

ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАНИЕ

Java для IT-профессий

Учебник

ИЗДАТЕЛЬСТВО
FOLIANT

Нур-Султан-2019

УДК 004
ББК 32.816
X 29

Автор:
Харди Дирк

Редактор оригинального издания:
Барт Александер

Рецензенты
Шангытбаева Г.А. – *PhD доктор, Актюбинский региональный государственный университет имени К. Жубанова*
Абдрахманова С.В. – *преподаватель спецдисциплин первой категории КГУ «Карагандинский профессионально-технический колледж»*
Файзрахманова Р.Р. – *преподаватель-практик КГУ «Карагандинский профессионально-технический колледж»*

X 29 **Java для IT-профессий:** Учебник / Пер. с немецкого. – Нур-Султан: Фолиант, 2019. – 280 с.

ISBN 978-601-338-401-6

«Java для IT-профессий» – учебник по одному из самых популярных в мире языков программирования. Благодаря разделению на три части – теоретическую, практическую и проектную – обучающиеся могут не только изучить основы, но и применить знания на практике.

Первая часть книги – теоретическая. Здесь изложены принципы функционирования структур Java, концепция объектно-ориентированного программирования, работа с БД, разработка приложений Android. Рассматривается GUI-программирование с AWT, GUI-конструктор NetBeans, принципы создания апплетов.

Вторая часть книги содержит практические задания разных уровней сложности. Большое количество примеров настоящего кода позволяет освоить на практике теоретический блок.

В третьей части представлены учебные ситуации, прочно связанные с изученными темами. В данном разделе автором предложена не только разработка кода, но и контроль выполнения этой работы.

Учебник предназначен для студентов учебных заведений технического и профессионального образования, обучающихся по специальности 1300000 «Связь, телекоммуникация и информационные технологии», а также для начинающих и более опытных программистов.

УДК 004
ББК 32.816

Предисловие

Программная платформа Java была разработана в начале 90-х годов, чтобы получить отдельную систему из современного языка программирования и системного окружения. Для этого было необходимо кроссплатформенное программирование, так как на любой платформе (даже у кофе-машины) требовалось лишь наличие системного окружения.

Прорыв технологии Java произошел в связи с развитием интернета в конце 90-х. Благодаря Java и соответствующим техникам (таким как апплеты) веб-программирование сильно продвинулось вперед. Сегодня программы Java используются во всех сферах: не только в веб-программировании, но и в качестве приложений для ПК, в мобильных вычислениях (*Mobile Computing*) или в качестве встраиваемых систем (*Embedded Systems*).

Поэтому работа с Java включает в себя не только изучение объектно-ориентированного языка программирования, но и углубленный разбор программной платформы Java – это очень важный аспект для обучающихся IT-профессиям.

Структура книги

Цель данной книги – представление языка **Java** максимально наглядно, с ориентацией на практику и обучение. Поэтому книга написана с практическим подходом. Автор считает, что именно в школьном образовании необходимо усиленно подготавливать учеников к сложным темам программирования путем наглядного и практического обучения. После этого общие аспекты программирования или даже разработки программного обеспечения можно лучше понимать и применять на практике.

Книга состоит из трех частей.

Первая часть книги представляет собой информационный блок и предлагает системное введение в язык **Java** и основы программной платформы **Java**. Введение в GUI-программирование, соединение с базами данных и разработка приложений **Android** (новая часть во 2-м издании) завершают данный информационный блок.

Вторая часть книги – это сборник практических заданий. После получения соответствующих навыков из информационного блока задания из этой части могут подготовить к дальнейшему разбору тем, а разные уровни сложности заданий позволят провести обширную работу на занятиях.

В третьей части книги содержатся ситуации, основанные на учебном модуле «Разработка и развертывание систем приложений» из базисного учебного плана для IT-профессий (специальная разработка приложений специалистами по программированию). Ситуации составлены по учебным модулям и в идеале должны представлять комплексные действия (планирование, проведение, контроль). По этой причине учебные ситуации составлены таким образом, что наряду с фазой планирования в центре внимания стоит не только выполнение программы, но и соответствующие методы тестирования для контроля программы и процесса разработки. Учебные ситуации можно также рассматривать как идеи для проекта.

Книга подходит для всех профильных курсов подготовки в сфере IT. Благодаря дифференцированным проектным заданиям она может быть использована во всех IT-профессиях (особенно специалистами по информатике), а также информационно-техническими ассистентами.

В качестве средства разработки в данной книге используется **NetBeans IDE 8.0.1**. Эта среда разработки доступна для бесплатного скачивания в сети Интернет.

Мы будем благодарны за предложения и критику к данной книге (возможно по E-Mail).

Дирк Харди

Летом 2015 г.

E-Mail: Hardy@DirkHardy.de

Издательство «Europa-Lehrmittel»

E-Mail: Info@Europa-Lehrmittel.de

Содержание

Предисловие.....	3
Структура книги.....	3
Часть 1 Введение в Java	11
1 Введение в технологию Java	13
1.1 Технология Java	13
1.1.1 Возникновение технологии Java.....	13
1.1.2 Характеристики технологии Java	14
1.1.3 Компоненты технологии Java.....	14
1.1.4 Компиляция программ Java.....	14
1.2 Язык Java	15
1.2.1 Развитие языка Java.....	15
1.2.2 Характеристики языка Java	15
1.2.3 Ключевые слова в Java	16
1.2.4 Процедурное, структурированное и объектно-ориентированное программирование Java.....	16
1.2.5 Составляющие программы Java.....	17
2 Первая программа Java	19
2.1 Создание проекта Java	19
2.2 Первая программа Java	22
2.2.1 Основная структура Java.....	22
2.2.2 Пакеты	22
2.2.3 Класс «Hello, World!» и главный метод main.....	23
2.2.4 Вывод данных на экран.....	24
2.2.5 Важные правила программы Java	25
2.3 Основополагающие конвенции в Java	25
2.3.1 Идентификатор (имя) в Java.....	25
2.3.2 Разделительный знак	26
2.3.3 Комментарии в Java.....	27
2.4 Типы данных и переменные	29
2.4.1 Переменные в Java	29
2.4.2 Простые типы данных	29
2.4.3 Объявление переменных.....	30
2.4.4 Операции с простыми типами данных	31
2.4.5 Постоянные переменные.....	32
3 Ввод и вывод в Java	33
3.1 Система вывода в Java	33
3.1.1 Вывод переменных.....	33
3.2 Ввод с консоли	35
3.2.1 Ввод цепи символов.....	35
3.2.2 Конвертация ввода	35

4	Операторы в Java	38
4.1	Арифметические операторы	38
4.1.1	Простые типы данных и их арифметические операторы	38
4.1.2	Оператор модуля	39
4.1.3	Операторы инкремента и декремента	39
4.2	Реляционные и логические операторы	40
4.2.1	Реляционные операторы	40
4.2.2	Логические операторы	41
4.3	Битовые и другие операторы	42
4.3.1	Логические битовые операторы	42
4.3.2	Битовые операции сдвига	43
4.3.3	Преобразование типов оператором CAST	43
4.3.4	Присваивание и связанное присваивание	44
4.4	Ранг оператора	45
5	Селекция и итерация	47
5.1	Селекция	47
5.1.1	Представление выбора с помощью блок-схемы программы	47
5.1.2	Одиночный выбор с помощью оператора if	48
5.1.3	Двойной выбор с помощью оператора if-else	48
5.1.4	Ветвление с помощью операторов if и if-else	50
5.1.5	Множественный выбор с помощью оператора switch	51
5.2	Итерационные циклы с постусловием и предусловием, циклы со счетчиком	54
5.2.1	Цикл do-while	54
5.2.2	Цикл while	56
5.2.3	Цикл for	57
5.2.4	Досрочный выход и пропуск итерации	59
6	Понятие классов в Java	60
6.1	Первый класс в Java	62
6.1.1	Структура класса Java	62
6.1.2	Типы значений и ссылок	64
6.2	Методы в Java	64
6.2.1	Структура метода	64
6.2.2	Возвращаемое значение метода	66
6.2.3	Локальные переменные	67
6.2.4	Передаваемые параметры метода	68
6.2.5	Перегрузка методов	72
6.2.6	Дополнительная информация о методах	73
6.3	Другие элементы классов	74
6.3.1	Конструкторы и деструктор	74
6.3.2	Ссылка this	78
6.3.3	Статические элементы класса	79
6.3.4	Константные элементы класса	80
6.4	Перечисляемые типы	80
6.4.1	Простые перечисления	80
6.4.2	Классы перечислений	81
7	Наследование в Java	83
7.1	Наследование в Java	83
7.1.1	Простое наследование	83
7.1.2	Применение наследования в Java	84
7.1.3	Доступ к атрибутам	86
7.1.4	Финальные классы	87
7.2	Полиморфизм	87
7.2.1	Класс объект	87

7.2.2 Присваивание в рамках иерархии наследования	89
7.2.3 Перезапись методов.....	90
7.3 Абстрактные базовые классы.....	93
7.3.1 Абстрактный базовый класс	93
7.4 Интерфейсы в Java	94
7.4.1 Структура интерфейса	94
8 Массивы в Java.....	97
8.1 Одномерные и многомерные массивы	97
8.1.1 Одномерные массивы.....	97
8.1.2 Цикл for each	99
8.1.3 Многомерные массивы.....	101
8.1.4 Копирование массивов	104
8.1.5 Массивы объектов.....	106
8.1.6 Передача массивов методам.....	107
8.2 Сортировка массивов.....	109
8.2.1 Сортировка методом выбора	109
8.2.2 Статический метод сортировки sort.....	111
8.2.3 Интерфейс Comparable.....	112
8.3 Особые классы массивов.....	113
8.3.1 Класс ArrayList	113
8.3.2 Класс HashMap.....	114
9 Файловые операции в Java.....	117
9.1 Чтение и запись файлов.....	118
9.1.1 Последовательное чтение и запись	118
9.1.2 Прямой доступ к файлам	120
9.2 Чтение и запись текстовых файлов	122
9.2.1 Запись текстовых файлов с помощью PrintWriter	122
9.2.2 Чтение текстовых файлов с помощью сканера	123
9.3 Сериализация объектов.....	124
9.4 Методы класса File.....	127
9.4.1 Методы класса File.....	127
9.4.2 Ведение папок	128
10 Темы Java для продвинутого уровня.....	130
10.1 Исключения – Exceptions	130
10.1.1 Исключения и перехват ошибок (try and catch)	130
10.1.2 Системные исключения	132
10.1.3 Финальный блок	134
10.1.4 Вызов исключений	135
10.1.5 Создание собственных классов исключений	136
10.2 Обобщенное программирование	137
10.2.1 Обобщенные методы	137
10.2.2 Обобщенные классы	138
10.2.3 Использование обобщенных списковых классов	139
11 GUI-программирование с помощью AWT.....	141
11.1 GUI-программирование.....	141
11.1.1 История GUI-программирования	141
11.1.2 Структура AWT	142
11.1.3 Основные понятия GUI-программирования.....	142
11.2 Первая программа GUI	143
11.2.1 Использование класса Frame.....	143
11.2.2 Написание собственного класса Frame.....	144
11.3 Вывод текстовой и графической информации	145
11.3.1 Paint и первый вывод текста	145

11.3.2	Добавление клиентской части	147
11.3.3	Простой графический вывод	147
11.3.4	Многострочный вывод текста	150
11.4	Событийно-ориентированное программирование	152
11.4.1	Основа событийно-ориентированного программирования	152
11.4.2	Типы событий и приемники событий	153
12	Элементы управления с помощью AWT и классов Swing	157
12.1	Элементы управления с помощью AWT	157
12.1.1	Простые элементы управления	157
12.1.2	Использование элементов управления	157
12.1.3	Реагирование на события	158
12.1.4	Пример приложения с простыми элементами управления	159
12.1.5	Упорядочивание элементов управления с помощью менеджера Layout	162
12.2	Элементы управления с помощью классов Swing	164
12.2.1	Основы классов Swing	164
12.2.2	Элементы управления Swing	165
12.2.3	Использование простых элементов управления Swing	166
12.2.4	Look and Feel	169
12.2.4	Look and Feel	169
12.3	Более сложные элементы классов Swing	169
12.3.1	Дерево JTree	169
12.3.2	Создание узлов в JTree	170
12.3.3	Обзор важных методов JTree	172
12.3.4	Реагирование на события JTree	172
12.3.5	Создание таблицы с помощью JTable	174
12.3.6	Обзор важных методов JTable	175
12.3.7	Реагирование на события JTable	176
12.3.8	Снабжение элементов управления полосой прокрутки	177
13	Меню, диалоги и апплеты	180
13.1	Создание меню с помощью AWT	180
13.1.1	Создание меню	180
13.1.2	Реагирование на события меню	181
13.1.3	Создание контекстного меню	182
13.1.4	Создание меню с помощью классов Swing	183
13.2	Диалоги	185
13.2.1	Использование стандартных диалогов	185
13.2.2	Создание собственных диалогов	188
13.3	Создание апплетов	191
13.3.1	Основы апплетов	191
13.3.2	Класс апплетов	192
13.3.3	Запуск апплетов	193
13.3.4	Элементы управления в апплетах	193
13.3.5	Создание апплетов с помощью классов Swing	195
14	GUI-конструктор NetBeans	197
14.1	GUI-конструктор NetBeans	197
14.1.1	Создание формы JFrame	197
14.1.2	Добавление элементов управления	199
14.1.3	Реагирование на события	200
14.2	Интеграция более сложных элементов управления	202
14.2.1	Контейнер регистра (панель регистра)	202
14.2.2	Панель инструментов (Toolbar)	203
14.2.3	Контейнер прокрутки (Scrollpane)	203

15 Соединение с базой данных	204
15.1 Доступ к базе данных с помощью Java	204
15.1.1 Соединение с базой данных с помощью JDBC.....	204
15.1.2 Сброс операций	207
15.2 Доступ к другим базам данных	208
15.2.1 Добавление драйвера	208
15.2.2 Другие драйверы баз данных.....	209
16 Разработка приложений Android	210
16.1 Основы приложений Android	210
16.1.1 Установка Android SDK.....	211
16.1.2 Подготовка NetBeans для проектов Android.....	212
16.2 Разработка Android-Apps	213
16.2.1 Создание проекта Android	213
16.2.2 Элементы прикладной программы Android.....	215
16.2.3 Адаптация расположения прикладной программы	217
16.2.4 Другие контейнеры расположения и элементы управления.....	218
16.2.5 Программирование элементов управления	219
16.2.6 Запрос базы данных SQLite.....	221
Часть 2 Задания	225
Задания	226
1 Задания к главе «Введение в технологию Java»	226
2 Задания к главе «Первая программа Java»	226
3 Задания к главе «Ввод и вывод в Java».....	227
4 Задания к главе «Операторы в Java»	228
5 Задания к главе «Селекция и итерация»	230
6 Задания к главе «Понятие классов в Java».....	234
7 Задания к главе «Наследование в Java»	237
8 Задания к главе «Массивы в Java»	240
9 Задания к главе «Файловые операции в Java»	245
10 Задания к главе «Темы Java для продвинутого уровня»	251
11 Задания к главе «GUI-программирование с помощью AWT».....	253
12 Задания к главе «Элементы управления с помощью AWT или классов Swing»	256
13 Задания к главе «Меню, диалоги и апплеты»	258
14 Задания к главе «GUI-конструктор NetBeans»	259
15 Задания к главе «Соединение с базой данных»	261
16 Задания к главе «Разработка приложений Android».....	262
Часть 3 Учебные ситуации	265
Учебная ситуация 1: Презентация с вводной информацией о языке Java (на русском или английском)	266
Учебная ситуация 2: Подготовка клиентской документации для реализации среды разработки в Java (на русском или английском).....	267
Учебная ситуация 3: Разработка шифра для внутренней системы памяти отдела поддержки сетевой компании	270
Учебная ситуация 4: Планирование, внедрение и анализ электронной анкеты.....	270
Учебная ситуация 5: Разработка программного обеспечения для представления метеорологических данных с помощью схемы «Модель – Вид – Контроллер»	270
Учебная ситуация 6: Разработка приложения для решения головоломки «Судoku»	276
Приложение А: Структурированная техника документации	270
Блок-схема программы (PAP):	279
Пример блок-схемы программы:.....	280
Алфавитный указатель	281

Часть **1**

Введение в Java

1.1	Технология Java	13
1.2	Язык Java	15
2.1	Создание проекта Java	19
2.2	Первая программа Java	22
2.3	Основополагающие конвенции в Java	25
2.4	Типы данных и переменные	29
3.1	Вывод в Java	33
3.2	Ввод с консоли	35
4.1	Арифметические операторы	38
4.2	Реляционные и логические операторы	40
4.3	Битовые и другие операторы	42
4.4	Ранг оператора	45
5.1	Селекция	47
5.2	Итерационные циклы с постусловием и предусловием, циклы со счетчиком	54
6.1	Первый класс в Java	62
6.2	Методы в Java	64
6.3	Другие элементы классов	74
6.4	Перечисляемые типы	80
7.1	Наследование в Java	83
7.2	Полиморфизм	87
7.3	Абстрактные базовые классы	93
7.4	Интерфейсы в Java	94
8.1	Одномерные и многомерные массивы	97
8.2	Сортировка массивов	109
8.3	Особые классы массивов	113
9.1	Чтение и запись файлов	118
9.2	Чтение и запись текстовых файлов	122
9.3	Сериализация объектов	124
9.4	Методы класса File	127
10.1	Исключения – Exceptions	130
10.2	Обобщенное программирование	137
11.1	GUI-программирование	141
11.2	Первая программа GUI	143
11.3	Вывод текстовой и графической информации	145
11.4	Событийно-ориентированное программирование	152
12.1	Элементы управления с помощью AWT	157
12.2	Элементы управления с помощью классов Swing	164

13.1 Создание меню с помощью AWT	180
13.2 Диалоги	185
13.3 Создание апплетов	191
14.1 GUI-конструктор NetBeans	197
14.2 Интеграция более сложных элементов управления.....	202
15.1 Доступ к базе данных с помощью Java	204
15.2 Доступ к другим базам данных	208
16.1 Основы приложений Android.....	210
16.2 Разработка прикладных и программ Android	213

1. Введение в технологию Java

1.1 Технология Java

1.1.1 Возникновение технологии Java

В начале 90-х годов в компании *Sun Microsystems* был разработан язык программирования, который можно было бы использовать не только на ПК, но и на разных электронных устройствах (например, на портативных мини-компьютерах или даже в умных кофемашинах). Этот язык программирования должен был называться *OAK*. Однако это название было защищено, так что разработчикам пришлось придумать новое имя: **JAVA**¹.

Первая версия Java была представлена *Sun Microsystems* в 1995 г. Язык с возможностью доступа к Интернет – его можно было записать в определенном браузере (*HotJava*). Компания *Netscape* заключила договор с *Sun Microsystems* в 1996 г., и так Java получила стремительное распространение через известный браузер «*Netscape*» (*Netscape Navigator*).

Сейчас Java является мощным инструментом для разработки Интернет- и десктоп-приложений. Также она участвует в разработках *Mobile Computing*.

На следующем рисунке представлен обзор компонентов.



¹ Название Java связано с большой страстью разработчиков к кофе, так как Java во многих странах (включая США) является синонимом кофе.

1.1.2 Характеристики технологии Java

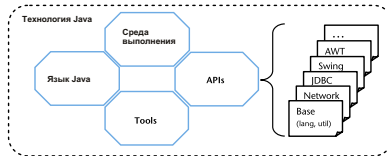
Большое преимущество Java – это кроссплатформенность. Исходный код Java переводится не на машинный код, а на один из видов промежуточного кода (байт-код). Потом этот байт-код может быть воспроизведен на всех платформах, имеющих соответствующую среду выполнения Java. Важнейшими характеристиками технологии Java являются:

- **Объектно-ориентированность:** Java – полностью объектно-ориентированный язык.
- **Кроссплатформенность:** Для большинства платформ была разработана среда выполнения Java, поэтому можно говорить об относительно высоком уровне кроссплатформенности. Существует, к примеру, среда выполнения у производственных систем Solaris (система «Unix»), Linux, Windows, также Mac OS X – тем самым, важнейшие производственные системы уже затронуты.
- **Надежность:** Java-программы работают под контролем в среде выполнения Java. Например, так называемый *garbage collector*, «сборщик мусора», обеспечивает надежное использование функции стирания памяти.
- **Современная разработка приложения:** С помощью Java можно программировать современные распределенные системы. Также доступ к базам данных поддерживается с помощью крупных библиотек.

1.1.3 Компоненты технологии Java

Java-технология состоит из различных компонентов, которые обеспечивают применение вышеописанных характеристик на практике. Наряду с собственным языком Java и средой выполнения к ним также относятся различные программные интерфейсы приложения APIs (*Application Programming Interfaces*). Все это объединяется в *Java Software Development Kit* (сокращенно *JDK*).

На следующем рисунке представлен обзор компонентов.



Большинство приложений может быть реализовано с помощью различных APIs. Отдельные APIs отвечают при этом за следующие сферы:

- **Base (lang и util):** Сборник классов для базового функционала: обработка строк, математические операции, форматированный вывод или обработка массивов.
- **Network:** С помощью этих классов можно реализовать сетевое программирование, например, через TCP-соединение и использование сокетов.
- **JDBC (Java Database Connectivity):** С помощью этих классов идет подключение к базам данных. Также они являются основой распределенных приложений.
- **Swing и AWT (Abstract Window Toolkit):** Эти классы предоставляют компоненты для разработки графических интерфейсов пользователя.

1.1.4 Компиляция программ Java

Исходный код Java переводится непосредственно не в исполняемый файл, а в один из видов промежуточных кодов (байт-код). Затем этот промежуточный код воспроизводится средой выполнения Java. При этом так называемая виртуальная машина *JVM* (*Java Virtual Machine*) переводит промежуточный код во внутренний, который является исполняемым кодом на соответствующей платформе. Актуальные виртуальные машины основаны на умных *Just-in-time*-компиляторах (*JIT*), таких как *HotSpot*, оптимизация которых позволяет очень быстро исполнять программы Java.