

# Программирование на языке Java.

## Новое в Java 1.1

Картузов А.В.

Скоро надо будет готовить приложение по Java 1.2 (срок выхода – декабрь 1998 года).  
А пока новшествами Java 1.1 являются :

### 1. Встроенные классы

Изменения, внесенные в сам язык и позволяющие включать классы друг в друга и в блоки программы.

### 2. Java Bean-объект

Механизм построения модульных программных компонентов, предназначенных для многократного использования.

### 3. Многоязыковая поддержка

Множество новых особенностей, позволяющих писать программы, которые работают в любой точке земного шара.

### 4. Новая модель обработки событий

Новая модель обработки событий в графических пользовательских интерфейсах, предназначенная для того, чтобы облегчить построение таких интерфейсов.

### 5. Другие новые особенности AWT

AWT в Java 1.1 включает в себя поддержку вывода на печать, операций копирования и вставки, контекстных меню, клавиш ускоренного вызова меню, а также операций передачи фокуса. В нем улучшена поддержка цвета, шрифтов, курсоров, прокрутки и манипуляции изображениями.

## **6. Апплеты**

Применение JAR-файлов позволяет сгруппировать все файлы апплета в единый архив. Цифровые подписи открывают возможность запуска апплетов, пользующихся полным доверием пользователя, в условиях минимальных ограничений, связанных с безопасностью.

## **7. Сериализация объектов**

Теперь объекты можно легко сериализовать для передачи по сети или записи на диск для постоянного хранения.

## **8. Отражение**

Программы на Java могут теперь отражаться (reflection) на самих себя или на другой класс с целью определения методов и переменных, заданных в этом классе, аргументов, передаваемых методам, и т.д. Reflection API позволяет также вызывать методы по имени.

## **9. Безопасность**

В Java 1.1 включены новые пакеты, которые обеспечивают поддержку цифровых подписей, обработку сообщений, обработку ключей и списков управления доступом.

## **10. Java Database Connectivity (JDBC)**

Новый пакет, позволяющий Java-программам посылать SQL-запросы серверам баз данных. В его состав входит "мост", который обеспечивает взаимодействие с существующими серверами баз данных ODBC.

## **11. Удаленный вызов методов**

Интерфейс для поддержки распределенных приложений на Java, в которых программа, работающая на одном компьютере, может вызывать методы Java-объектов, хранящихся на другом компьютере.

## **12. Java 1.1 : пакет за пакетом**

Пакеты и классы в библиотеке классов Java взаимосвязаны и взаимозависимы. Многие из новых особенностей Java 1.1 базируются на классах, принадлежащих к нескольким различным пакетам Java API. Ниже описывается каждый из 23 пакетов, образующих ядро Java 1.1 API. Здесь представлены новые пакеты, а также рассмотрены изменения, внесенные в имеющиеся пакеты.

## **13. Java.applet**

Кроме возможности создавать JAR-файлы и апплеты с цифровой подписью и вводить новые атрибуты в тег <applet>, в пакет java.applet никаких других существенных изменений внесено не было.

## **14. java.awt**

В пакете java.awt содержатся новые классы и интерфейсы, предназначенные для поддержки вывода на печать, контекстных меню и клавиш ускоренного вызова меню, а также для обеспечения улучшенной поддержки размещения элементов пользовательского интерфейса на экране, работы с курсорами, полосами прокрутки, цветами и отсечением изображений. В нескольких классах реализована поддержка новой модели обработки событий пакета AWT, хотя большая часть работы с событиями сконцентрирована в одном из нескольких новых пакетов, входящих в пакет java.awt.

### **14.1. java.awt.datatransfer**

Классы и интерфейсы данного пакета образуют фундамент для обмена информацией

между приложениями (а также внутри приложения). В этом пакете также содержатся классы, поддерживающие модель передачи информации путем операций копирования и вставки через буфер обмена (clipboard). Можно предполагать, что в будущем пакет будет расширен таким образом, чтобы обеспечить поддержку механизма обмена информацией посредством технологии drag-and-drop. Один из основополагающих механизмов обмена информацией, поддерживаемых данным пакетом, базируется на использовании API сериализации объектов (Object Serialization API) пакета java.io.

## 14.2. java.awt.event

В данном пакете определены классы и интерфейсы новой модели обработки событий пакета AWT. Классы и интерфейсы этого пакета можно разделить на три категории:

- Классы событий — классы, которые действительно описывают события.
- Слушатели событий (listeners) — интерфейсы, определяющие методы, которые должны быть реализованы объектами, заинтересованными в получении уведомлений о событиях определенного типа.
- Адаптеры событий (adaptors) — тривиальные безоператорные реализации интерфейсов слушателей событий, которые хорошо приспособлены для создания подклассов.

Все события и слушатели событий, определенные в этом пакете, представляют собой расширения класса EventObject или интерфейса Event-Listener, определенные в пакете java.util.

## 14.3. java.awt.image

В этом пакете содержатся два новых класса фильтров изображений, в которых используется улучшенное масштабирование изображений. Также внесены изменения в классы MemoryImageSource и PixelGrabber.

## 14.4. java.awt.peer

Изменения, внесенные в этот пакет, в основном отражают изменения в java.awt. Например, к ним относятся новые интерфейсы, которые описывают зависящие от платформы контекстные меню и полосы прокрутки.

## 15. java.beans

В этом пакете содержится широко разрекламированный JavaBeans API, предназначенный для создания и использования встраиваемых программных компонентов многократного использования. Классы и интерфейсы данного пакета можно применять на трех различных уровнях:

- Для создания средств построения приложений, которые программист (и даже не программист) может использовать при создании приложений из отдельных Bean-объектов.
- Для разработки Bean-объектов, которые можно применять в упомянутых выше средствах построения приложений.
- Для разработки приложений (без использования упомянутых выше средств построения), в которых применяются Bean-объекты.

Большинство классов и интерфейсов данного пакета предназначено для разработчиков приложений или более сложных Bean-объектов. Программистам, которые только используют Bean-объекты или разрабатывают простые Bean-объекты, не обязательно изучать большинство особенностей данного пакета. Работа средств построения приложений, применяющих Bean-объекты, базируется на Reflection API, определенном в пакете `java.lang.reflect`, а во многих Bean-объектах используются преимущества API сериализации объектов (Object Serialization API), определенного в пакете `java.io`. В JavaBeans API применяется та же модель обработки событий, что и в Java 1.1 AWT, и классы и интерфейсы данного пакета, обеспечивающие обработку событий, являются расширениями соответствующего класса и интерфейса, определенных в пакете `java.util`.

## 16. java.io

Пакет `java.io` — самый большой пакет ядра Java на сегодняшний день. Это случилось потому, что в Java 1.1 добавлены следующие компоненты:

- Полный набор новых классов потока символов в дополнение к большинству существующих классов ввода-вывода через поток байтов. Эти новые потоки чтения-записи обеспечивают высокую эффективность и многоязыковую поддержку ввода-вывода текстовой информации.

- Новые классы и интерфейсы для поддержки сериализации объектов.
- Ряд новых типов семейства IOException.

## 17. java.lang

В этом пакете содержится ряд новых семейств типа Exception и Error, а также новые классы Byte, short и Void, с добавлением которых всем простым типам данных Java (включая тип void) соответствуют определенные типы объектов. Это обстоятельство играет большую роль для пакета java.lang.reflect, где определяется новый Reflection API. К тому же, существенные изменения, направленные на поддержку работы Reflection API, внесены и в класс Class. В классах Class и ClassLoader содержатся методы, обеспечивающие нахождение ресурсов, связанных с данным классом, таких как изображения, аудиофайлы, файлы свойств и т.д. Ресурсы играют важную роль в обеспечении многоязыковой поддержки в Java 1.1.

## 18. java.lang.reflect

Этот новый пакет позволяет программе на Java проверять структуру классов Java и отражать ее на собственную структуру. В java.lang.reflect содержатся классы, которые описывают переменные, методы и конструкторы классов и позволяют программе получить исчерпывающую информацию о любом объекте, массиве, методе, конструкторе или переменной. Пакет java.Beans в огромной степени базируется именно на этом пакете.

## 19. Java.math

В этом новом пакете содержится только два класса, которые поддерживают арифметические операции над целыми числами и над числами с плавающей точкой произвольной точности. В классе BigInteger определены также методы реализации арифметических операций с модулями чисел, проверки простых чисел и других особенностей, необходимых для криптографии.

## 20. java.net

Изменения, внесенные в пакет java.net, реализованы на достаточно низком уровне. К

ним относятся: широковещательные сокет, опции для сокетов в стиле UNIX, а также исключения новых типов, которые обеспечивают большую глубину детализации при обработке сетевых исключений.

## **21. java.rmi**

В этом пакете определены основополагающие классы и интерфейсы, которые используются для удаленного вызова методов. Большинство классов данного пакета представляют собой типы исключений. Подпакеты пакета `java.rmi` обеспечивают дополнительные, более специфичные функции. В тех случаях, когда удаленным методам в качестве аргументов необходимо передавать объекты, удаленный вызов метода базируется на сериализации объектов, обеспечиваемой пакетом `java.io`.

## **22. java.rmi.dgc**

В этом небольшом пакете определены классы и интерфейсы, которые необходимы для реализации распределенного сбора мусора (DGC — Distributed Garbage Collection).

## **23. java.rmi.registry**

Это другой небольшой пакет, в котором определены классы и интерфейсы, необходимые для предоставления клиенту Java возможности осуществлять поиск удаленных объектов по имени или возможности объявления сервером Java услуг, которые он может предоставить пользователю.

## **24. java.rmi.server**

Этот пакет является самым большим среди пакетов удаленного вызова методов и представляет собой сердцевину RMI. В нем определены классы и интерфейсы, позволяющие Java-программе создавать объекты, которые могут использоваться удаленно другими Java-программами.

## **25. java.security**

В этом пакете содержатся классы и интерфейсы, являющиеся фундаментальными абстракциями элементов криптографии: открытых и закрытых ключей, сертификатов, подтверждений о приеме сообщений и цифровых подписей. В данном пакете отсутствуют реализации таких абстракций. По своему замыслу Java Security API является независимым от реализации. В Java 1.1 отсутствуют реализации по умолчанию, но наряду с этим пакетом можно использовать реализации, осуществляемые продавцом конкретного программного продукта. Базовая реализация механизма безопасности основывается на использовании класса `BigInteger`, который определен в пакете `java.math`.

## **26. java.security.acl**

В этом пакете определены интерфейсы высокого уровня и некоторые исключения, предназначенные для работы со списками управления доступом (ACL — access control list).

## **27. java.security.interfaces**

В этом пакете определено несколько интерфейсов, которые необходимы для работы схемы безопасности Java Security API, независимой от реализации.

## **28. java.sql**

В этом пакете содержится Java Database Connectivity (JDBC) API. Находящиеся здесь классы и интерфейсы позволяют программам на Java посылать SQL-запросы базам данных и получать на них ответы.

## **29. java.text**

Классы и интерфейсы этого пакета используются для организации многоязыковой поддержки. В пакете содержатся классы, предназначенные для поддержки форматов представления даты, времени, чисел и текстовых сообщений в стилях, которые приняты в местности, назначенной по умолчанию, или в любой заданной местности. В нем также содержатся классы, обеспечивающие сопоставление строк в соответствии с

правилами, принятыми в данной местности, и просмотр символов, слов и предложений в строке в соответствии со стилем, принятым в данной местности.

### **30. java.util**

Как следует из названия данного пакета, в `java.util` содержатся вспомогательные классы. В этот пакет в Java 1.1 добавлены новые классы для поддержки моделей обработки событий пакетов AWT и Java Beans, для определения местности (`locale`) и групп ресурсов (`resource bundles`), используемых для обеспечения многоязыковой поддержки, а также классы для работы с датой, временем и временными поясами.

### **31. java.util.zip**

В этом пакете реализованы классы для расчета контрольных сумм потоков данных, а также для сжатия и архивирования потоков данных с использованием библиотеки сжатия ZLIB и форматов файлов ZIP и GZIP.