

---

# Предисловие

---

## К читателю

В конце 1995 года язык программирования Java вырвался на просторы Интернета и моментально завоевал популярность. Технология Java обещала стать *универсальным связующим звеном*, соединяющим пользователей с информацией, откуда бы эта информация ни поступала — от веб-серверов, из баз данных, поставщиков информации или любого другого источника, который только можно было себе вообразить. И действительно, у Java есть все, чтобы выполнить эти обещания. Это весьма основательно сконструированный язык, получивший широкое признание. Его встроенные средства защиты и безопасности обнадежили как программистов, так и пользователей программ на Java. Язык Java изначально обладал встроенной поддержкой для решения таких сложных задач, как сетевое программирование, взаимодействие с базами данных и многопоточная обработка. с

С 1995 года было выпущено девять главных версий комплекта Java Development Kit. За последние двадцать лет прикладной программный интерфейс (API) увеличился от 200 до 4 тысяч классов. Теперь прикладной программный интерфейс API охватывает самые разные предметные области, включая конструирование пользовательских интерфейсов, управление базами данных, интернационализацию, безопасность и обработку данных в формате XML.

Книга, которую вы держите в руках, является первым томом десятого издания. С выходом каждого издания ее главный автор старался как можно быстрее следовать очередному выпуску Java Development Kit, каждый раз переписывая ее, чтобы вы, читатель, могли воспользоваться преимуществами новейших средств Java. Настоящее издание обновлено с учетом новых языковых средств, появившихся в версии Java Standard Edition (SE) 8.

Как и все предыдущие издания этой книги, настоящее издание по-прежнему *адресуется серьезным программистам, которые хотели бы пользоваться Java для разработки настоящих проектов*. Автор этой книги представляет себе вас, дорогой читатель, как грамотного специалиста с солидным опытом программирования на других языках, кроме Java, и надеется, что вам не нравятся книги, которые полны игрушечных примеров вроде программ управления тостерами или животными в зоопарке либо “прыгающим текстом”. Ничего подобного вы не найдете в этой книге. Цель автора — помочь вам понять язык Java и его библиотеки в полной мере, а не создать иллюзию такого понимания.

В книге вы найдете массу примеров кода, демонстрирующих почти все обсуждаемые языковые и библиотечные средства. Эти примеры намеренно сделаны как можно более простыми, чтобы сосредоточиться на основных моментах. Тем не менее они в своем большинстве совсем не игрушечные и не “срезают острых углов”. Все они могут послужить вам неплохой отправной точкой для разработки собственного кода.

Автор книги предполагает, что вы стремитесь (и даже жаждете) узнать обо всех расширенных средствах, которые Java предоставляет в ваше распоряжение. Поэтому в первом томе настоящего издания подробно рассматриваются следующие темы.

- Объектно-ориентированное программирование.
- Рефлексия и прокси-классы.
- Интерфейсы и внутренние классы.
- Обработка исключений.
- Обобщенное программирование.
- Каркас коллекций.
- Модель приемников событий.
- Проектирование графического пользовательского интерфейса инструментальными средствами библиотеки Swing.
- Параллельное программирование.

В связи со стремительным ростом библиотеки классов Java одного тома оказалось недостаточно для описания всех языковых средств Java, о которых следует знать серьезным программистам. Поэтому книга была разделена на два тома. В первом томе, который вы держите в руках, главное внимание уделяется фундаментальным понятиям языка Java, а также основам программирования пользовательского интерфейса. А второй том посвящен средствам разработки приложений масштаба предприятия и усовершенствованному программированию пользовательских интерфейсов. В нем вы найдете подробное обсуждение следующих тем.

- Поточный прикладной программный интерфейс API.
- Обработка файлов и регулярные выражения.
- Базы данных.
- Обработка данных в формате XML.
- Аннотации.
- Интернационализация.
- Сетевое программирование.
- Расширенные компоненты графического пользовательского интерфейса.
- Усовершенствованная графика.
- Платформенно-ориентированные методы.

При написании книги ошибки и неточности неизбежны. И автору книги очень важно знать о них. Но он, конечно, предпочел бы узнать о каждой из них только один раз. Поэтому перечень часто задаваемых вопросов, исправлений, ошибок и обходных приемов был размещен по адресу <http://horstmann.com/corejava>, куда вы можете обращаться за справкой.

## Краткий обзор книги

В **главе 1** дается краткий обзор тех функциональных возможностей языка Java, которыми он отличается от других языков программирования. В ней сначала поясняется, что было задумано разработчиками Java и в какой мере им удалось воплотить

задуманное в жизнь. Затем приводится краткая история развития языка Java и показывается, как он стал тем, чем он есть в настоящее время.

В **главе 2** сначала поясняется, как загрузить и установить инструментарий JDK, а также примеры программ к этой книге. Затем рассматривается весь процесс компиляции и запуска трех типичных программ на Java (консольного приложения, графического приложения и апплета) только средствами JDK, текстового редактора, специально ориентированного на Java, а также интегрированной среды разработки на Java.

В **главе 3** начинается обсуждение языка программирования Java. В этой главе излагаются самые основы: переменные, циклы и простые функции. Если у вас имеется опыт программирования на C или C++, вам нетрудно будет усвоить материал этой главы, поскольку синтаксис этих языковых средств, по существу, ничем не отличается в Java. А если вам приходилось программировать на языках, не похожих на C, например на Visual Basic, прочитайте эту главу с особым вниманием.

Ныне объектно-ориентированное программирование (ООП) — господствующая методика программирования, и ей в полной мере отвечает язык Java. В **главе 4** представлено понятие *инкапсуляции* — первой из двух фундаментальных составляющих объектной ориентации, а также механизмы, реализующие ее в Java: классы и методы. В дополнение к правилам языка Java здесь также приводятся рекомендации по правильному объектно-ориентированному проектированию. И, наконец, в этой главе будет представлен замечательный инструмент **javadoc**, форматирующий комментарии из исходного кода в набор веб-страниц с перекрестными ссылками. Если у вас имеется опыт программирования на C++, можете лишь бегло просмотреть эту главу. А тем, кому раньше не приходилось программировать на объектно-ориентированных языках, придется потратить больше времени на усвоение принципов ООП, прежде чем изучать Java дальше.

Классы и инкапсуляция — это лишь часть методики ООП, и поэтому в **главе 5** представлен еще один ее краеугольный камень — *наследование*. Наследование позволяет взять существующий класс и модифицировать его в соответствии с конкретными потребностями программирующего. Это — основополагающий прием программирования на Java. Механизм наследования в Java очень похож на аналогичный механизм в C++. Опять же программирующие на C++ могут сосредоточить основное внимание лишь на языковых отличиях в реализации наследования.

В **главе 6** поясняется, как пользоваться в Java понятием *интерфейса*. Интерфейсы дают возможность выйти за пределы простого наследования, описанного в главе 5. Овладение интерфейсами позволит в полной мере воспользоваться объектно-ориентированным подходом к программированию на Java. После интерфейсов рассматриваются *лямбда-выражения* в качестве краткого способа выражения блока кода, который может быть выполнен впоследствии. И, наконец, в главе 6 рассматривается также удобное языковое средство Java, называемое *внутренними классами*.

**Глава 7** посвящена *обработке исключений* — надежному механизму Java, призванному учитывать тот факт, что непредвиденные ситуации могут возникать и в грамотно написанных программах. Исключения обеспечивают эффективный способ отделения кода нормальной обработки от кода обработки ошибок. Но даже после оснащения прикладной программы проверкой всех возможных исключительных ситуаций в ней все-таки может произойти неожиданный сбой. Во второй части этой главы будет представлено немало полезных советов по организации отладки программ. Кроме того, в главе 7 рассматривается весь процесс отладки на конкретном примере.

В **главе 8** дается краткий обзор *обобщенного программирования*. Обобщенное программирование делает прикладные программы легче читаемыми и более безопасными. В этой главе будет показано, как применяется строгая типизация, исключается потребность в неприглядном и небезопасном приведении типов и как преодолеваются трудности на пути совместимости с предыдущими версиями Java.

**Глава 9** посвящена каркасу коллекций на платформе Java. Всякий раз, когда требуется сначала собрать множество объектов, а в дальнейшем извлечь их, приходится обращаться к коллекции, которая наилучшим образом подходит для конкретных условий, вместо того чтобы сбрасывать их в обычный массив. В этой главе будут продемонстрированы те преимущества, которые дают стандартные, предварительно подготовленные коллекции.

С **главы 10** начинается программирование графических пользовательских интерфейсов. В этой главе будет показано, как создаются окна, как в них выполняется раскраска, рисуются геометрические фигуры, форматируется текст многими шрифтами и как изображения выводятся на экран.

В **главе 11** подробно обсуждается модель событий AWT — *абстрактного оконного инструментария*. В ней будет показано, как писать код, реагирующий на такие события, как щелчки кнопкой мыши или нажатия клавиш. Попутно вам предстоит ознакомиться с тем, как правильно обращаться с базовыми элементами графического пользовательского интерфейса вроде кнопок и панелей.

**Глава 12** посвящена более подробному обсуждению инструментальных средств Swing. Набор инструментов Swing позволяет строить межплатформенный графический пользовательский интерфейс. В этой главе вы ознакомитесь с различными видами кнопок, текстовых компонентов, рамок, ползунков, комбинированных списков, меню и диалоговых окон. Но знакомство с некоторыми из более совершенных компонентов Swing будет отложено до второго тома настоящего издания.

В **главе 13** будет показано, как разворачивать прикладные программы, будь то приложения или апплеты. В частности, в ней описывается, как пакетировать программы в файлы формата JAR и как доставлять приложения через Интернет с помощью технологии Java Web Start и механизмов апплетов. И, наконец, поясняется, каким образом программы на Java способны сохранять и извлекать информацию о конфигурации после своего разворачивания.

**Глава 14** завершает первый том настоящего издания обсуждением параллельного программирования, которое позволяет выполнять программируемые задачи параллельно. Это очень важное и любопытное применение технологии Java в эпоху многоядерных процессоров, которые нужно загрузить работой, чтобы они не простаивали.

В **приложении А** перечислены зарезервированные слова языка Java.

## Условные обозначения

Как это принято во многих книгах по программированию, моноширинным шрифтом выделяется исходный код примеров.



**НА ЗАМЕТКУ!** Этой пиктограммой выделяются примечания.



**СОВЕТ.** Этой пиктограммой выделяются советы.



**ВНИМАНИЕ!** Этой пиктограммой выделяются предупреждения о потенциальной опасности.



**НА ЗАМЕТКУ C++!** В этой книге имеется немало примечаний к синтаксису C++, где разъясняются отличия между языками Java и C++. Вы можете пропустить их, если у вас нет опыта программирования на C++ или же если вы склонны воспринимать этот опыт как страшный сон, который лучше забыть.

Язык Java сопровождается огромной библиотекой в виде прикладного программного интерфейса (API). При упоминании вызова какого-нибудь метода из прикладного программного интерфейса API в первый раз в конце соответствующего раздела приводится его краткое описание. Эти описания не слишком информативны, но, как мы надеемся, более содержательны, чем те, что представлены в официальной оперативной доступной документации на прикладной программный интерфейс API. Имена интерфейсов выделены **полужирным моноширинным** шрифтом, а число после имени класса, интерфейса или метода обозначает версию JDK, в которой данное средство было внедрено, как показано ниже.

**Название прикладного программного интерфейса 1.2**

Программы с доступным исходным кодом организованы в виде примеров, как показано ниже.

**Листинг 1.1.** Исходный код из файла `InputTest/InputTest.java`

## Примеры исходного кода

Все примеры исходного кода, приведенные в этой книге, доступны в архивированном виде на посвященном ей веб-сайте по адресу <http://horstmann.com/corejava>. Их можно извлечь из файла архива любой из распространенных программ разархивирования или с помощью утилиты **jar**, входящей в состав набора инструментов разработки Java Development Kit (JDK). Подробнее об установке JDK и примеров кода речь пойдет в главе 2.

## Благодарности

Написание книги всегда требует значительных усилий, а ее переписывание не намного легче, особенно если учесть постоянные изменения в технологии Java. Чтобы сделать книгу полезной, необходимы совместные усилия многих преданных делу людей, и автор книги с удовольствием выражает признательность всем, кто внес свой посильный вклад в настоящее издание книги.

Большое число сотрудников издательств Prentice Hall оказали неоценимую помощь, хотя и остались в тени. Я хотел бы выразить им свою признательность за их усилия. Как всегда, самой горячей благодарности заслуживает мой редактор из издательства Prentice Hall Грег Доенч (Greg Doench) — за сопровождение книги

на протяжении всего процесса ее написания и издания, а также за то, что он позволил мне пребывать в блаженном неведении относительно многих скрытых деталей этого процесса. Я благодарен Джули Нахил (Julie Nahil) за оказанную помощь к подготовке книги к изданию, а также Дмитрию и Алине Кирсановым — за литературное редактирование и набор рукописи книги. Приношу также свою благодарность моему соавтору по прежним изданиям Гари Корнеллу (Gary Cornell), который с тех пор обратился к другим занятиям.

Выражаю большую признательность многим читателям прежних изданий, которые сообщали о найденных ошибках и внесли массу ценных предложений по улучшению книги. Я особенно благодарен блестящему коллективу рецензентов, которые тщательно просмотрели рукопись книги, устранив в ней немало досадных ошибок.

Среди рецензентов этого и предыдущих изданий хотелось бы отметить Чака Аллисона (Chuck Allison) из университета долины Юты, Ланса Андерсона (Lance Anderson, Oracle), Пола Андерсона (Paul Anderson, Anderson Software Group), Алека Битона (IBM), Клиффа Берга, Эндрю Бинстока (Andrew Binstock, Oracle), Джошуа Блоха (Joshua Bloch), Дэвида Брауна (David Brown), Корки Картрайта (Corky Cartwright), Френка Коена (Frank Cohen, PushToTest), Криса Крейна (Chris Crane, devXsolution), доктора Николаса Дж. Де Лилло (Dr. Nicholas J. De Lillo) из Манхэттенского колледжа, Ракеша Дхупара (Rakesh Dhoopar, Oracle), Дэвида Гири (David Geary), Джима Гиша (Jim Gish, Oracle), Брайана Гетца (Brian Goetz, Oracle), Анжелу Гордон (Angela Gordon) и Дэна Гордона (Dan Gordon, Electric Cloud), Роба Гордона (Rob Gordon), Джона Грея (John Gray) из Хартфордского университета, Камерона Грегори (Cameron Gregory, olabs.com), Марти Холла (Marty Hall, coreservlets.com, Inc.), Винсента Харди (Vincent Hardy, Adobe Systems), Дэна Харки (Dan Harkey) из университета штата Калифорния в Сан-Хосе, Вильяма Хиггинса (William Higgins, IBM), Владимира Ивановича (Vladimir Ivanovic, PointBase), Джерри Джексона (Jerry Jackson, SA Technologies), Тима Киммета (Tim Kimmet, Walmart), Криса Лаффра (Chris Laffra), Чарли Лаи (Charlie Lai, Apple), Анжелику Лангер (Angelika Langer), Дуга Лэнгстона (Doug Langston), Ханг Лау (Hang Lau) из университета имени Макгилла, Марка Лоуренса (Mark Lawrence), Дуга Ли (Doug Lea, SUNY Oswego), Грегори Лонгшора (Gregory Longshore), Боба Линча (Bob Lynch, Lynch Associates), Филиппа Милна (Philip Milne, консультанта), Марка Моррисси (Mark Morrissey) из научно-исследовательского института штата Орегон, Махеш Нилаканта (Mahesh Neelakanta) из Атлантического университета штата Флорида, Хао Фам (Hao Pham), Пола Филона (Paul Pillion), Блейка Рэгсдейла (Blake Ragsdell), Стюарта Реджеса (Stuart Reges) из университета штата Аризона, Рича Розена (Rich Rosen, Interactive Data Corporation), Питера Сандерса (Peter Sanders) из университета ЭССИ (ESSI), г. Ницца, Франция, доктора Пола Сангеру (Dr. Paul Sanghera) из университета штата Калифорния в Сан-Хосе и колледжа имени Брукса, Пола Сэвинка (Paul Sevinc, Teamup AG), Деванг Ша (Devang Shah, Oracle), Бредли А. Смита (Bradley A. Smith), Стивена Стелтинга (Steven Stelting, Oracle), Кристофера Тэйлора (Christopher Taylor), Люка Тэйлора (Luke Taylor, Valtech), Джорджа Тхируватукала (George Thiruvathukal), Кима Топли (Kim Topley, StreamingEdge), Джанет Трауб (Janet Traub), Пола Тиму (Paul Tuma, консультанта), Питера Ван Дер Линдена (Peter van der Linden), Кристиана Улленбума (Christian Ullенboom), Берта Уолша (Burt Walsh), Дана Ксю (Dan Xu, Oracle) и Джона Завгрена (John Zavgren, Oracle).

*Кей Хорстманн, Биль/Бьен, Швейцария, ноябрь 2015 г.*

## От издательства

Вы, читатель этой книги, и есть главный ее критик и комментатор. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересно услышать и любые другие замечания, которые вам хотелось бы высказать в наш адрес.

Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам бумажное или электронное письмо либо просто посетить наш веб-сайт и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится или нет вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Посылая письмо или сообщение, не забудьте указать название книги и ее авторов, а также ваш обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию последующих книг.

Наши электронные адреса:

E-mail: [info@williamspublishing.com](mailto:info@williamspublishing.com)

WWW: <http://www.williamspublishing.com>

Наши почтовые адреса:

в России: 127055, г. Москва, ул. Лесная, д.43, стр. 1

в Украине: 03150, Киев, а/я 152