



Java Introduction

android



Преподаватель

Александр Загоруйко
sunmeatrich@gmail.com



Программа курса

- Java – 16 пар
- Android – 14 пар

Программа Java SE (ПП и ООП)

- Введение, алгоритмы
- Переменные, типы данных, операции
- Условные операторы, циклы, псевдографика
- Массивы, строки, регулярные выражения
- Методы, перечисления, аннотации, JavaFX
- ООП, инкапсуляция, структуры данных, JCF
- Агрегирование, наследование, полиморфизм
- Сбор мусора, исключения, обобщения
- Streams, threads, синхронизация потоков
- Экзамен

Структура занятия

















- Ответы на вопросы по ДЗ
- Повторение прошлой темы
- Новая тема (презентация)
- Перемены 10-15 минут
- Продолжение презентации
- Практика (пишем код)
- Выдача домашнего задания

Примеры практики и ДЗ


















1. Написать программу, которая находит среднее арифметическое значение трёх вещественных чисел.
2. Написать программу, которая находит корень линейного уравнения $ax + b = 0$.
3. Пользователь вводит число и степень. Программа вычисляет указанную степень этого числа (Scanner, [Math.pow](#)).
4. Написать программу, которая предлагает пользователю ввести радиус окружности, а затем считает площадь и длину этой окружности. Число Pi задать в программе как вещественную константу.
5. Написать программу, которая предоставляет пользователю возможность ввести с клавиатуры количество гривен, и переводит это количество в доллары и евро.

В каждом блоке в среднем по 15 заданий разных уровней сложности.

Файлы заданий

 0. print.doc	28.10.2015 22:20	Документ Micros...	192 КБ
 1. scanner.doc	06.12.2015 21:59	Документ Micros...	30 КБ
 2. casting.doc	10.06.2016 10:02	Документ Micros...	33 КБ
 3. reminder.doc	28.10.2015 22:20	Документ Micros...	32 КБ
 4. conditions part 1.doc	28.10.2015 22:20	Документ Micros...	30 КБ
 5. conditions part 2.doc	20.12.2015 21:21	Документ Micros...	33 КБ
 6. bit operations.doc	07.02.2016 20:19	Документ Micros...	28 КБ
 7. loops part 1.doc	20.12.2015 21:19	Документ Micros...	34 КБ
 8. loops part 2.doc	28.10.2015 22:20	Документ Micros...	197 КБ
 9. arrays part 1.doc	20.12.2015 21:23	Документ Micros...	32 КБ
 10. arrays part 2.doc	04.02.2016 10:30	Документ Micros...	123 КБ
 11. java fx.doc	11.02.2016 12:14	Документ Micros...	37 КБ
 12. strings.doc	08.02.2016 9:51	Документ Micros...	49 КБ
 13. methods.doc	08.04.2016 15:47	Документ Micros...	34 КБ
 14. recursion.doc	16.03.2016 22:42	Документ Micros...	45 КБ
 15. final task.doc	03.03.2016 12:02	Документ Micros...	273 КБ

Файлы презентаций

 0. hello world android.ppt	22.12.2016 12:59	Презентация Мис...	1 992 КБ
 0. hello world qa.ppt	14.11.2016 11:11	Презентация Мис...	1 896 КБ
 1. programming introduction.ppt	22.12.2016 12:59	Презентация Мис...	1 490 КБ
 2. data types, variables.ppt	22.12.2016 12:58	Презентация Мис...	529 КБ
 3. casting.ppt	22.12.2016 12:58	Презентация Мис...	230 КБ
 4. conditions.ppt	22.12.2016 12:58	Презентация Мис...	376 КБ
 5. bit operations.ppt	22.12.2016 12:57	Презентация Мис...	372 КБ
 6. while, do while.ppt	22.12.2016 12:57	Презентация Мис...	271 КБ
 7. for, nested loops.ppt	22.12.2016 12:57	Презентация Мис...	275 КБ
 8. arrays.ppt	22.12.2016 12:56	Презентация Мис...	513 КБ
 9. sorting.ppt	22.12.2016 12:56	Презентация Мис...	1 699 КБ
 10. multidimensional arrays.ppt	22.12.2016 12:55	Презентация Мис...	868 КБ
 11. strings.ppt	22.12.2016 12:54	Презентация Мис...	632 КБ
 12. regular expressions.ppt	22.12.2016 12:54	Презентация Мис...	989 КБ
 13. static methods.ppt	22.12.2016 12:55	Презентация Мис...	422 КБ
 14. recursion.ppt	22.12.2016 12:55	Презентация Мис...	3 827 КБ
 15. java convention.ppt	19.12.2016 11:33	Презентация Мис...	248 КБ

Mystat

← → ↻ <https://mystat.itstep.org>

Mystat

Вход в систему

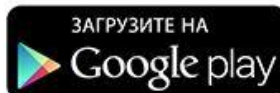
Одесса ▾

Alex

































.....

Войти

Получить доступ в Mystat



Сдача ДЗ

	ФИО студента	 14.06.16 	 10.06.16 	 07.06.16 	 03.06.16 	 31.05.16 	 27.05.16 	 24.05.16 	 13.05.16 
1	 Даценко Валентина Вячеславовна (Нет допуска)	<input type="text" value="0"/>	<input type="text" value="0"/>	12	12	12	12	12	10
2	 Ищенко Татьяна Александровна (Нет допуска)	<input type="text" value="0"/>	12	12 	12	12	<input type="text" value="0"/>	12	10
3	 Косс Дмитрий Александрович (Допуск с разрешения бухгалтерии)	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	12	<input type="text" value="0"/>
4	 Кушнеров Александр Петрович	<input type="text" value="0"/>	12	12 	12	12	12	12	10
5	 Липа Александр Александрович	<input type="text" value="0"/>	<input type="text" value="0"/>	12	12	12	11	12	10
6	 Мантас Екатерина Александровна (Допуск с разрешения бухгалтерии)	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	12 	12	12	12	10 
7	 Матюшкин Петр Сергеевич	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	12	10	12	9 	<input type="text" value="0"/>
8	 Михова Алла Ивановна	<input type="text" value="0"/>	11 	12	12	11 	12	12 	10

Файловый сервер

\\fs \ Преподаватели
\ Загоруйко А \ Public

Литература

<https://yadi.sk/d/fFCsGkIPyMHRF>

- Сьерра, Бейтс (Headfirst)
- Шилдт
- Хорстманн
- Блох
- Эккель



Литература must read

<https://yadi.sk/d/QFOmkkfyq3puD>

- Для (будущих) тимлидов
- Про компиляторы
- Личностное развитие
- Проектирование
- Публичные выступления
- Собеседования
- Совершенный код

История развития языка

В 1991 году в компании «Sun Microsystems» команда программистов, среди которых были Патрик Нотон, Билл Джой и Джеймс Гослинг начали работу над проектом «Green», целью которого было создать аппаратную часть – компьютер, и программное обеспечение для этого компьютера для повседневного использования.

Создатели языка



Патрик Нотон



Джеймс Гослинг



Билл Джой

Вначале был C++

Чтобы не связывать разработку с конкретной платформой, Гослинг и его команда начали с расширения компилятора C++. Со временем, однако, стало ясно, что один C++, как его ни расширяй, не сможет удовлетворить всем потребностям. Поэтому в середине 1991 года был задуман новый язык **Oak**.

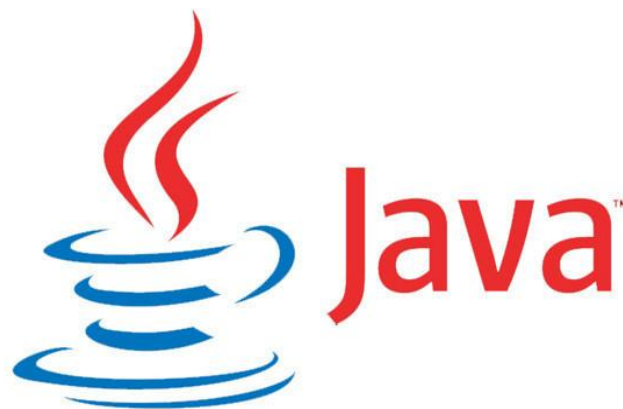
История развития языка

Язык **Oak** («**Дуб**») разрабатывался для программирования бытовых электронных устройств. Впоследствии он был переименован в **Java** и стал использоваться для написания клиентских приложений и серверного программного обеспечения.



История развития языка

Язык назван в честь марки кофе Java, поэтому на официальной эмблеме языка изображена чашка с



парящим кофе. Существует и другая версия происхождения названия языка, связанная с аллюзией на кофе-машину, как пример бытового устройства, для программирования которого изначально создавался язык.

19 августа 1991 года

Коллектив разработчиков команды **Green** демонстрирует идеи базового пользовательского интерфейса и графическую систему основателю компании Sun **Скотту Макнили**.



4 сентября 1992 года

Завершена разработка устройства Star7

<https://www.youtube.com/watch?v=1CsTH9S79qI>



Лето 1993 года

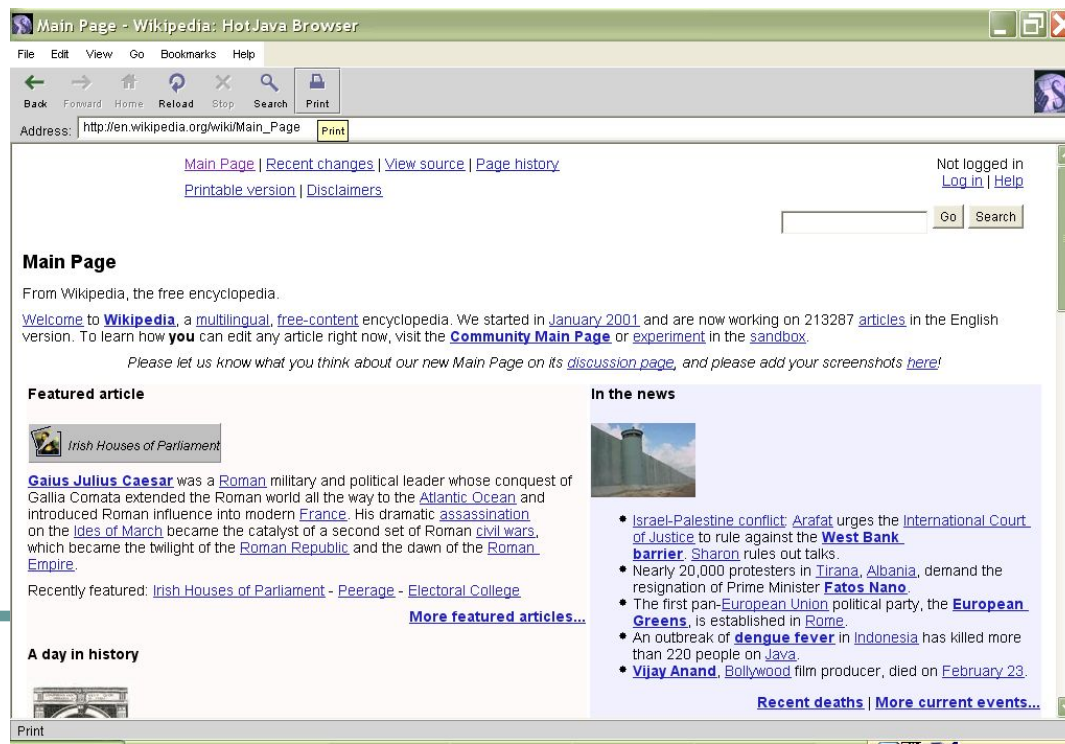
Патрик Нотон пролетает 300 тыс. миль, продавая Оак всем, кто занимается бытовой электроникой и интерактивным телевидением; тем временем темп, с которой люди получают доступ к Internet, головокружительно нарастает.

Также ведутся переговоры с компанией 3DO относительно разработки ОС для приставок. Президент 3DO Трип Хокинс (основатель Electronic Arts) предлагает купить технологию, но Макнили отказывается, и сделка срывается.

The logo for 3DO, featuring the letters '3DO' in a bold, white, serif font against a black background. A small trademark symbol (TM) is visible to the right of the 'O'.

Обращение к web

К осени 1994 года Нотон и его коллега по Sun Джонатан Пэйн закончили писать WebRunner (HotJava) - браузер, написанный на языке Java.



23 мая 1995 года

Компания Sun официально представляет Java и HotJava на выставке SunWorld '95.



Кто владеет технологией

Java – это кроссплатформенный, объектно-ориентированный (пусть и не на 100%, в отличие от C#) язык программирования. С 2010 года владельцем технологии Java является компания Oracle.

<https://www.oracle.com/java/index.html>

<http://docs.oracle.com/en/>

ORACLE

Кроссплатформенность

Программы на Java транслируются в байт-код, выполняемый виртуальной машиной Java (JVM) — программой, обрабатывающей байтовый код и передающей инструкции оборудованию как интерпретатор. Достоинством подобного способа выполнения программ является полная независимость байт-кода от операционной системы и оборудования, что позволяет выполнять Java-приложения на любом устройстве, для которого существует соответствующая виртуальная машина.

Безопасность

Другой важной особенностью технологии Java является гибкая система безопасности, в рамках которой исполнение программы полностью контролируется виртуальной машиной. Любые операции, которые превышают установленные полномочия программы (например, попытка несанкционированного доступа к данным или соединения с другим компьютером), вызывают немедленное прерывание.

Надёжность

В языке **Java** отсутствуют механизмы, потенциально приводящие к ошибкам: арифметика указателей, неявное преобразование типов с потерей точности и т.п. Присутствует строгий контроль типов, обязательный контроль исключительных ситуаций. Многие логические ошибки обнаруживаются на этапе компиляции.

Стандартные библиотеки

Многие задачи, встречающиеся при разработке программного обеспечения, уже решены в рамках стандартных библиотек. Использование объектно-ориентированного подхода позволяет легко использовать готовые объекты в своих программах. Для запуска приложения необходима установка **JRE**, содержащей полный набор библиотек, даже если не все из них используются в приложении.

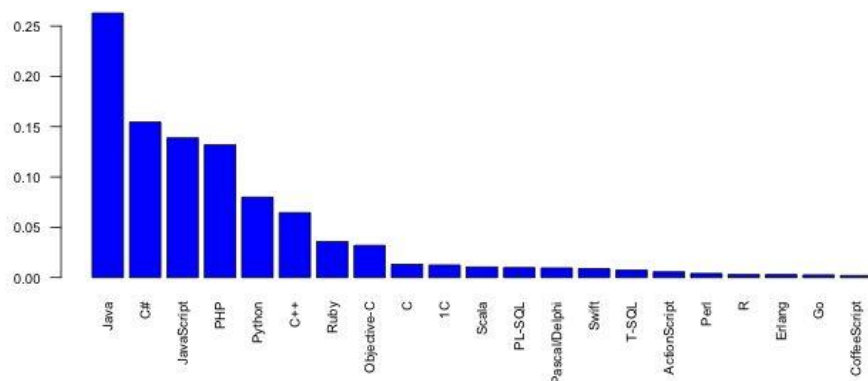
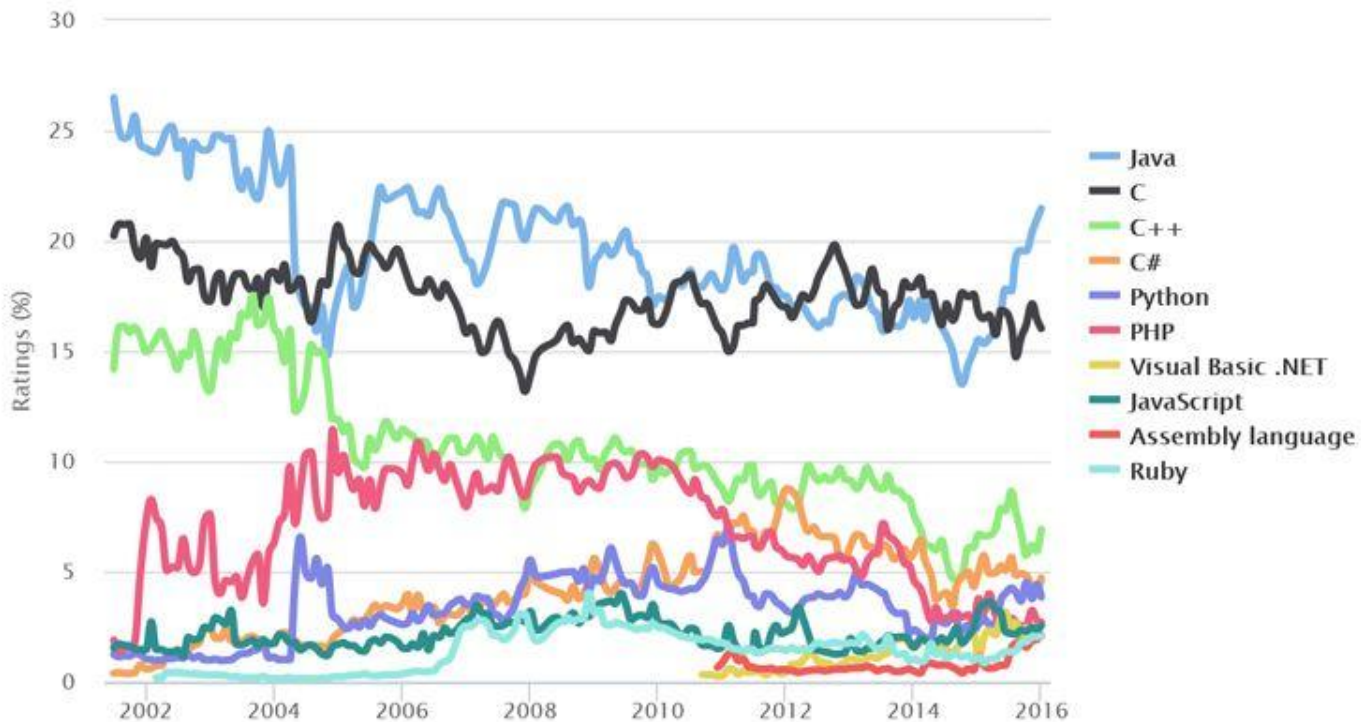
Другие плюсы

- Освобождение памяти при работе программы осуществляется автоматически с помощью «сборщика мусора»
- Имеется механизм автоматического генерирования документации на основе комментариев, размещённых в тексте программ
- На языке **Java** возможно реализовать абсолютно разные по способу функционированию и сфере использования программы

Мода на виртуальные машины

Идеи, заложенные в концепцию и различные реализации среды виртуальной машины Java, вдохновили множество энтузиастов на расширение перечня языков, которые могли бы быть использованы для создания программ, исполняемых на виртуальной машине. Эти идеи нашли также выражение в спецификации общеязыковой инфраструктуры CLI, заложенной в основу платформы .NET компанией Microsoft.

Рейтинги языков



№	Язык	Доля рынка, %
1	Java	26.3
2	C#	15.4
3	JavaScript	13.8
4	PHP	13.1
5	Python	8
6	C++	6.4
7	Ruby	3.5
8	Objective-C	3.2
9	C	1.3
10	1C	1.2
11	Scala	1.0

Семейства технологий Java

Java SE — Java Standard Edition, основное издание Java, содержит компиляторы, API, Java Runtime Environment; подходит для создания пользовательских приложений, в первую очередь — для настольных систем.

Java EE — Java Enterprise Edition, представляет собой набор спецификаций для создания программного обеспечения уровня предприятия.

Java ME — Java Micro Edition, создана для использования в устройствах, ограниченных по вычислительной мощности, например, в мобильных телефонах, КПК, встроенных системах;

JavaFX — технология, являющаяся следующим шагом в эволюции Java как Rich Client Platform; предназначена для создания графических интерфейсов корпоративных приложений и бизнеса.

Java Card — технология предоставляет безопасную среду для приложений, работающих на смарт-картах и других устройствах с очень ограниченным объёмом памяти и возможностями обработки.

Где используется Java?

По утверждению Oracle, более 3 миллиардов устройств работают на Java. Большинство крупных компаний (Amazon, eBay, LinkedIn, Yahoo) так или иначе используют Java. В конце-концов, Minecraft написан на Java 😊

Но если серьёзно, то где применяется Java в реальном мире?

Android

Android — свободная операционная система для мобильных телефонов, планшетов, смарт-часов и телевизоров, использующая ядро Linux, и принадлежащая **Google**. С момента выхода первой версии в сентябре 2008 года произошло более 45 обновлений системы. Эти обновления касаются исправления обнаруженных ошибок и добавления новой функциональности в систему.



Финансовая сфера

Многие мировые инвестиционные банки, типа Goldman Sachs, Citigroup, Barclays, Standard Chartered и другие используют Java для написания фронт-энд и бэк-энд офисных электронных систем, систем регулирования и подтверждения, проектов обработки данных и некоторых других.

Преимущественно Java используется при написании серверных приложений, в большинстве своём без какого-либо пользовательского интерфейса, которые получают данные с одного сервера, обрабатывают их и отправляют дальше.

Web

Также Java широко используется в области web-приложений. Огромное количество RESTful сервисов было создано с использованием Spring MVC, Struts 2.0 и похожих фреймворков. Даже простейшие приложения, основанные на Servlet, JSP и Struts, достаточно популярны в различных государственных проектах. Многие web-приложения государственных, оздоровительных, страховых, образовательных, оборонительных организаций написаны на Java.

J2ME

Несмотря на то, что появление iOS и Android практически уничтожило J2ME рынок, в мире ещё огромное количество дешёвых телефонов, использующих J2ME. J2ME всё ещё популярен в таких средствах, как Blu-ray, карточки и телевизионные приставки. Одна из причин такой популярности WhatsApp – он также доступен на J2ME.

Возможности языка

- автоматическое управление памятью
- расширенные возможности обработки исключительных ситуаций
- богатый набор средств фильтрации ввода-вывода
- набор стандартных коллекций: `arraylist`, `lists`, `sets`, `maps` и т.д.
- наличие простых средств создания сетевых приложений
- наличие классов, позволяющих выполнять HTTP-запросы и обрабатывать ответы
- встроенные в язык средства создания многопоточных приложений
- унифицированный доступ к базам данных
- поддержка обобщений (начиная с версии 1.5) и многое другое

Версии языка

- JDK 1.0 (23 января 1996)
- JDK 1.1 (19 февраля 1997)
- J2SE 1.2 (8 декабря 1998)
- J2SE 1.3 (8 мая 2000)
- J2SE 1.4 (6 февраля 2002)
- J2SE 5.0 (30 сентября 2004)
- Java SE 6 (11 декабря 2006)
- Java SE 7 (7 июля 2011)
- Java SE 8 (18 марта 2014 года)
- Java SE 9 (ожидается 21 сентября 2017 года)

Виртуальная машина

Чтобы пользоваться программным обеспечением, написанным на Java или разрабатывать свои собственные программы, понадобится скачать и установить на компьютер **виртуальную машину Java** - Java Virtual Machine, или сокращённо JVM. JVM - это программа, являющаяся прослойкой между операционной системой и Java программой.

Виртуальная машина

Виртуальная машина – абстрактное вычислительное устройство, которое может быть реализовано программно. Она имеет логический набор команд, определяющих операции для вычислительного устройства. По сути, это небольшая операционная система. Она формирует необходимый уровень абстракции, где достигается независимость от платформы и используемого оборудования.

Переносимость

Переносимость в Java достигается за счет того, что Java-программа (.java) компилируется не непосредственно в команды какой-либо конкретной ЭВМ, а в, так называемый, **байт-код Java** - команды некоторой абстрактной машины, называемой виртуальной машиной Java (**JVM**). Конечным результатом (исполняемым модулем) является файл класса (.class) - программа в байт-коде.

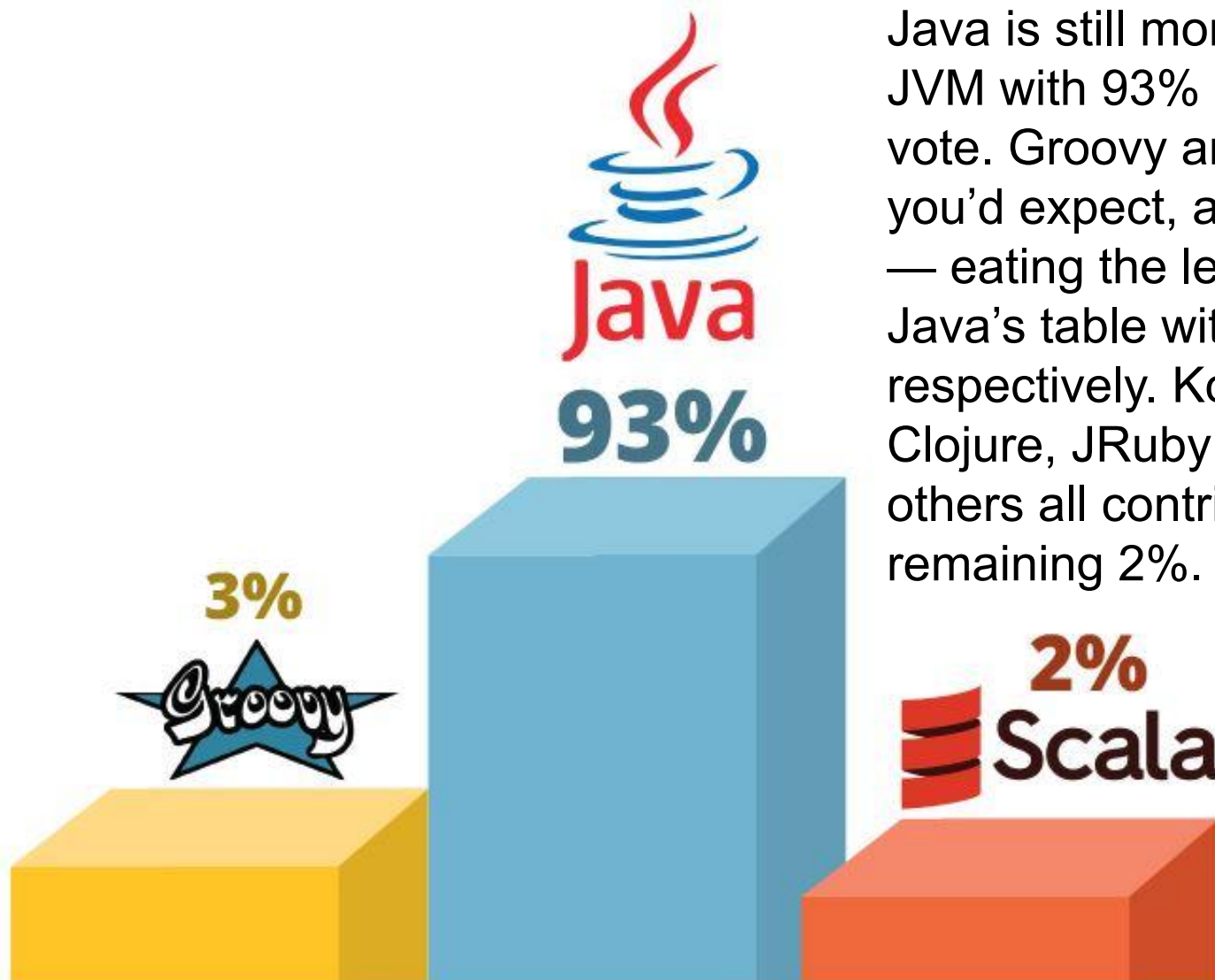
Состав JVM

- **загрузчик классов** (class loader), выполняющий загрузку классов в виртуальную машину во время выполнения программы
- **верификатор классов** (class verifier), выполняющий при загрузке класса проверку корректности его байт-кода
- **интерпретатор** (runtime interpreter), выполняющий интерпретацию команд байт-кода
- **Just-In-Time (JIT) – компилятор**, выполняющий при первом вызове каждого метода его компиляцию в объектный код целевой платформы (**native**-код)

Состав JVM

- **стэк (stack)** - структура данных, используемая для хранения локальных переменных, и передачи параметров при вызове методов
- **куча (heap)** – нераспределенная область динамической памяти для размещения объектов
- **сборщик мусора (garbage collector)** – постоянно действующий код, освобождает неиспользуемые области памяти

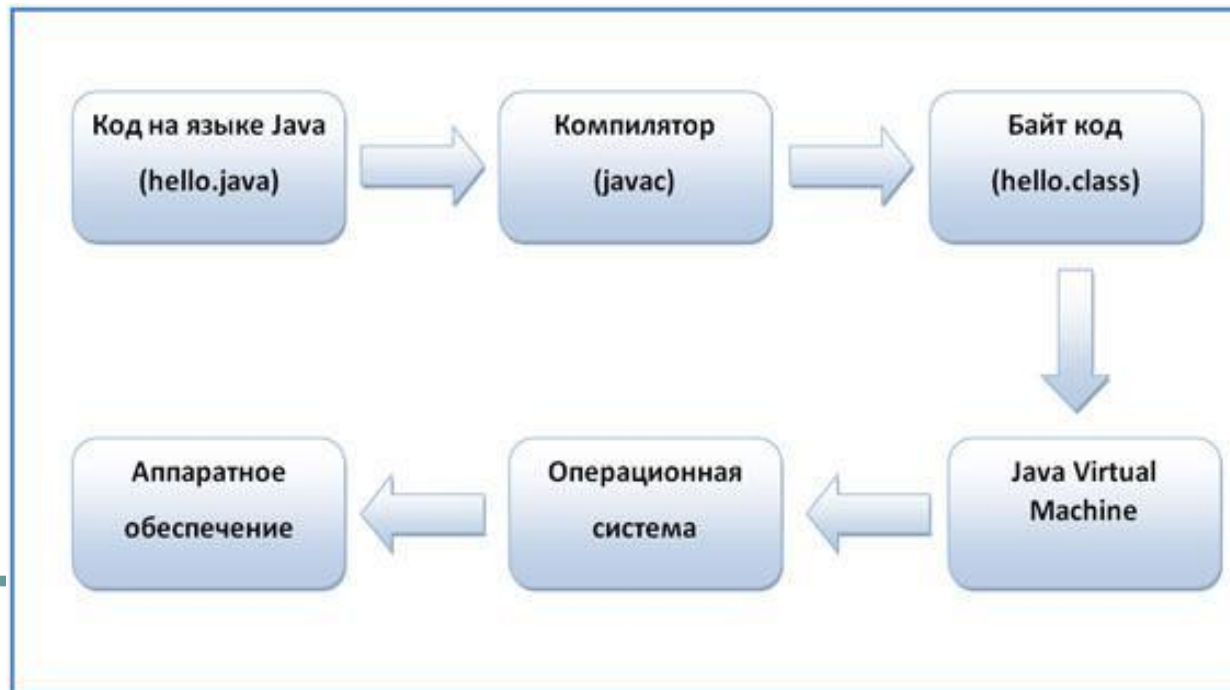
ЯЗЫКИ JVM



Java is still monopolizing the JVM with 93% of the total vote. Groovy and Scala, as you'd expect, are next in line — eating the leftovers from Java's table with 3% and 2% respectively. Kotlin, Ceylon, Clojure, JRuby and many others all contribute to the remaining 2%.

Java архитектура

- Программист создаёт файл с расширением JAVA и помещает в него исходный код программы
- С помощью компилятора на основании исходного кода создаётся файл с расширением CLASS, в него компилятор помещает байтовый код программы - команды для виртуальной машины
- При запуске программы виртуальной машиной загружается и выполняется файл с байт-кодом



Байт-код

Байт-код Java — набор инструкций, исполняемых виртуальной машиной Java. Каждый код операции байт-кода — один байт. Используются не все 256 возможных значений кодов операций. 51 из них зарезервированы для использования в будущем.

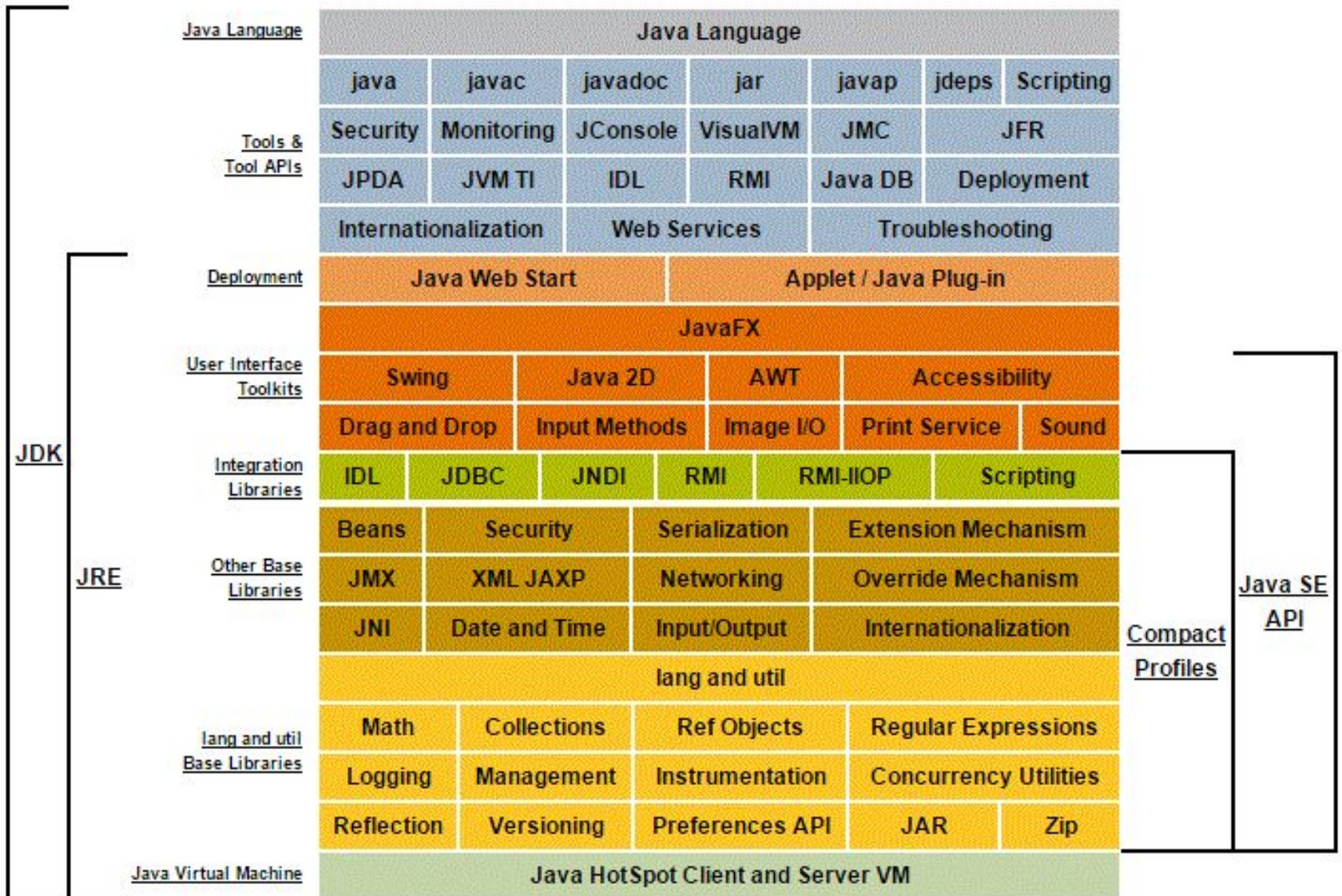
```
0000000000: CA FE BA BE 00 00 00 32 00 15 0A 00 04 00 11 09
0000000010: 00 03 00 12 07 00 13 07 00 14 01 00 03 62 61 72
0000000020: 01 00 12 4C 6A 61 76 61 2F 6C 61 6E 67 2F 53 74
0000000030: 72 69 6E 67 38 01 00 06 3C 69 6E 69 74 3E 01 00
0000000040: 03 28 29 56 01 00 04 43 6F 64 65 01 00 0F 4C 69
0000000050: 6E 65 4E 75 60 62 65 72 54 61 62 6C 65 01 00 06
0000000060: 67 65 74 42 61 72 01 00 14 28 29 4C 6A 61 76 61
0000000070: 2F 6C 61 6E 67 2F 53 74 72 69 6E 67 38 01 00 06
0000000080: 73 65 74 42 61 72 01 00 15 28 4C 6A 61 76 61 2F
0000000090: 6C 61 6E 67 2F 53 74 72 69 6E 67 38 29 56 01 00
00000000A0: 0A 53 6F 75 72 63 65 46 69 6C 65 01 00 08 46 6F
00000000B0: 6F 2E 6A 61 76 61 0C 00 07 00 08 0C 00 05 00 06
00000000C0: 01 00 03 46 6F 6F 01 00 10 6A 61 76 61 2F 6C 61
00000000D0: 6E 67 2F 4F 62 6A 65 63 74 00 21 00 03 00 04 00
00000000E0: 00 00 01 00 02 00 05 00 06 00 00 00 03 00 01 00
00000000F0: 07 00 08 00 01 00 09 00 00 00 1D 00 01 00 01 00
000000100: 00 00 05 2A B7 00 01 B1 00 00 00 01 00 0A 00 00
000000110: 00 06 00 01 00 00 00 01 00 01 00 0B 00 0C 00 01
000000120: 00 09 00 00 00 1D 00 01 00 01 00 00 00 05 2A B4
000000130: 00 02 B0 00 00 00 01 00 0A 00 00 00 06 00 01 00
000000140: 00 00 04 00 01 00 00 00 0E 00 01 00 09 00 00 00
000000150: 22 00 02 00 02 00 00 00 06 2A 28 B5 00 02 B1 00
000000160: 00 00 01 00 0A 00 00 00 0A 00 02 00 00 00 07 00
000000170: 05 00 08 00 01 00 0F 00 00 00 02 00 10
```

Инструкции байт-кода

Инструкции можно разделить на несколько групп:

- загрузка и сохранение (например, ALOAD_0, ISTORE)
- арифметические и логические операции (например, IADD, FCMPL)
- преобразование типов (например, I2B, D2I)
- создание и преобразование объекта (например, NEW, PUTFIELD)
- управление стеком (например, DUP, POP)
- операторы перехода (например, GOTO, IFEQ)
- вызовы методов и возврат (например, INVOKESTATIC, IRETURN)

Концептуальная диаграмма



SDK

SDK (от англ. *software development kit*) — комплект средств разработки, который позволяет специалистам по программному обеспечению создавать приложения для определённого пакета программ, программного обеспечения базовых средств разработки, аппаратной платформы, компьютерной системы, игровых консолей, операционных систем и прочих платформ.

JDK

Java Development Kit (сокращенно JDK) — бесплатно распространяемый компанией Oracle Corporation комплект разработчика приложений на языке Java, включающий в себя компилятор Java (javac), стандартные библиотеки классов Java, примеры, документацию, различные утилиты и исполнительную систему Java (JRE).

JRE

Java Runtime Environment — минимальная реализация виртуальной машины, необходимая для исполнения Java-приложений, без компилятора и других средств разработки. Состоит из **виртуальной машины и библиотеки Java-классов**. JRE распространяется свободно и для большинства платформ может быть загружена с сайта Oracle. Входит в состав JDK.

IDE

Все современные интегрированные среды разработки приложений на Java, такие как NetBeans IDE, Android Studio, IntelliJ IDEA, Borland JBuilder, Eclipse - опираются на сервисы, предоставляемые JDK. Большинство из них для компиляции Java-программ используют компилятор из комплекта JDK.

NetBeans

NetBeans IDE — свободная интегрированная среда разработки для всех платформ Java — Java ME, Java SE и Java EE. Пропагандируется Oracle, владельцем технологии Java, как базовое средство для разработки ПО на языке Java и других языках (C, C++, PHP, Fortran и др.)

<https://netbeans.org/>

Eclipse

Eclipse IDE — свободная интегрированная среда разработки для Java SE, Java EE и Java ME. Пропагандируется IBM, одним из важнейших разработчиков корпоративного ПО.

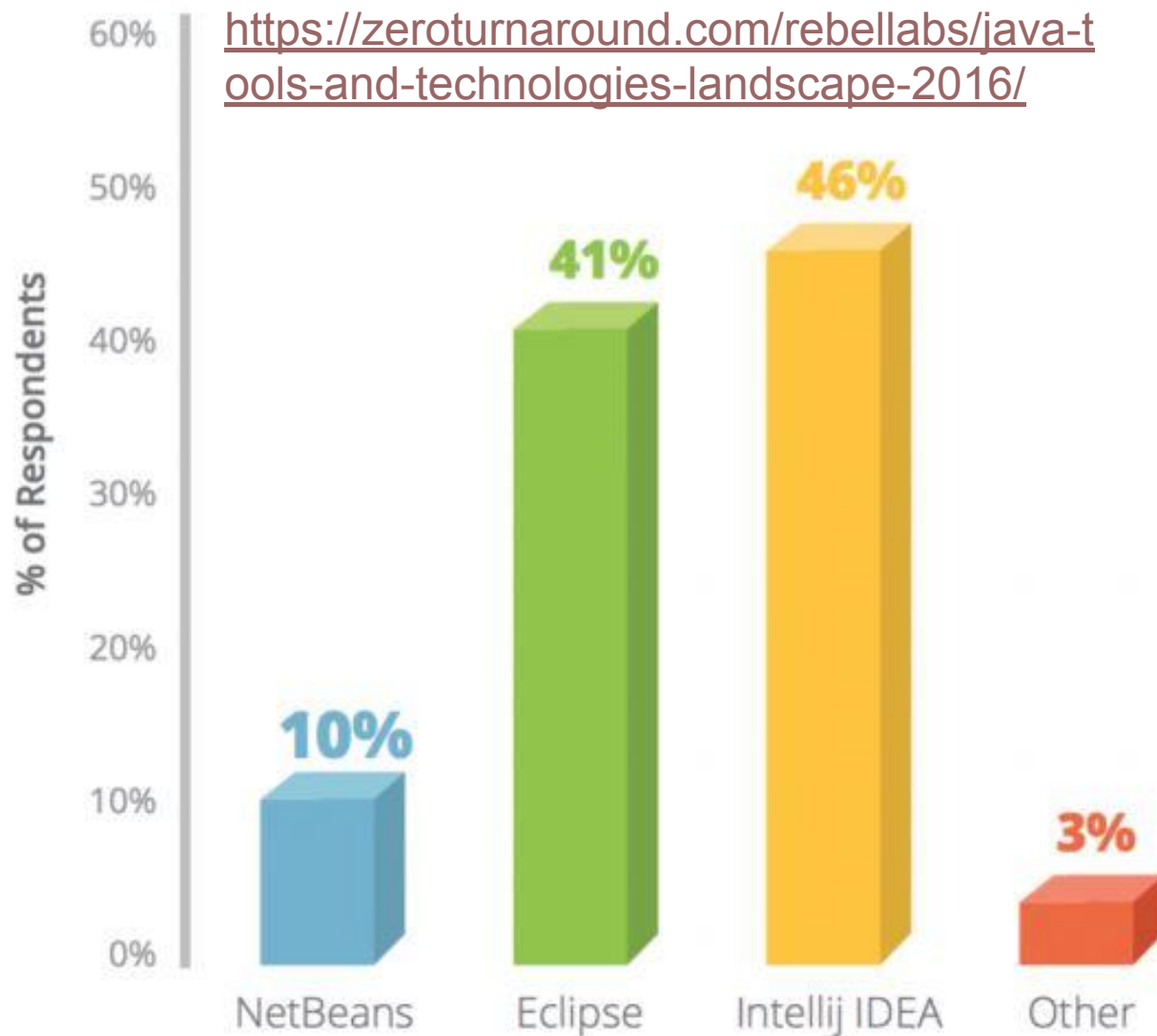
<https://www.eclipse.org/downloads/>

IDEA

IntelliJ IDEA — среда разработки для платформ Java SE, Java EE и Java ME. Разработчик — компания JetBrains. Распространяется в двух версиях: свободной бесплатной (Community Edition) и коммерческой проприетарной (Ultimate Edition).

<https://www.jetbrains.com/idea/download/>

Рейтинг IDE



Полезные ссылки

- <http://habrahabr.ru/post/67777/>
- <http://www.quizful.net/test>
- <http://www.vogella.com/tutorials/java.html>
- <http://developer.alexanderklimov.ru/android/java/java.php>
- <http://jexp.ru>

Домашнее задание

- Установить виртуальную машину Java
- Установить одну или несколько IDE
- Прочитать 1-2 начальные главы рекомендованных учебников по Java