

# ПРЕДИСЛОВИЕ

**В** конце 1995 г. язык программирования Java вырвался на Internet-сцену и моментально завоевал популярность. Технология Java обещала стать *универсальным клеем*, соединяющим пользователей с информацией, откуда бы эта информация ни поступала — от Web-серверов, из баз данных, поставщиков информации, либо любого другого источника, который только можно было себе вообразить. И на самом деле, Java обладает всеми возможностями соответствовать этим обещаниям. Это исключительно основательно спроектированный язык, который получил признания всех основных участников рынка за исключением Microsoft. Его встроенные средства защиты и безопасности обнадежили как программистов, так и пользователей Java-программ. Java изначально обладала встроенной поддержкой таких сложных задач, как сетевое программирование, подключаемость к базам данных и многопоточность, обеспечивая простоту реализации всех этих вещей.

С 1995 г. Sun Microsystems выпустила семь главных версий Java Development Kit. За последние 11 лет интерфейс прикладного программирования (API) вырос с 200 до 3000 классов. Теперь API охватывает такие различные области, как конструирование пользовательских интерфейсов, управление базами данных, интернационализация, безопасность и обработка XML.

Книга, которую вы держите в руках — это первый том восьмого издания. С выходом каждого издания мы старались как можно быстрее следовать очередному выпуску Java Development Kit, каждый раз переписывая ее, чтобы вы могли воспользоваться преимуществами новейших средств Java. Настоящее издание отражает средства, имеющиеся в составе Java Standard Edition (SE) 6.

Как и все предыдущие издания этой книги, мы по-прежнему *ориентируем ее на аудиторию серьезных программистов*, которые хотят использовать Java для разработки реальных проектов. Мы воспринимаем вас, дорогой читатель, как программиста с солидным опытом в языках программирования, отличных от Java, и надеемся, что вы не любите книг, полных игрушечных примеров (вроде программ управления тостерами или животными в зоопарке либо “прыгающим текстом”). Ничего подобного в этой книге вы не найдете. Наша цель — достижение полного понимания языка Java и его библиотеки, а не создание иллюзии такого понимания.

В книге вы найдете массу примеров кода, демонстрирующих почти все обсуждаемые языковые и библиотечные средства. Эти примеры мы намеренно сделали насколько возможно простыми, чтобы сосредоточиться на основных моментах, но, тем не менее, в большинстве своем, они вовсе не игрушечные и не “срезают острых углов”. Все они могут стать хорошей отправной точкой для разработки вашего собственного кода.

Мы предполагаем, что вы хотите, а то и жаждете узнать обо всех расширенных средствах, которые предлагает Java вашему вниманию. Так, например, мы предлагаем детальное рассмотрение следующих тем.

- Объектно-ориентированное программирование.
- Рефлексия и прокси-классы.
- Интерфейсы и вложенные классы.

- Модель слушателей событий.
- Проектирование графического интерфейса пользователя с использованием инструментария Swing.
- Обработка исключений.
- Обобщенное программирование.
- Каркас коллекций.
- Параллелизм.

В связи со стремительным ростом библиотеки классов Java, одного тома оказалось недостаточно для описания всех средств Java, о которых следует знать серьезным программистам. Первый том, который вы держите в руках, сосредоточен на фундаментальных концепциях языка Java, наряду с основами программирования пользовательского интерфейса. Второй том посвящен средствам разработки приложений масштаба предприятия и усовершенствованному программированию пользовательских интерфейсов. В нем вы найдете детальное обсуждение следующих тем.

- Файлы и потоки.
- Распределенные объекты.
- Базы данных.
- Расширенные компоненты графического пользовательского интерфейса.
- “Родные” (native) методы.
- Обработка XML.
- Сетевое программирование.
- Расширенная графика.
- Интернационализация.
- JavaBeans.
- Аннотации.

В этом издании мы перегруппировали содержимое двух томов. В частности, многопоточность теперь рассматривается в первом томе, поскольку ее значение существенно возросло.

При написании книги ошибки и неточности неизбежны. Мы очень заинтересованы в том, чтобы знать о них. Но, конечно же, мы бы предпочли узнавать о каждой из них только однажды. Поэтому мы поместили список часто задаваемых вопросов, исправлений ошибок и “обходных путей” по адресу <http://horstmann.com/corejava>.

## Краткий тур по книге

**Глава 1** предоставляет обзор возможностей Java, которые отличают его от других языков программирования. Мы объясним, что было задумано проектировщиками языка, и в какой мере им удалось воплотить задуманное. Затем приведем краткую историю того, как развивался язык Java, и как он стал тем, чем он есть сегодня.

В **главе 2** мы расскажем о том, как загрузить и установить Java, а также примеры программ для книги. Затем мы проведем вас через компиляцию и запуск трех типичных программ Java — консольного приложения, графического приложения и апплета — все это с использованием только JDK, специального Java-ориентированного текстового редактора и Java IDE.

В **главе 3** начинается дискуссия о языке программирования Java. В этой главе мы раскроем основы: переменные, циклы и простые функции. Если вы — программист на C или C++, для вас все должно пройти гладко, поскольку синтаксис этих языковых средств по существу очень похож. Если вы обладаете опытом в языке, не похожем на C, таком как Visual Basic, вам следует прочесть эту главу особенно внимательно.

Объектно-ориентированное программирование (ООП) в наши дни — доминирующая программистская практика, и Java — в полной мере объектно-ориентированный язык. В **главе 4** представлена инкапсуляция — первый из двух фундаментальных строительных блоков объектной ориентации, а также механизмы Java, реализующие ее — классы и методы. В дополнение к правилам языка Java мы также приводим советы по правильному объектно-ориентированному дизайну. И, наконец, мы опишем изумительный инструмент `javadoc`, который форматирует комментарии из вашего кода в набор Web-страниц с перекрестными ссылками. Если вы знакомы с C++, то можете лишь бегло просмотреть эту главу. Программисты, пришедшие из мира не объектно-ориентированных языков, должны потратить больше времени для овладения концепциями ООП, прежде чем дальше изучать Java.

Классы и инкапсуляция — это лишь часть идеологии ООП, и потому **глава 5** представляет еще один ее краеугольный камень — *наследование*. Наследование позволяет взять существующий класс и модифицировать его в соответствии с вашими потребностями. Это — фундаментальная техника в программировании на Java. Механизм наследования в Java довольно похож на тот же механизм в C++. Опять-таки, программисты на C++ могут сосредоточиться лишь на отличиях между языками.

В **главе 6** будет показано, как использовать в Java понятие *интерфейса*. Интерфейсы позволяют вам выйти за пределы простого наследования, описанного в главе 5. Овладение интерфейсами позволит вам получить полный доступ ко всей мощи объектно-ориентированного подхода Java к программированию. Мы также раскроем удобное техническое средство Java, называемое *встроенными классами*. Встроенные классы помогают сделать ваш код янее и более согласованным.

С **главы 7** начинается серьезное прикладное программирование. Каждый программист Java должен знать хоть немного о программировании графических пользовательских интерфейсов, и здесь вы найдете его основы. Мы покажем, как создаются окна, как выполняется рисование в них, как рисуются геометрические фигуры, как форматируется текст с применением множества шрифтов и как отображаются графические изображения.

**Глава 8** содержит детальную дискуссию о модели событий AWT — *абстрактного оконного инструментария*. Вы увидите, как писать код, реагирующий на такие события, как щелчки кнопками мыши или нажатия клавиш. По ходу дела вы ознакомитесь с обработкой базовых элементов графического пользовательского интерфейса, таких как кнопки и панели.

**Глава 9** посвящена более детальному обсуждению инструментария Swing. Пакет инструментов Swing позволяет строить межплатформенный графический интерфейс пользователя. Вы познакомитесь с различными типами кнопок, текстовых компонентов, рамок, ползунков, окон списков, меню и диалоговых окон. Однако некоторые из наиболее совершенных компонентов мы отложим до второго тома.

В **главе 10** будет показано, как разворачивать ваши программы — будь то приложения или апплеты. Мы опишем, как пакетировать программы в файлы JAR, и как доставлять приложения через Internet с применением Java Web Start и механизмов апплетов. И, наконец, мы объясним, как программы Java могут сохранять и извлекать конфигурационную информацию после своего разворачивания.

**Глава 11** посвящена *обработке исключений* — устойчивому механизму Java, предназначенному противостоять тому факту, что плохие вещи случаются и с хорошими программами. Исключения обеспечивают эффективный способ отделения кода нормальной обработки от кода обработки ошибок. Конечно, даже после укрепления вашей программы проверкой всех возможных исключительных ситуаций, может произойти сбой в ожидаемой работе. Во второй половине этой главы мы предоставим огромное количество полезных советов по организации отладки. И, наконец, приведем пример полного сценария отладки.

В **главе 12** представлен обзор *обобщенного программирования* — главного преимущества Java SE 5.0. Обобщенное программирование делает ваши программы легче читаемыми и более безопасными. Мы покажем, как используется строгая типизация, и исключаются необходимость в некрасивых и небезопасных приведениях типа, а также как справиться со сложностями, происходящими из-за необходимости сохранения совместимости с предыдущими версиями Java.

Тема **главы 13** — каркас коллекций платформы Java. Всякий раз, когда вы хотите собирать множество объектов и извлекать их позднее, вам понадобится использовать коллекцию, которая наилучшим образом подходит к вашим конкретным условиям, вместо того, чтобы сбрасывать их в обычный массив. В этой главе будет продемонстрировано, какие преимущества дают стандартные коллекции, которые заранее подготовлены для вас.

**Глава 14** завершает книгу обсуждением многопоточности, которая позволяет программировать выполняемые параллельно задачи. (Под потоком понимается поток управления внутри программы.) Мы покажем, как создаются потоки и как осуществляется их синхронизация. Средства многопоточности существенно изменились после Java SE 5.0, и мы расскажем вам о новых ее механизмах.

В **приложении** вы найдете зарезервированные слова языка Java.

## Соглашения

Как это принято во многих компьютерных книгах, моноширинный шрифт используется для представления исходного кода.

**На заметку!**

Такой пиктограммой выделяются замечания.

---

**Совет.**

Такой пиктограммой выделяются советы.

---

**Внимание!**

Такой пиктограммой выделяются предупреждения об опасности.

---



**На заметку!** В нашей книге есть множество замечаний, касающихся C++, в которых объясняются отличия между языками Java и C++. Вы можете пропустить их, если не имеете опыта работы с C++ или если склонны воспринимать этот опыт как страшный сон, который лучше забыть.

---

API

## Интерфейс прикладного программирования

Язык Java сопровождается огромной библиотекой интерфейса прикладного программирования (API). Используя вызов API впервые, в конце соответствующего раздела мы добавляем краткое описание, сопровождаемое такой пиктограммой. Эти описания не слишком информативны, но, как мы надеемся, более информативны, чем те, что представлены в официальной онлайн-документации по API. Мы снабжаем каждую врезку по API номером версии, в которой средство впервые было представлено, чтобы помочь тем читателям, которые не используют самую последнюю версию Java.

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

### Листинг 1.1. Содержимое файла `WelcomeApplet.java`

---

## Коды примеров

Коды примеров, рассмотренных в книге, доступны как на Web-сайте, посвященном книге (<http://horstmann.com/corejava>), так и на Web-сайте издательства.

## 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 Web-сервер 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 координаты:

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

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

Информация для писем из:

России: 115419, Москва, а/я 783

Украины: 03150, Киев, а/я 152