class)

Тип в языке программирования Java™, определяющий реализацию особого вида объекта. Описание класса определяет экземпляр класса, его переменные и методы. Так же определяются интерфейсы и суперклассы. По умолчанию суперклассом любого класса является Object.

метод класса (типа) (class method)

Метод, который вызывается безотносительно ссылки на конкретный объект. Методы класса влияют на класс в целом, а не на конкретный его экземпляр.

компонент (component)

Программный модуль, поддерживаемый контейнером. Компоненты конфигурируемы на стадии разработки. Платформа J2EE определяет четыре вида компонент: корпоративные компоненты (enterprise beans), Web-компоненты, апплеты и клиентские приложения.

const

Зарезервированное ключевое слово языка Java™. Однако, не используется текущими версиями Java.

контейнер (container)

Сущность, обеспечивающая управление, безопасность, разработку и сервисы выполнения компонент. Кроме того, каждый тип контейнера (EJB, Web, JSP, сервлет, апплет или приложение-клиент) также предоставляет свои специфические сервисы.

персистенция (сохраняемость), управляемая контейнером (container-managed persistence)

Механизм, при котором передача данных между экземпляром корпоративной компоненты и менеджером расположенных ниже ресурсов управляется контейнером корпоративной компоненты (enterprise bean).

транзакция, управляемая контейнером (container-managed transaction)

Транзакция, границы которой определяются EJB-контейнером. Экземпляр корпоративной компоненты (enterprise bean) должен использовать транзакции, управляемые контейнерами.

Common Object Request Broker Architecture - CORBA

Технология построения распределенных объектных приложений, специфицируемая группой по развитию стандартов объектного программирования (Object Management Group - OMG).

базовый класс (core class)

Публичный класс (или интерфейс), являющийся стандартным членом Java™ Platform. Обязательным свойством таких классов является их доступность в любой операционной системе, поддерживающей среду Java. Программой, "полностью" написанной на языке Java, называется программа, использующая только такие классы, и, следовательно, обладающая свойством независимости от платформы.

базовые пакеты (core packages)

Необходимый набор программных интерфейсов приложений (Application Programming Interfaces - APIs) платформы Java, который поддерживается в любой реализации.

делегирование (delegation)

Передача (делегирование) функций - способность объекта или потока внутри объекта выполнять под именем клиента запросы к другим удаленным объектам.

инкапсуляция (encapsulation)

Локализация (упрячивание) части данных в пределах класса. Поскольку объекты инкапсулируют данные и реализацию, пользователь может рассматривать объект как черный ящик, предоставляющий услуги. Переменные и методы экземпляров класса могут добавляться, удаляться или изменяться, но до тех пор, пока услуги, предоставляемые объектом, не изменяются, нет необходимости переписывать код, использующий данный объект. См. также переменная экземпляра класса и метод экземпляра класса.

Enterprise JavaBeans™ - EJB

Архитектура, предназначенная для развития и установки объектно-ориентированных, распределенных, корпоративных приложений. Приложения, написанные с использованием архитектуры Enterprise JavaBeans™, являются масштабируемыми, многопользовательскими и безопасными.

Браузер HotJava™ (HotJava(™) Browser)

Легко настраиваемый Web-браузер, разработанный компанией Sun Microsystems и написанный на языке программирования Java™.

interface

Ключевое слово языка программирования Java™, используемое для определения набора методов и постоянных значений (класса специального вида). Интерфейс в дальнейшем может реализовываться классами, которые определяют этот интерфейс с ключевым словом implements.

Java™

Торговая марка компании Sun для ряда технологий по созданию и безопасной работе программного обеспечения как в автономных, так и в сетевых средах.

комплект разработчика для Java (Java Development Kit - JDK™)

Среда программирования для написания апплетов и приложений в языке программирования Java.

платформа Java™ (Java™ Platform)

Состоит из языка Java, предназначенного для написания программ, набора интерфейсов прикладного программирования (APIs), библиотек классов, других программ,

используемых при разработке и компиляции, программ проверки ошибок, а также виртуальной машины Java, которая загружает и выполняет файлы классов.

Кроме того, платформа Java подчиняется набору требований совместимости для гарантии непротиворечивых и совместимых реализаций. Реализации, выполняющие требования совместимости, могут приобретать квалификацию заданной марки совместимости фирмы Sun. Java 2 - настоящее поколение платформы Java.

сервлет (servlet)

Java программа, которая расширяет функциональные возможности Web-сервера, динамически генерируя содержание и взаимодействуя с Web-клиентами при помощи принципа запрос-ответ.

контейнер сервлета (servlet container)

Контейнер, обеспечивающий сетевые службы, при помощи которых посылаются запросы и ответы, декодируются запросы и форматируются ответы. Все контейнеры сервлетов должны поддерживать HTTP-протокол, но могут также поддерживать дополнительные протоколы, например, HTTPS.

распределенный контейнер сервлета (servlet container, distributed)

Контейнер сервлета, запускающий Web-приложения, которые помечены как распределенные и выполняются на нескольких виртуальных машинах Java. При этом виртуальные машины могут быть запущены, как на одном, так и на разных компьютерах.

контекст сервлета (servlet context)

Объект, содержащий представление (вид) Web-приложения, в котором запущен сервлет. Используя контекст, сервлет может вести журнал событий, получать URL-ссылки на ресурсы, а также устанавливать и хранить атрибуты, которые могут использоваться другими сервлетами в приложении.

отображение сервлета (servlet mapping)

Определяет связь между структурой URL и сервлетом. Используется для отображения запросов в сервлеты. Если контейнер, обрабатывающий запрос, является JSP-контейнером, то неявно отображается URL, содержащий расширение .jsp.

сессия (session)

Объект, используемый сервлетом, для прослеживания взаимодействий пользователя с Web-приложением при помощи множества HTTP-запросов.

суперкласс (superclass)

Класс, из которого "произведены" другие классы. См. также подкласс, подтип.

super

Ключевое слово языка программирования JavaTM, используемое для доступа к членам класса, наследуемого классом, из которого производится вызов.

супертип (supertype)

Все интерфейсы или классы, расширяемые или реализуемые данным типом. См. также подтип, суперкласс.

Swing

Кодовое название совокупности графических компонентов, которые выполняются на любой платформе, поддерживающей виртуальную машину JavaTM. Данные компоненты могут обеспечивать большие функциональные возможности, вследствие того, что они целиком написаны на языке JavaTM.

synchronized

Ключевое слово языка программирования JavaTM, которое при применении к методу или блоку кода, гарантирует, что данный код будет выполняться не более чем одним потоком одновременно.