



ОГЛАВЛЕНИЕ

Об авторе	10
О рецензентах	11
Предисловие	14
Вопросы, освещаемые в книге	14
Что нужно для чтения этой книги	16
Для кого эта книга	16
Соглашения	16
Отзывы и пожелания	17
Скачивание исходного кода примеров	17
Список опечаток	18
Нарушение авторских прав	18
Вопросы	18
Глава 1.	
Знакомство с NetBeans	19
Введение	19
Получение NetBeans	20
Установка NetBeans	23
Microsoft Windows	24
Mac OSx	24
Linux	24
Другие платформы	25
Процедура установки	25
Первый запуск NetBeans	31
Настройка NetBeans для разработки Java EE-приложений	32
Интегрирование NetBeans со сторонним сервером приложений	33
Интегрирование NetBeans с СУРБД стороннего производителя	36
Развертывание нашего первого приложения	40
Подсказки NetBeans для эффективной разработки	43
Автозавершение кода	43
Шаблоны кода	47
Клавиши быстрого вызова	49
Изучение визуальных индикаторов NetBeans	53
Функция ускорения разработки HTML5	54
Резюме	59

Глава 2.

Разработка веб-приложений с использованием JavaServer Faces 2.2	60
Введение в JavaServer Faces	60
Разработка нашего первого приложения JSF	61
Создание нового проекта JSF	61
Добавление в страницу возможности ввода данных.....	66
Создание именованного компонента CDI.....	73
Реализация страницы подтверждения	77
Запуск приложения.....	78
Проверка допустимости в JSF.....	80
Шаблоны фейслетов	83
Добавление шаблона фейслетов	84
Использование шаблона.....	86
Контракты библиотек ресурсов	90
Составные компоненты	96
Потоки Faces Flow	101
Поддержка HTML5.....	108
HTML5-подобная разметка	108
Сквозные атрибуты	111
Резюме	113

Глава 3.

Библиотеки компонентов JSF	114
Использование компонентов PrimeFaces в JSF-приложениях.....	114
Использование компонентов ICEfaces в JSF-приложениях	120
Использование компонентов RichFaces в JSF-приложениях.....	128
Резюме	133

Глава 4.

Взаимодействие с базами данных через Java Persistence API	135
Создание первой сущности JPA	136
Добавление сохраняемых полей в сущность	145
Создание объекта доступа к данным.....	147
Автоматическое создание сущностей JPA	153
Именованные запросы и JPQL	162
Проверка допустимости со стороны компонентов	164
Отношения сущностей	164
Создание приложений JSF из сущностей JPA	172
Резюме	179

Глава 5.**Реализация уровня бизнес-логики на сеансовых компонентах EJB 180**

Введение в сеансовые компоненты.....	181
Создание сеансового компонента в NetBeans	181
Доступ к компонентам из клиента	193
Запуск клиента	196
Управление транзакциями в сеансовых компонентах	197
Реализация аспектно-ориентированного программирования с помощью интерцепторов.....	199
Реализация класса интерцептора	200
Декорирование компонентов EJB аннотацией @Interceptors	202
Служба таймеров EJB.....	203
Автоматическое создание сеансовых компонентов из сущностей JPA	206
Резюме	211

Глава 6.**Контексты и внедрение зависимостей 213**

Введение в CDI.....	213
Квалификаторы.....	219
Стереотипы.....	225
Типы привязки интерцепторов	227
Собственные контексты	232
Резюме	234

Глава 7.**Обмен сообщениями с применением JMS
и компонентов, управляемых сообщениями..... 236**

Введение в JMS	236
Создание ресурсов JMS из NetBeans	237
Реализация продюсера сообщений JMS	243
Обработка сообщений компонентами, управляемыми сообщениями.....	250
Наблюдение за приложением в действии	254
Резюме	256

Глава 8.**Прикладной интерфейс JSON Processing 257**

Объектная модель JSON-P	257
Создание данных в формате JSON с использованием объектной модели JSON-P	258
Пример	261
Парсинг данных в формате JSON с использованием объектной модели JSON-P	265
Потоковая модель JSON-P	268
Создание данных JSON с применением потоковой модели JSON-P	269
Парсинг данных JSON с применением потоковой модели JSON-P	271
Резюме	274

Глава 9.

Прикладной интерфейс WebSocket 275

Исследование приемов использования веб-сокетов на типовых примерах.....	275
Опробование примера приложения Echo	277
Программный код на Java	278
Программный код на JavaScript	279
Создание собственных приложений с веб-сокетами	281
Создание пользовательского интерфейса	283
Создание серверной конечной точки веб-сокета	286
Реализация поддержки веб-сокетов на стороне клиента	288
Резюме	291

Глава 10.

Веб-службы RESTful на основе JAX-RS 293

Создание веб-службы RESTful на основе существующей базы данных	294
Анализ сгенерированного кода	296
Тестирование веб-службы RESTful	300
Создание Java-клиента веб-службы RESTful.....	307
Создание JavaScript-клиента веб-службы RESTful.....	313
Резюме	317

Глава 11.

Веб-службы SOAP на основе JAX-WS..... 318

Введение в веб-службы.....	318
Создание простой веб-службы.....	319
Тестирование веб-службы	325
Создание клиента для веб-службы.....	327
Экспортирование компонентов EJB в виде веб-служб.....	332
Реализация новых веб-служб в виде EJB.....	332
Экспортирование существующих EJB в виде веб-служб	335

Создание веб-службы из существующего файла WSDL.....	338
Резюме	339

Предметный указатель	341
-----------------------------------	------------



ОБ АВТОРЕ

Дэвид Хеффельфингер (David R. Heffelfinger) – технический директор Ensoode Technology LLC – консалтинговой компании, специализирующейся на разработке программного обеспечения и расположенной в районе большого Вашингтона, округ Колумбия. Дэвид – профессиональный архитектор, проектировщик и разработчик программного обеспечения с 1995 года и использует Java в качестве основного языка программирования с 1996 года. Работал во многих крупномасштабных проектах для ряда клиентов, в числе которых департамент США по Национальной безопасности, Freddie Mac, Fannie Mae и Министерство обороны США. Имеет степень магистра в области разработки программного обеспечения Южного методического университета. Также является главным редактором Ensoode.net (<http://www.ensode.net>), веб-сайта, посвященного Java, Linux и другим технологиям. Часто выступает на конференциях Java-разработчиков, таких как JavaOne. Вы можете следовать за Дэвидом в Твиттере, его учетная запись: [@ensode](#).



О РЕЦЕНЗЕНТАХ

Саурабх Чхаджед (Saurabh Chhajed) – обладатель сертификатов «Cloudera Certified Developer for Apache Hadoop» и «Certified Java/J2EE Programmer» с 5-летним опытом профессиональной разработки корпоративных приложений с применением новейших фреймворков, инструментов и шаблонов проектирования. Имеет большой опыт применения методологий гибкой разработки и активно продвигает новые технологии, такие как NoSQL и приемы обработки Больших Данных. Саурабх оказывал помощь некоторым крупным компаниям из США в создании их корпоративных систем, что называется «с нуля». В свободное от работы время любит путешествовать и обожает делиться опытом в своем блоге (<http://saurzcode.in>).

Халиль Каракосе (Halil Karaköse) – независимый разработчик программного обеспечения. В 2005 году закончил университет Işık University в Турции с квалификацией инженера по вычислительной технике.

Десять лет работал в индустрии телекоммуникаций, в таких компаниях, как Turkcell и Ericsson. В 2014 оставил работу в Ericsson и основал собственную консалтинговую компанию KODFARKI (<http://kodfarki.com>).

Основное свое внимание он уделяет разработке программ на Java, с применением Java EE, Spring и Primefaces. Также любит проводить практические занятия по программированию на Java. Всегда проявлял большой интерес к Java-инструментам, повышающим скорость разработки, таким как NetBeans и IntelliJ IDEA. В свободное время занимается бегом, лыжами, иногда любит сразиться в «Pro Evolution Soccer».

Марио Перес Мадуро (Mario Pérez Madueño) родился в 1975 году в Турине, а сейчас живет в Барселоне. В 2010 году закончил Открытый университет Каталонии (Open University of Catalonia, UOC) с квали-

фикацией инженера по вычислительной технике. Марио – большой энтузиаст применения технологий Java SE, ME и EE, и уже много лет участвует в программе «приемочных испытаний сообществом» NetBeans (NetBeans Community Acceptance Testing program, NetCAT). Также был техническим рецензентом книг «Java EE 5 Development with NetBeans 6» и «Building SOA-based Composite Applications Using NetBeans IDE 6» (обе выпущены издательством Packt Publishing).

Я хотел бы выразить благодарность моей жене Марии (María) за ее безоговорочную помощь и поддержку всех моих начинаний, а также Мартина (Martín) и Матиаса (Matías), дающих мне силы идти вперед.

Дэвид Салтер (David Salter) – архитектор и разработчик корпоративного программного обеспечения, занимающийся этой работой с 1991 года. Истоки его отношений с Java восходят к самому началу развития этого языка, когда он использовал Java 1.0 для создания настольных приложений и апплетов для интерактивных веб-сайтов. Дэвид занимается разработкой корпоративных приложений на Java с использованием технологии Java EE (и J2EE), а также с применением открытых решений, начиная с 2001 года. Его перу принадлежат книги «NetBeans IDE 8 Cookbook» и «Seam 2.x Web Development» (обе выпущены издательством Packt Publishing). Также является соавтором книги «Building SOA-Based Composite Application Using NetBeans IDE 6», Packt Publishing.

Хочу поблагодарить мою семью за поддержку. Особое спасибо моей жене – люблю тебя.

Манжит Сингх Сони (Manjeet Singh Sawhney) – в настоящее время работает в крупной консалтинговой компании в Лондоне на должности главного консультанта по организации корпоративных данных. Прежде работал в разных крупных организациях, занимаясь разработкой программного обеспечения, оказанием помощи в выработке технических решений и организации корпоративных данных. Манжит имеет опыт использования множества языков программирования, но отдает предпочтение языку Java. Обучаясь в аспирантуре, он

также работал репетитором в одном из 100 лучших университетов в мире, где преподавал Java студентам начальных курсов и привлекался к приему экзаменов и оценке дипломных проектов. Свой профессиональный опыт Манжит приобрел в работе над несколькими ответственными проектами ПО для обслуживания клиентов в сфере финансов, телекоммуникационных услуг, розничной торговли и в государственных учреждениях.

Я очень благодарен своим родителям; моей жене Джаспал (Jaspal); моему сыну Кохинуру (Kohinoor); и моей дочери Прабхнур (Prabhnoor), за их поддержку и терпение, когда я, занимаясь рецензированием этой книги, оторвал от семьи несколько моих вечеров и выходных.



ПРЕДИСЛОВИЕ

Java EE 7 является последней версией спецификации Java EE, в которую добавлено несколько новых возможностей для упрощения разработки корпоративных приложений. В эту последнюю версию Java EE были включены новые версии существующих API Java EE. Так, например, в JSF 2.2 значительно улучшена поддержка создания диалоговых мастеров с применением FaceFlows и добавлена поддержка HTML5. В NetBeans появилась поддержка новых особенностей JPA 2.1, таких как Bean Validation и многих других. Сеансовые компоненты EJB теперь могут автоматически генерироваться средой NetBeans, что существенно упрощает использование возможностей EJB, таких как транзакции и параллельное выполнение. Дополнительные особенности CDI, такие как квалификаторы, стереотипы и другие теперь легко могут быть задействованы с помощью мастеров NetBeans. Значительно упрощена работа с JMS 2.0, что позволяет легко и быстро разрабатывать приложения, обменивающиеся сообщениями. Java EE включает новый Java API JSON Processing (JSON-P), что упрощает обработку данных в формате JSON. Кроме того, в состав NetBeans была включена поддержка некоторых особенностей, позволяющих легко и просто разрабатывать веб-службы RESTful и SOAP.

В этой книге мы исследуем все возможности NetBeans, которые предназначены для разработки корпоративных приложений Java EE 7.

Вопросы, освещаемые в книге

Глава 1, «Знакомство с NetBeans», представляет введение в NetBeans, а также знакомит с подсказками, экономящими время, и приемами, которые позволяют более эффективно разрабатывать приложения Java.

Глава 2, «Разработка веб-приложений с использованием JavaServer Faces 2.2», объясняет, как с помощью NetBeans можно облегчить разработку веб-приложений, использующих преимущества фреймворка JavaServer Faces 2.2.

Глава 3, «Библиотека компонентов JSF», показывает, насколько просто с помощью NetBeans создавать JSF-приложения с применением популярных библиотек компонентов JSF, таких как PrimeFaces, RichFaces и ICEfaces.

Глава 4, «Взаимодействие с базами данных через Java Persistence API», объясняет, как с помощью NetBeans упрощается разработка приложений, использующих возможности Java Persistence API (JPA), включая автоматическую генерацию сущностей JPA из существующих схем баз данных. В этой главе также объясняется, как сгенерировать завершенное веб-приложение из существующей схемы базы данных всего несколькими щелчками мыши.

Глава 5, «Реализация уровня бизнес-логики на сеансовых компонентах EJB», наглядно демонстрирует, насколько NetBeans упрощает разработку сеансовых компонентов EJB 3.1.

Глава 6, «Контексты и внедрение зависимостей», показывает, как новый CDI API, введенный в Java EE 6, упрощает интегрирование различных уровней корпоративного приложения.

Глава 7, «Обмен сообщениями с применением JMS и компонентов, управляемых сообщениями», посвящена технологиям обмена сообщениями Java EE, таким как Java Message Service (JMS) и Message-Driven Beans (MDB), демонстрируя функциональность NetBeans, которая упрощает разработку приложений, использующих возможности этих API.

Глава 8, «Прикладной интерфейс JSON Processing», рассказывает, как обрабатывать данные в формате JSON с применением нового прикладного интерфейса JSON-P.

Глава 9, «Прикладной интерфейс WebSocket», рассказывает, как использовать новый прикладной интерфейс Java к веб-сокетам (WebSocket) для создания веб-приложений, поддерживающих полноценные двусторонние взаимодействия между клиентом и сервером.

Глава 10, «Веб-службы RESTful на основе JAX-RS», рассматривает создание веб-служб RESTful на основе JAX-RS, попутно демонстрируя, как NetBeans может автоматически генерировать веб-службы RESTful, а также клиентские RESTful-приложения на Java и JavaScript.

Глава 11, «Веб-службы SOAP на основе JAX-WS», объясняет, как с помощью NetBeans можно облегчить разработку веб-служб SOAP с применением прикладного интерфейса Java API for XML (JAX-WS).

Что нужно для чтения этой книги

Для чтения этой книги нужно установить комплект разработчика Java – Java Development Kit (JDK) версии 7.0 (или выше) и NetBeans версии 8.0 (или выше) в редакции Java EE.

Для кого эта книга

Если вы Java-разработчик и желаете создавать приложения Java EE, используя преимущества NetBeans для автоматизации рутинных задач, эта книга для вас. Знакомство с NetBeans или Java EE совершенно необязательно.

Соглашения

В этой книге вы обнаружите несколько стилей оформления текста, которые разделяют различные виды информации. Ниже приводятся примеры этих стилей и поясняется их значение.

Элементы программного кода в тексте, имена таблиц в базах данных, имена папок и файлов, расширения файлов, пути к каталогам в файловой системе, фиктивные адреса URL, ввод пользователя и учетные записи в Twitter оформляются так: «Для поиска каталога JDK NetBeans использует переменную окружения `JAVA_HOME`».

Блоки кода оформляются следующим образом:

```
<package com.ensode.flowscope.namedbeans;

import javax.annotation.PostConstruct;
import javax.annotation.PreDestroy;
import javax.faces.flow.FlowScoped;
import javax.inject.Named;

@Named
@FlowScoped(«registration»)
public class RegistrationBean {
    ...
```

Чтобы привлечь ваше внимание к определенной части в блоке кода, соответствующие строки или элементы будут выделены жирным шрифтом:

```
package com.ensode.flowscope.namedbeans;

import javax.annotation.PostConstruct;
import javax.annotation.PreDestroy;
import javax.faces.flow.FlowScoped;
```

```
import javax.inject.Named;

@Named
@FlowScoped("registration")
public class RegistrationBean {
    ...
}
```

Любой ввод или вывод в командной строке оформляется так:

```
chmod +x filename.sh
```

Важные (ключевые) слова в тексте выделяются жирным. Слова, которые вы видите на экране, в меню или в диалогах, оформляются так: «Чтобы загрузить NetBeans, щелкните на кнопке **Download** (Загрузить)».



Так оформляются предупреждения или важные примечания.



Так оформляются советы и рекомендации.

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или может быть не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв прямо на нашем сайте www.dmkpress.com, зайдя на страницу книги и оставить комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com, при этом напишите название книги в теме письма.

Если есть тема, в которой вы квалифицированы, и вы заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Скачивание исходного кода примеров

Скачать файлы с дополнительной информацией для книг издательства «ДМК Пресс» можно на сайте www.dmkpress.com или www.dmkpress.com в разделе «Читателям – Файлы к книгам».

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы удостовериться в качестве наших текстов, ошибки всё равно случаются. Если вы найдёте ошибку в одной из наших книг – возможно, ошибку в тексте или в коде – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от расстройств и поможете нам улучшить последующие версии этой книги.

Если вы найдёте какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу dmkpress@gmail.com, и мы исправим это в следующих тиражах.

Нарушение авторских прав

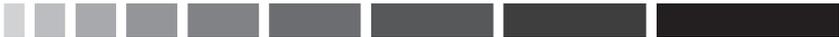
Пиратство в Интернете по-прежнему остается насущной проблемой. Издательства «ДМК Пресс» и «Pакт» очень серьезно относятся к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в Интернете с незаконно выполненной копией любой нашей книги, пожалуйста, сообщите нам адрес копии или веб-сайта, чтобы мы могли принять меры.

Пожалуйста, свяжитесь с нами по адресу электронной почты dmkpress@gmail.com со ссылкой на подозрительные материалы.

Мы высоко ценим любую помощь по защите наших авторов, и помогающую нам предоставлять вам качественные материалы.

Вопросы

Вы можете присылать любые вопросы, касающиеся данной книги, по адресу dm@dmk-press.ru или questions@packtpub.com. Мы постараемся разрешить возникшие проблемы.



ГЛАВА 1.

Знакомство с NetBeans

В этой главе мы рассмотрим, как приступить к работе с NetBeans. Будут затронуты следующие темы:

- ◆ введение;
- ◆ получение NetBeans;
- ◆ установка NetBeans;
- ◆ первый запуск NetBeans;
- ◆ настройка NetBeans для разработки Java EE-приложений;
- ◆ развертывание нашего первого приложения;
- ◆ подсказки NetBeans, повышающие эффективность разработки.

Введение

NetBeans является **интегрированной средой разработки** (Integrated Development Environment, **IDE**) и, в дополнение к этому, платформой. Хотя первоначально IDE NetBeans могла использоваться только для разработки приложений на Java, начиная с версии 6, NetBeans поддерживает несколько языков программирования. Это либо встроенная поддержка, либо поддержка, осуществляемая путем установки дополнительных расширений. NetBeans имеет встроенную поддержку следующих языков программирования: Java, C, C++, PHP, HTML и JavaScript. Посредством расширений поддерживаются также Groovy, Scala и другие языки.

Однако NetBeans не только интегрированная среда разработки, но еще и платформа. Разработчики могут использовать NetBeans API для создания расширений NetBeans или автономных приложений.



С краткой историей NetBeans можно познакомиться по адресу: <http://NetBeans.org/about/history.html>.

Хотя NetBeans поддерживает несколько языков программирования, всё-таки основным ее языком является Java, поэтому она наиболее удобна для разработки на Java. Как Java IDE, NetBeans имеет встроенную поддержку приложений Java SE (Standard Edition), которые обычно работают на настольных компьютерах или ноутбуках; приложений Java ME (Micro Edition), которые обычно работают на портативных устройствах, таких как сотовые телефоны или PDA; и приложений Java EE (Enterprise Edition), которые обычно работают на больших серверах и могут поддерживать одновременную работу тысяч пользователей.

В этой книге мы сосредоточимся на изучении возможностей NetBeans, используемых при разработке Java EE-приложений, а также на том, как максимально полно использовать возможности NetBeans, позволяющие более эффективно разрабатывать приложения Java EE.

Некоторые из функций NetBeans, которые мы рассмотрим, позволяют существенно ускорить разработку веб-приложений с использованием **JavaServer Faces (JSF)**, веб-фреймворка на стандартных компонентах Java EE, предоставляя отправные точки для артефактов такого рода. Также будет рассмотрено, как с помощью NetBeans автоматизировать создание сущностей **Java Persistence API (JPA)** из существующей схемы базы данных (JPA – стандартный инструмент объектно-реляционного отображения, включенный в состав Java EE).

В дополнение к веб-разработке будет рассмотрено, как с помощью NetBeans упрощается разработка компонентов **Enterprise JavaBeans (EJB)** и веб-служб. Мы также увидим, как просто написать компонент EJB, и клиента веб-службы, воспользовавшись некоторыми преимуществами NetBeans.

Перед тем как воспользоваться вышеупомянутыми преимуществами NetBeans, конечно, нужно установить NetBeans, как это описано в следующем разделе.

Получение NetBeans

NetBeans можно загрузить по адресу: <http://www.netbeans.org>.

Чтобы загрузить NetBeans (см. рис. 1.1), щелкните на кнопке **Download** (Загрузить). После щелчка откроется страница, со списком всех доступных дистрибутивов NetBeans (см. рис. 1.2).

Разные дистрибутивы NetBeans содержат разные комплекты с разными функциональными возможностями. В табл. 1.1 перечислены

некоторые комплекты NetBeans и описана функциональность, которую они предоставляют:

Таблица 1.1. Комплекты NetBeans

Комплект NetBeans	Описание
Java SE	Позволяет разрабатывать приложения Java для настольных компьютеров.
Java EE	Позволяет разрабатывать приложения Java Standard Edition (обычные приложения для настольных компьютеров) и Java Enterprise Edition (корпоративные приложения, работающие на «большом железе»).
C/C++	Позволяет разрабатывать приложения на языках C или C++.
HTML5 & PHP	Позволяет разрабатывать веб-приложения с использованием HTML5 и/или популярного языка PHP.
All	Включает функциональность всех комплектов поставки NetBeans

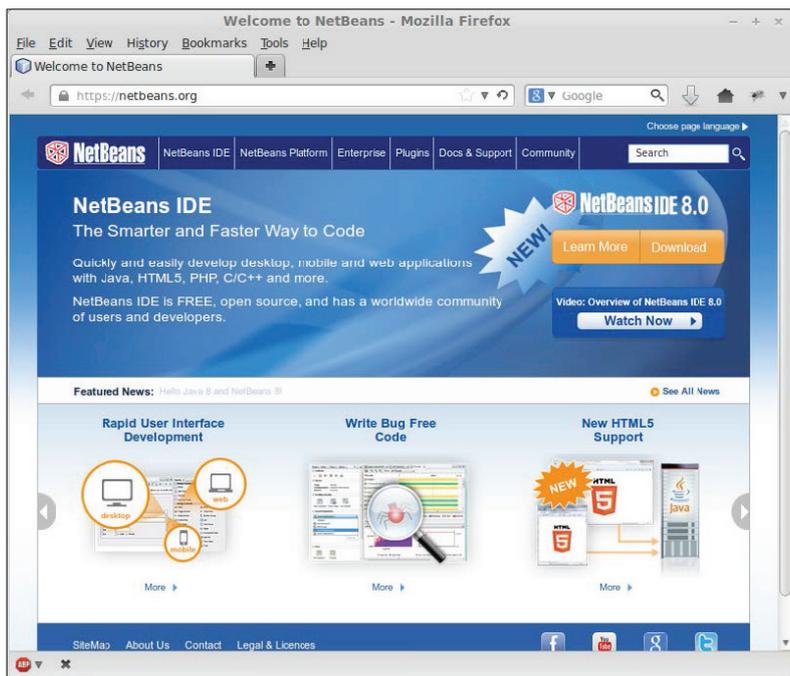


Рис. 1.1. Главная страница сайта <http://netbeans.org>

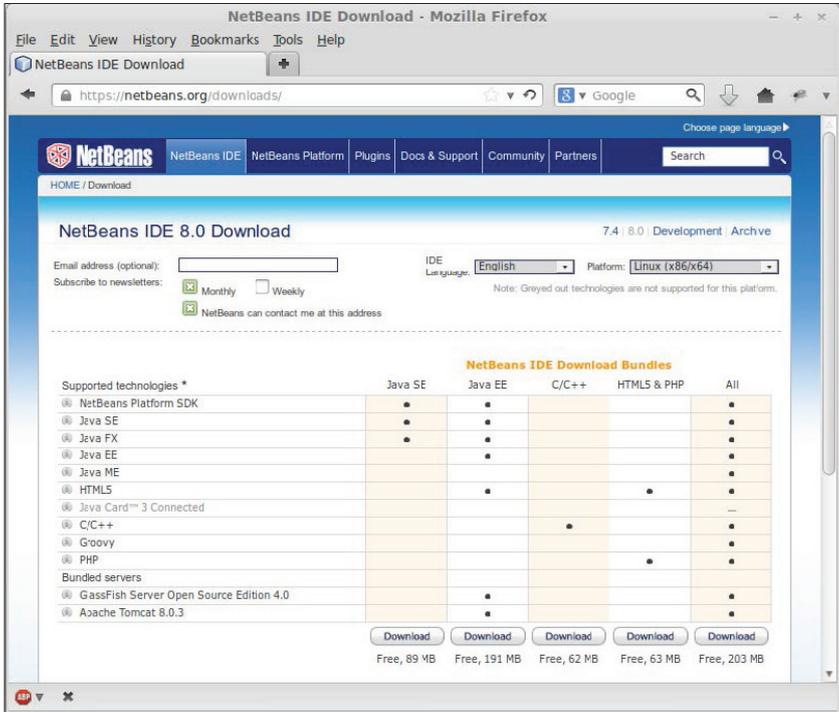


Рис. 1.2. Страница со списком дистрибутивов NetBeans

Для опробования примеров, приведенных в этой книге, необходим комплект **Java EE** или **All**.



*Все снимки экрана в этой книге были сделаны при использовании комплекта **Java EE**. В комплектации **All** NetBeans может выглядеть немного иначе, в частности можно заметить появление некоторых дополнительных пунктов меню.*

Официально поддерживаются следующие платформы:

- Windows;
- Linux (x86/x64);
- Mac OS X.

Дополнительно NetBeans может выполняться на любой платформе, где установлена версия Java 7 или выше. Также доступна для загрузки версия NetBeans, не зависящая от операционной системы, которая будет выполняться на любой из этих платформ.



Даже при том, что версия NetBeans, не зависящая от операционной системы, может выполняться на всех поддерживаемых платформах, рекомендуется использовать версию для конкретной платформы.

Страница загрузки NetBeans сама определит используемую операционную систему для получения доступа к соответствующему дистрибутиву, а используемая платформа будет выбрана по умолчанию. Если дело обстоит иначе или если требуется загрузить NetBeans для установки на другой рабочей станции, требуемую платформу можно выбрать (см. рис. 1.2) в раскрывающемся списке **Platform** (Платформа).

После выбора платформы щелкните на кнопке **Download** (Загрузить), соответствующей выбранному комплекту NetBeans. Для разработки Java EE-приложений нужен комплект Java EE или комплект All. После этого дистрибутив NetBeans будет загружен в указанный каталог.



Приложения Java EE должны развертываться на сервере приложений. На рынке существует несколько серверов приложений, между тем NetBeans в комплектациях **Java EE** и **All** уже содержат в себе GlassFish и Tomcat. Tomcat является популярным контейнером сервлета с открытым исходным кодом и может использоваться для развертывания приложений, использующих JSF. Однако он не поддерживает других технологий Java EE, таких как EJB или JPA. GlassFish – сервер приложений, полностью совместимый с Java EE. Мы будем использовать поставляемый в комплекте сервер приложений GlassFish для развертывания и выполнения наших примеров.

Установка NetBeans

Для установки NetBeans требуется наличие в системе комплекта разработчика Java (Java Development Kit, JDK) версии 1.7 или выше.



Поскольку эта книга адресована опытным разработчикам Java, мы не будем тратить много времени на объяснения, как установить и настроить JDK, так как мы можем обоснованно предположить, что все читатели этой книги уже имеют опыт установки JDK. Инструкции по установке JDK можно найти по адресу: <http://docs.oracle.com/javase/7/docs/webnotes/install/index.html>.

Установка NetBeans немного отличается в зависимости от платформы. В следующих нескольких разделах мы объясним, как установить NetBeans на каждой поддерживаемой платформе.

Microsoft Windows

NetBeans для платформ Microsoft Windows загружается в виде исполняемого файла с названием, подобным `netbeans-8.0-javaee-windows.exe` (точное имя зависит от версии и комплектности NetBeans, выбранной для загрузки). Чтобы установить NetBeans на платформах Windows, просто перейдите к папке, куда был загружен дистрибутив NetBeans, и дважды щелкните на исполняемом файле.

Mac OSx

Для Mac OS X загруженный файл называется наподобие `netbeans-8.0-javaeemacosx.dmg` (точное имя зависит от версии и комплектности NetBeans, выбранной для загрузки). Для установки NetBeans перейдите в каталог, куда был загружен файл, и дважды щелкните на нем.

Linux

NetBeans для Linux загружается в форме сценария командной оболочки. Имя файла будет похоже на `netbeans-8.0-javaee-linux.sh`, (точное имя зависит от версии и комплектности NetBeans, выбранной для загрузки).

Прежде чем NetBeans можно будет установить в Linux, загруженный следует сделать исполняемым. Это можно выполнить с помощью командной строки, перейдя в каталог, куда был загружен установщик NetBeans, и выполнив следующую команду:

```
chmod +x filename.sh
```

Замените `filename.sh` именем файла, соответствующим платформе и комплектности NetBeans. После это можно запустить установку из командной строки:

```
./filename.sh
```

И вновь замените `filename.sh` именем файла, соответствующим платформе и комплектности NetBeans.

Другие платформы

NetBeans для других платформ можно загрузить в виде независимого от платформы ZIP-файла с именем, похожим на: netbeans-8.0-201403101706-javaee.zip (точное имя файла может измениться в зависимости от конкретной версии и комплектности NetBeans, выбранной для загрузки).

Чтобы установить NetBeans на одной из этих платформ, извлеките файлы из ZIP-архива в любой подходящий каталог.

Процедура установки

Несмотря на то, что на разных платформах установка запускается по-разному, сам процесс установки мало чем отличается.



Исключением из этого правила является установка из ZIP-файла, в котором, по сути, отсутствует программа-установщик. Установка этой версии NetBeans заключается в простом извлечении файлов из архива в любой подходящий каталог.

После запуска программы установки NetBeans на экране должно появиться окно, как показано на рис. 1.3.



Рис. 1.3. Начальное окно мастера установки NetBeans