

Изучаем Java

Погружение в реальное
программирование

Кэти Сьерра
Берт Бэйтс
Триша Джи

3-е издание

УДК 004.4
ББК 32.973.26-018.2
И39

Kathy Sierra, Bert Bates, and Trisha Gee
Head First Java, 3rd Edition

© 2026 “Astana International Publishing” LLP Authorized Russian translation of the English edition
of Head First Java, 3rd edition ISBN 9781491910771

© 2022 Kathy Sierra, Bert Bates, and O’Reilly Media, Inc.

This translation is published and sold by permission of O’Reilly Media, Inc., which owns or controls all rights
to publish and sell the same.

Сьерра, Кэти.

И39 Изучаем Java. 3-е издание / Сьерра Кэти, Бэйтс Берт, Джи Триша : [перевод с ан-
глийского М. Райтман]. — Алматы : Астана иностранная пресса, 2026. — 752 с. —
(O’Reilly. Книги по программированию).

ISBN 978-601-12-3826-7

«Изучаем Java» — это не просто книга. Она не только научит вас теории языка Java и объектно-ориентированного программирования, она сделает вас программистом. В ее основу положен уникальный метод обучения на практике. В отличие от классических учебников информация дается не в текстовом, а в визуальном представлении. Вы освоите в деле все самое нужное: синтаксис и концепции языка, работу с потоками, работу в сети, распределенное программирование. Вся теория закрепляется интереснейшими примерами и тестами.

УДК 004.4
ББК 32.973.26-018.2

ISBN 978-601-12-3826-7

© Райтман М. А., перевод на русский язык, 2026
© Издание на русском языке, оформление.
ТОО «Издательство «Астана иностранная пресса», 2026

Все права защищены. Любое воспроизведение, полное или частичное, в том числе на интернет-ресурсах, а также запись в электронной форме для частного или публичного использования возможны только с разрешения правообладателя.

От Кэти и Берта:

Нашему мозгу — за то, что всегда с нами!

(несмотря на шаткие доказательства)

От Триши:

Исре — за то, что всегда рядом

(при избытке доказательств)

Вот бы существовала книга по
Java, более увлекательная, чем
ожидание в очереди на продление
водительских прав! Это что-то из
области фантастики...

Отзывы к третьему изданию книги

«Какая увлекательная и необычная книга! Я много лет преподаю Java и могу честно сказать, что это самый захватывающий ресурс по обучению программированию, с которым я когда-либо сталкивалась. Он вдохновляет меня снова погрузиться в изучение Java».

— Энджи Джонс, Java Champion

«В этой книге самое понятное объяснение потоков данных и лямбда-функций в Java, которое я когда-либо читал, — без лишнего шума и пафоса! Функциональное программирование подается с юмором и отличным стилем. Мне снова захотелось изучать Java! Если бы все писали код так же понятно, как это объясняют авторы...»

— Эрик Норманд, преподаватель Clojure, автор книги Grokking Simplicity

«Как же я хотел, чтобы такая книга попала мне в руки, когда я только начинал изучать Java! Ее так весело читать, что порой забываешь, что перед тобой серьезный учебник. Третье издание — огромный шаг вперед. Здесь есть все, что нужно современному Java-разработчику, чтобы стать профи. А иллюстрации — просто шедевр! Я не раз смеялся вслух. Большое спасибо авторам-чемпионам Java: Кэти, Берту и Трише!»

— доктор Хайнц Кабутц, The Java Specialists' Newsletter, www.javaspecialists.eu

«Обожаю стиль преподавания Head First Java! Это техническая книга, но читается, как художественная литература — невозможно оторваться. Здесь оригинальные иллюстрации, меткие аналогии и забавные диалоги между разработчиком и компилятором. Но главное — авторы помогают читателю взглянуть на программирование по-новому, а это бесценно. Авторы — настоящие волшебники. Рекомендую каждому, кто хочет изучить Java или прокачать свои навыки!»

— Мала Гупта, Developer Advocate компании JetBrains, автор и Java Champion

«Новички в Java часто спрашивают меня, с какой книги лучше начинать. Я всегда отвечаю: с Head First Java! Первые издания от Кэти Сьерры и Берта Бейтса перевернули подход к обучению программированию, сделав его максимально ориентированным на ученика. Это была настоящая революция. А теперь с ними еще и Триша Джи — один из лучших Java-инженеров и преподавателей, мой личный гуру в сложных вопросах Java! Третье издание не просто вернуло меня в прошлое, оно снова научило меня новому, даже после 20 лет работы с языком!»

— Мартейн Вербург, «дьявольский разработчик», Java Champion, Principal Group Manager по Java в Microsoft

«Head First Java — это уже легенда, и сложно представить лучшего человека для ее обновления, чем Триша Джи. Я всегда знала, что она крутая, но теперь поняла, что она еще и невероятно смешная! Третье издание получилось одновременно авторитетным, актуальным и безумно увлекательным учебником. Если существует лучший способ изучить Java, я о нем не слышала!»

— Холли Камминс, старший инженер Quarkus в Red Hat

«Эта книга — просто фантастика! В ней целая куча фигурных скобок, остроумных шуток, объектов, метафор, синтаксиса, живого кода и искреннего уважения к читателю. Авторы сделали изучение Java серьезным, но при этом увлекательным и запоминающимся. Особенно мне понравились метакогнитивные подсказки, приглашения почувствовать себя компилятором, истории, яркие визуализации и общее ощущение, что программирование открыто и доступно каждому».

— Кевлин Хенни, соавтор книги «97 вещей, о которых должен знать каждый Java-программист»

«Как же мне хотелось бы узнать про эту книгу, когда я только начинала изучать Java! Если вы хотите освоить Java легко, с юмором и увлечением (кто бы мог подумать, что это вообще возможно?), особенно если у вас нет технического образования, это идеальный выбор. Никогда не думала, что буду смеяться над учебником по программированию. Книга написана блестяще, остроумно и увлекательно, а главное — очень полезно!»

— Грейс Янсен, Developer Advocate, IBM

«Когда начинаешь изучать Java, перед тобой море книг и курсов. Проблема лишь в том, что большинство из них перегружено техническими деталями, и ты постоянно думаешь: “Когда же все это закончится?” Head First Java идет совершенно другим путем и превращает учебу в увлекательное путешествие. Ребусы с собаками, загадки и игры с карандашом делают обучение не только интересным, но и помогают запомнить важные моменты. Жаль, что такой книги не было, когда я начинал!»

— Саймон Риттер, заместитель технического директора в Azul и Java Champion

«Эта книга всегда вызывает у меня приятные воспоминания. Я впервые читал ее еще в университете, а теперь очень рад видеть обновленное издание. Head First Java — простая и понятная книга, с примерами кода, которые легко усваиваются. Рекомендую всем Java-разработчикам!»

— Нельсон Джало, технический предприниматель, основатель платформы Amigoscode.com и канала Amigoscode на YouTube

«Head First Java — это книга, которую я дал своему сыну, когда он решил изучать Java. Почему именно ее? Я знал, что веселые иллюстрации захватят его внимание лучше, чем любая другая книга по Java. Изучение Java с этой книгой больше похоже на игру на детской площадке, чем на занятия в классе».

— Кевин Нильсон, инженер-программист Google и руководитель Silicon Valley Java User Group

«Я завидую тем программистам, которые сегодня начинают изучать Java: у них есть эта потрясающая книга. Когда я учил Java двадцать лет назад, все было совсем не так увлекательно. Но с этой книгой точно скучать не придется! Я еще не встречал учебника, где Java-компилятор сражается с виртуальной машиной. Это по-настоящему круто!»

— Тагир Валеев, Java Champion и Technical Lead IntelliJ IDEA в JetBrains

«Почти 20 лет назад я прочитал первое издание Head First Java, когда еще был начинающим разработчиком. И вот третье издание снова приятно удивляет! За это время многое изменилось, в том числе и способы подачи материала. Head First Java — отличный современный учебник, который помогает быстро усвоить ключевые концепции без лишних деталей и без упрощений, а также предлагает материалы для дальнейшего изучения. Особенно хороши разделы про потоки данных, многозадачность и NIO».

— Майкл Симонс, Java Champion, инженер Neo4j, автор немецкого справочника Spring Boot Buch

«Если вы хотите разрабатывать на Java, эта книга — именно то, что нужно. Head First Java предлагает простые и элегантные примеры, которые ясно показывают, как применять Java на практике. Отличный выбор для тех, кто начинает свой путь Java-разработчика».

— Саньхун Ли, Alibaba Cloud

«Head First Java — техническая книга, которая совсем не похожа на техническую книгу. Она веселая, непринужденная и наполнена жизненными аналогиями, которые помогают легко разобраться даже в самых сложных темах. Это идеальное знакомство с богатой и многообразной экосистемой Java».

— Абрахам Марин-Перес, главный консультант команды Cosota

«Если вам нравится легкость и юмор в обучении, то лучшей книги для изучения Java, чем третье издание Head First Java, я даже представить не могу. Практичная и при этом игривая, познавательная и увлекательная — это идеальный учебник для начинающих разработчиков, которые хотят сразу перейти от теории к практике».

— Марк Лой, тренер, автор книги *Smaller C*, соавтор книг *Learning Java* и *Java Swing*

Положительные отзывы о предыдущих изданиях, а также о других книгах, написанных Кэти и Бертом в соавторстве

«Это не обычный учебник, а нечто похожее на дружелюбный интерфейс: благодаря легкой и остроумной подаче авторов изучение Java становится увлекательным путешествием, полным ожидания “а что же дальше?”»

— Уоррен Койфель, *Software Development Magazine*

«Единственный способ понять, насколько учебник хорош — это проверить, как он учит. Head First Java справляется с этим блестяще. Поначалу книга казалась мне слишком несерьезной, но вскоре я осознал, что именно в таком стиле я глубоко и основательно усваиваю материал. Подход Head First Java действительно делает учебу проще».

— slashdot (отзыв пользователя honestpuck)

«Помимо живого и увлекательного стиля, который быстро превращает вас из новичка в настоящего Java-воина, Head First Java подробно раскрывает многие практические темы, которые другие книги обычно оставляют читателю «для самостоятельного изучения». Это умно, иронично, стильно — и, главное, полезно. Не каждый учебник сумеет в таком формате обучить сериализации объектов и сетевым протоколам».

— доктор Дэн Рассел, директор по исследованиям пользовательского опыта в IBM Almaden Research Center и преподаватель искусственного интеллекта в Стэнфорде

«Она быстрая, дерзкая и захватывающая. Осторожно: вы можете действительно чему-то научиться!»

— Кен Арнольд, бывший старший инженер Sun Microsystems, соавтор (с Джеймсом Гослингом, создателем Java) книги *The Java Programming Language*

«Технология Java сегодня повсюду. Если вы занимаетесь разработкой ПО и еще не изучили Java, пришло время начинать — и начинать именно с Head First!»

— Скотт Макнили, бывший председатель совета директоров, президент и CEO Sun Microsystems

«Head First Java — это как встреча Монти Пайтона и «банды четырех». Материал идеально разбит на головоломки, истории, вопросники и примеры. Благодаря им вы освоите Java гораздо быстрее, чем с любым другим учебником по программированию».

— Дуглас Поу, *Columbia Java Users Group*

«Прочитав Head First Java, вы снова почувствуете, что учеба может быть увлекательной. Если вы хотите изучить Java, но у вас нет технического бэкграунда, эта книга станет настоящей находкой. Она превращает освоение сложного языка в увлекательный и интересный процесс».

— Джудит Тейлор, Southeast Ohio Macromedia User Group

«Если хотите освоить Java, прекращайте поиски. Это первая техническая книга с графическим интерфейсом! Уникальный формат дает преимущества, которых вы не найдете ни в одной другой книге по Java. Приготовьтесь к захватывающему путешествию в мир Java».

— Нил Р. Бауман, капитан и CEO в компании Geek Cruises

«Я просто не мог оторваться от коротких историй, кода с выносками, интервью и головоломок».

— Майкл Юань, автор книги Enterprise J2ME

«Head First Java придает новый смысл фразе «Для этого есть O'Reilly». Я взял эту книгу, потому что о ней говорили как о революционной, и она полностью оправдала мои ожидания. Это весело, дерзко, актуально и интерактивно. Читая ее, вы словно учитесь и смеетесь вместе с друзьями. Если хотите по-настоящему понять Java — эта книга для вас».

— Эндрю Поллак, www.thenorth.com

«Эта книга просто потрясающая! Я в шоке».

— Флойд Джонс, старший технический писатель, BEA

«Я чувствую, что с меня будто сняли тяжесть тысячи других книг».

— Уорд Каннингем, изобретатель Wiki и основатель Hillside Group

«Я смеялся, плакал, и книга меня действительно зацепила».

— Дэн Стейнберг, главный редактор java.net

«Сначала я просто катался от смеха. Затем понял, что книга не только точна, но и великолепно объясняет сложные темы, такие как паттерны проектирования».

— доктор Тимоти А. Бадд, доцент кафедры информатики Орегонского государственного университета, автор книги C++ for Java Programmers

«Идеальный учебник для тех, кто не любит скучные лекции. Книга поможет вам понять, как решать задачи на практике, — без лишних сложностей и занудства».

— Трэвис Каланик, основатель Scour и Red Swoosh, член MIT TR100

Авторы книг из серии Head First

Кэти Сьерра

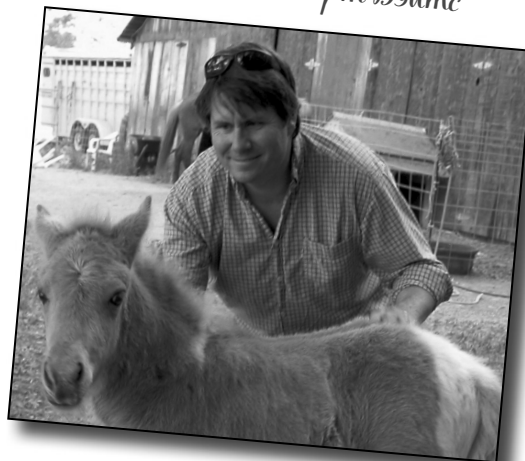


Кэти увлеклась теорией обучения еще во времена, когда разрабатывала игровые приложения для Virgin, MGM и Amblin. Позже она вела курс «Разработка новых подходов в сфере распространения информации» в Калифорнийском университете в Лос-Анджелесе, а также была главным тренером Java в Sun Microsystems. Кроме того, Кэти основала сайт JavaRanch.com (сейчас он известен как CodeRanch.com), который дважды получал награду Jolt Cola Productivity (в 2003 и 2004 годах).

В 2015 году Electronic Frontier Foundation вручила ей премию Pioneer Award за вклад в обучение пользователей и создание устойчивых сообществ.

В последнее время Кэти занимается изучением передовых технологий и науки о движении и приобретении навыков, известной как экологическая динамика (Eco-D). Ее работа по применению подхода Eco-D для тренировки лошадей открыла намного более гуманный метод верховой езды, который одних приводит в восторг, а других — в замешательство (а порой и в ужас!). Лошади, которых тренируют с помощью подхода Кэти, становятся более счастливыми, здоровыми и спортивными по сравнению со своими собратьями, обучаемыми традиционными способами.

Берт Бэйтс



До того, как заняться писательством, Берт работал разработчиком и специализировался на старых подходах к искусственному интеллекту (в основном экспертных системах), операционных системах реального времени и сложных системах планирования.

В 2003 году Берт вместе с Кэти написал Head First Java, с чего началась вся серия книг Head First. С тех пор он выпустил множество книг по Java, консультировал Sun Microsystems и Oracle по вопросам сертификации Java, а сейчас помогает тренерам, преподавателям, профессорам, авторам и редакторам создавать первоклассные учебные материалы для студентов.

Берт увлекается игрой в го. В 2016 году он с ужасом и восторгом одновременно наблюдал за тем, как искусственный интеллект AlphaGo разгромил знаменитого игрока Ли Седоля. В последнее время Берт активно применяет экологическую динамику (Eco-D), чтобы улучшить свою игру в гольф и даже научить кое-чему своего попугайчика по имени Боке.

По счастливой случайности Берт знает Тришу Джи больше восьми лет. Триша недавно присоединилась к авторам серии Head First, сделав книги серии еще более интересными и полезными.

Соавтор третьего издания этой книги



Триша Джи

Триша начала работать с Java еще в 1997 году, когда ее университет одним из первых внедрил этот «новенький» язык в программу преподавания информатики. С тех пор она была и разработчиком, и консультантом, создавая Java-приложения для самых разных сфер: от банковского дела и производства до некоммерческих организаций и высокоскоростной финансовой торговли.

Триша всегда любила делиться опытом, который накопила за годы работы, и поэтому решила стать разработчиком-адвокатом. Теперь она пишет блоги, выступает на конференциях и создает обучающие видео. Пять лет Триша проработала Java-разработчиком-адвокатом в JetBrains, а затем еще два года руководила там командой Java-адвоката. Благодаря этой деятельности она узнала о реальных проблемах, с которыми ежедневно сталкиваются Java-разработчики.

Триша и Берг долго обсуждали идею обновления Head First Java. Она с большой теплотой вспоминает их регулярные звонки: общение с таким опытным и душевным человеком, как Берг, всегда помогало ей сохранять спокойствие и оптимизм. Подход Берга и Кэти к обучению стал важным ориентиром и вдохновением в ее собственной работе последние десять лет.

Содержание (краткое)

Введение	23
1 Ныржаем: быстрый заплыв	37
2. Классы и объекты	63
3 Примитивы и ссылки	85
4 Повадки объектов в Java	107
5 Разработка программы	131
6 все что вы хотели знать о Java API	161
7 Наследование и полиморфизм	203
8 Интерфейсы и абстрактные классы	235
9 Конструкторы и сборка мусора	273
10 Числа и статические члены	311
11 Коллекции и обобщенные типы	345
12 Лямбда-выражения и потоки данных	405
13 Обработка исключений	457
14 Создание графического интерфейса	497
15 Библиотека Swing	545
16 Сериализация и ввод-вывод файлов	575
17 Сетевое взаимодействие и потоки выполнения	623
18. Состояние гонки и неизменяемые данные	675
Приложение А: Готовый код	709
Приложение Б: Одиннадцать тем, которые не попали в основную часть книги...	719

Содержание

Введение

Ваш мозг на Java. Вы решили изучить Java, а ваш мозг в этот момент пытается вам помочь и решает: «Это запоминать не стоит! Гораздо важнее помнить, какие животные могут вас съесть и можно ли кататься на сноуборде голым». Так как же убедить свой мозг, что от знания Java зависит ваша жизнь?

Как пользоваться книгой	23
Для кого эта книга?	24
Кому не подойдет эта книга?	24
Мы знаем, о чем вы думаете	25
И мы знаем, о чем думает ваш мозг	25
Метапознание: мышление о мышлении	27
Вруби мозг на всю катушку!	29
И не надейтесь, это еще не все!	30

1

Погружение в Java

Java открывает перед вами новый дивный мир. С момента появления первой версии 1.02 Java мгновенно захватила разработчиков своей простотой, объектно-ориентированными возможностями, автоматическим управлением памятью и — самое крутое — обещанием работать везде. Уже сейчас вы быстро напишете код, скомпилируете его и запустите. В этой главе вас ждут основы синтаксиса, циклы, условные операторы и ответ на вопрос «почему Java настолько крутая?». Погнали погружаться!



Прорываясь сквозь поверхность	37
Как работает Java	38
Что вы будете делать на Java	39
Кратчайшая история Java	40
Структура кода в Java	43
Структура класса	44
Написание класса с методом main()	45
Что поместить внутри основного метода?	48
Защелкивай, и защелкивай, и...	49
Простые логические проверки	49
Условное ветвление	51
Разработка серьезного бизнес-приложения	52
Одно буднее утро в доме Боба, где правит Java	53
Фразоплет 3000	55

2

Путеводитель по объектам

Говорят, здесь будут объекты. В первой главе весь код мы поместили в метод main(), но это еще не объектно-ориентированное программирование. Настало время выйти за рамки процедурных функций и создать свои собственные объекты! Мы разберемся, почему объектно-ориентированное программирование такое увлекательное, узнаем, в чем разница между классом и объектом, и как объекты могут сделать вашу жизнь лучше.

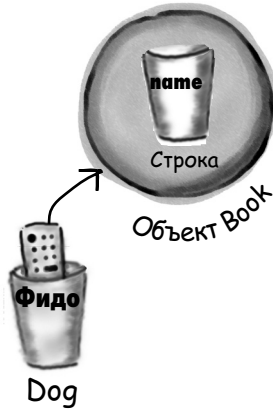


Путешествие в Объектино	63
Битва за кресло	64
Лора на мгновение опередила Брэда	66
Итак, Брэд, сторонник ООП, получил кресло и стол, да?	67
А как же метод rotate() у Амoeba?	68
Интрига меня уже в могилу загоняет!	69
Кто тут главный по мебели?	69
Проектируя класс, не забывайте о создаваемых объектах:	70
Различия между классом и объектом	71
Класс — это не объект	71
Создайте свой первый объект	72
Создание и тестирование объектов Movie	73
Быстрее! Валим из main!	74

3

Разбираемся с переменными

Не все сводится к числам и строкам. Переменные бывают примитивные и ссылочные. Достаточно ли для жизни только чисел, строк и массивов? А что делать, если у вас есть объект PetOwner, у которого живет Dog? Или автомобиль с объектом Engine? В этой главе поговорим о типах данных, выясним их отличия и узнаем, какие данные можно хранить в переменных. А еще выясним, каково жить на куче с автоматической сборкой мусора.



Знай свои переменные	85
Объявление переменной	86
Мне, пожалуйста, двойной эспрессо... Нет, лучше возьму целое	87
Хотите точно вместить значение в переменную?	88
Избегайте этого ключевого слова!	89
Управление объектом Dog	90
Жизнь в куче	93
Жизнь и смерть в куче	94
Массив как поднос со стаканчиками	95
Массивы — тоже объекты	95
Создание массива Dog	96
Управляйте объектом Dog	97
Что происходит, если ссылка на Dog находится в массиве Dog?	97
Пример класса Dog	98

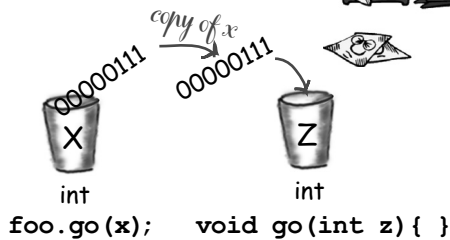
4

Как ведут себя объекты

Поведение зависит от состояния, а состояние — от поведения.

Мы уже знаем, что у объектов есть состояние (переменные экземпляра) и поведение (методы). Но теперь разберемся, как эти две вещи связаны. Поведение объекта зависит от его состояния: например, «если вес собаки меньше 14 фунтов, она лает, а если больше — рычит». Пришло время поменять пару вещей и посмотреть, что получится!

Передача по значению означает передачу через копирование



Повадки объектов в Java	107
Помните: класс описывает, что объект знает и делает	108
Громкость лая пропорциональна размеру	109
Вы можете передавать методу разные значения	110
Значения из метода можно возвращать	111
Методу можно передать сразу несколько значений.	112
В Java передача осуществляется по значению.	
Это значит, что передается копия	113
Применение параметров и возвращаемых значений	115
Инкапсуляция	116
Как себя ведут объекты в массиве	119
Объявление и инициализация переменных экземпляра	120
Разница между переменными экземпляра и локальными переменными	121
Сравниваем переменные (примитивы или ссылки)	122

Мы построили воронку для стартапа

5

Сверхмощные методы

A							
B	cabista						
C							
D			poniez				
E							
F							
G				hacqi			
	0	1	2	3	4	5	6

Прокачайте свои методы. Вы уже чуть-чуть поиграли с переменными, объектами и написали немного кода. Но для чего-то большего нужны инструменты посерьезнее: циклы, операторы и, может быть, случайные числа? А что, если строка вдруг захочет стать числом? Согласитесь, полезная фишка! В этой главе вы научитесь всему этому на примере настоящей программы, а заодно создадите игру типа «Битва за рынок» (что-то вроде «Морского боя»).

Особо мощные методы	131	Финальный код для SimpleStartup и SimpleStartupTestDrive	142
Создадим игру в стиле «Морского боя»: «Битва за рынок»	132	Полная версия игрового взаимодействия	143
Во-первых, высокоуровневое проектирование	133	Псевдокод для класса SimpleStartupGame	144
Плавное введение в игру «Битва за рынок. Простая версия».	134	Метод main() игры	146
Разработка класса	135	Методы random() и getUserInput()	147
Пишем реализацию метода	137	Последний класс: GameHelper	148
Разработка тестового кода для класса SimpleStartup	138	Сыграем?	149
Тестовый код для класса SimpleStartup	139	Что это? Ошибка?	149
Метод checkYourself()	140	Подробнее о циклах for	150
Объясняем новое	141	Обычные (не улучшенные) циклы for	150
		Прохождение цикла	151
		Улучшенный цикл for	152
		Приведение примитивов	153

6

Используем библиотеку Java

«Приятно знать, что в пакете java.util есть ArrayList. Но как бы я сама о этого додумалась?»

- Джулия, 31 год, модель



Зачем изобретать велосипед? Java идет в комплекте с сотнями готовых классов. Зачем создавать с нуля то, что уже сделали за вас? У вас есть дела поважнее.

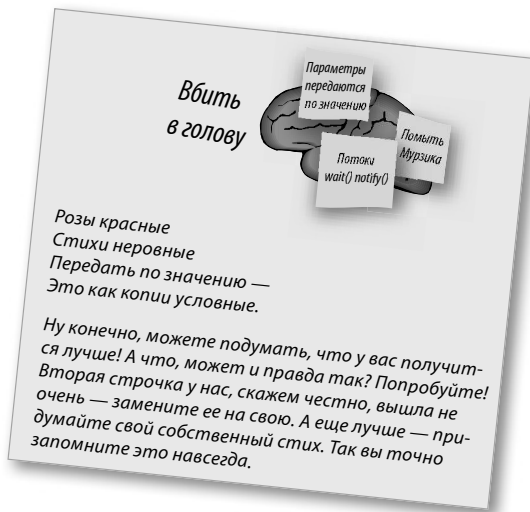
Ваша задача — писать только тот код, который уникален для вашего приложения.

Библиотека Java (Java API) — огромный склад классов, готовых к работе.

Использование библиотеки Java	161	Давайте фиксировать класс StartUp	174
В предыдущей главе мы оставили небольшую интригу, а именно — ошибку	162	Битва за рынок. Полная версия	176
Как это должно выглядеть	162	Что нужно изменить?	177
Где закралась ошибка	162	За что (и когда) отвечает каждый элемент в игре StartupBust	178
Что же случилось?	163	Псевдокод для настоящего класса StartupBust	180
Давайте пофиксим?	164	Финальная версия класса StartUp	186
Первый вариант слишком сложный	165	Супермощные логические выражения	187
Второй вариант лучше, но пока не идеален	165	Продолжение класса GameHelper...	189
Очнитесь и почувствуйте дух библиотеки	168	Использование библиотеки (Java API)	190
Что можно сделать с помощью ArrayList	169	Необходимо знать полное имя* класса, который вы хотите использовать в своем коде.	191
Сравнение ArrayList с обычным массивом	173	Как работать с API	194
		Чтение документации к классу	198

7 Жизнь в Объектино

Пишите программы с мыслями о будущем. Что, если бы ваш код можно было легко расширить другим разработчиком? А что, если бы код был гибким и легко адаптировался к изменениям? В этой главе мы разберем пять шагов к хорошему проектированию классов, три способа полиморфизма и восемь фишек для гибкости кода. А если поторопитесь, получите в подарок четыре совета по наследованию.

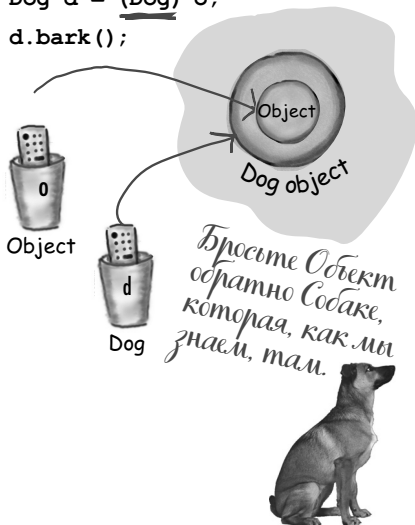


Прекрасная жизнь в Объектино	203	Проектирование иерархии наследования	214
И снова битва за кресло...	204	Отношения IS-A и HAS-A	215
А что насчет метода rotate() фигуры Амоеба?	205	Но погодите! Это еще не все!	216
Принципы наследования	206	Как узнать, что наследование оформлено правильно	217
Пример наследования:	207	Наследование при проектировании — хорошо или плохо?	219
Построим иерархию наследования на примере симулятора животных	208	В чем же настоящая ценность наследования?	220
Наследование для предотвращения дублирования кода в дочерних классах	209	Наследование гарантирует, что все классы-потомки содержат методы родительского типа*	221
Все ли животные питаются одинаково?	210	Заключение контракта: правила переопределения	228
Ищем новые возможности наследования	211	Перегрузка метода	229
Какой метод вызывается?	213		

8 Серьезный полиморфизм

Наследование — только начало. Если хотите стать гуру полиморфизма, придется разобраться с интерфейсами. Зачем? Они дают гибкость, недостижимую при обычном наследовании. Что такое интерфейс? Это полностью абстрактный класс, который невозможно создать напрямую. Зачем он тогда нужен? Читайте дальше и узнаете!

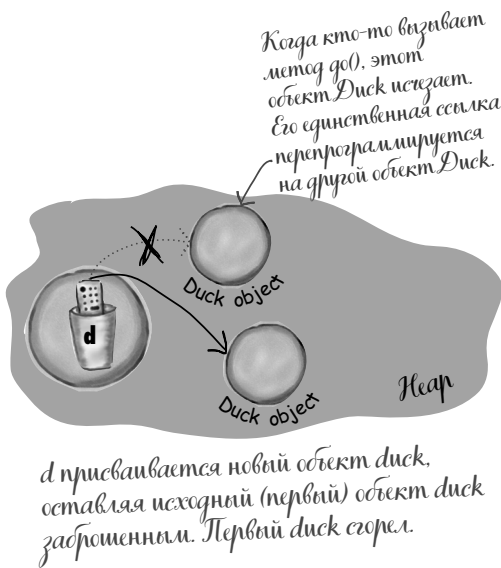
```
Object o = al.get(id);
Dog d = (Dog) o;
d.bark();
```



Серьезный полиморфизм	234	Использование полиморфических ссылок типа Object имеет свою цену...	249
Мы что-то забыли, когда разрабатывали это?	236	Когда собака перестает быть собакой...	250
А как выглядит объект new Animal()?	238	Объекты не лают	251
Компилятор не позволит вам создать экземпляр для абстрактного класса	239	Ощутите свой внутренний объект	252
Различия абстрактных и конкретных классов	240	Полиморфизм означает «много форм»	253
Абстрактные методы	241	Что делать, если нужно изменить контракт	256
Вы обязаны реализовывать все абстрактные методы	242	Варианты повторного использования существующих классов в программе для зоомагазина	257
Абстрактные и конкретные классы	243	Интерфейс спешит на помощь!	262
Полиморфизм в действии	244	Создание и реализация интерфейса Pet	263
Эх! Теперь нам нужно хранить еще и кошек	245	Классы из разных иерархий наследования могут реализовывать один и тот же интерфейс	264
А как быть с не-животными? Вот бы сделать универсальный класс, принимающий любые объекты?	246	Вызов версии метода из родительского класса	266
Что же спрятано внутри этого ультра-супер-мегакласса Object?	247		

9 Жизнь и смерть объекта

Объекты появляются и исчезают. Вы ответственны за их появление и исчезновение. Вы решаете, когда и как создавать объекты, а также когда наступает момент с ними «попрощаться». Память освободит сборщик мусора (gc), но как объекты попадают на кучу и когда их можно отпускать? Поговорим о куче, стеке, области видимости, конструкторах, суперконструкторах, ссылках на null и о том, когда объект становится кандидатом на сборку мусора.

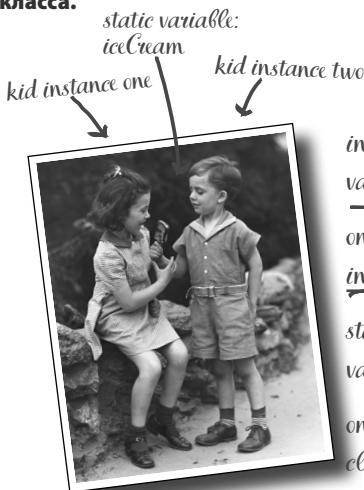


Жизнь и смерть объектов	272	Экспресс-обзор: четыре главных факта о конструкторах	287
Стек и куча: где все хранится	274	Погодите-ка... мы ведь так и не обсудили суперклассы, наследование и их связь с конструкторами!	288
Методы в стеке	275	Роль конструкторов суперкласса в жизни объекта	289
А как же быть с локальными переменными, представляющими собой объекты?	276	Создавая объект Nippo, вы наследуете свойства и методы классов Animal и Object	290
Если локальные переменные в стеке, то где переменные экземпляра?	277	Как вызвать конструктор суперкласса?	291
Чудо создания объекта	278	Могут ли потомки появиться раньше родителей?	292
Мы действительно вызываем метод Duck()? Потому что выглядит это именно так	279	Конструкторы суперкласса с аргументами	293
Конструируем объект duck	280	Вызов одного перегруженного конструктора из другого	294
Инициализация состояния нового объекта Duck	281	Мы выяснили, как появляется объект, а сколько он существует?	296
Используем конструктор, чтобы задать важное начальное состояние объекта Duck*	282	По поводу ссылочных переменных	298
Упрощаем процесс создания объекта Duck	283		
Всегда ли компилятор создает конструктор без аргументов? Нет!	284		
Наличие перегруженных конструкторов означает, что в вашем классе несколько конструкторов	285		

10 Важность чисел

Пора немного посчитать! В API Java есть множество удобных методов для работы с числами: абсолютные значения, округление, максимумы и минимумы. А как насчет форматирования? Иногда нужно вывести число с двумя знаками после запятой или расставить запятые в нужных местах. А даты? Или конвертация строки в число и наоборот? В этой главе мы начнем с изучения статических методов и переменных.

Статические переменные являются общими для всех экземпляров класса.



Без чисел никуда	311	Java упакует примитивы где угодно (ну, почти)	328
Методы Math: почти как глобальные	312	Постойте! Это еще не все!	
Разница между обычными и статическими методами	313	Обертки также содержат статические служебные методы!	330
Насчет классов со статическими методами	314	А теперь наоборот, превращаем простое числовое значение в строку	331
Статическим методам недоступны переменные экземпляра!	315	Форматирование чисел	332
Статические методы не могут обращаться к нестатическим методам!	316	Форматирование под микроскопом...	333
Статическая переменная: одно и то же значение для всех экземпляров класса	317	Знак процента (%) сообщает: «Вставьте аргумент сюда» (и отформатируйте его по указанной инструкции)	334
Инициализация статической переменной	319	В строке форматирования используется свой простой синтаксис	335
Статические финальные переменные = константы	320	Спецификатор форматирования	336
Модификатор final подходит не только для статических переменных...	321	Единственный обязательный спецификатор — это тип	337
Методы Math	324	Что случится, если использовать несколько аргументов?	338
Обертка для примитивов	326	Еще кое-что... статический импорт	339
Java сама упакует примитивы	327		