1. **Объясните понятия JRE, JDK, JavaSE, JavaME, JavaEE**

**JRE(Java Runtime Environment)** - минимальная реализация виртуальной машины, необходимая для исполнения Java-приложений, без компилятора и других средств разработки. Состоит из виртуальной машины Java Virtual Machine и библиотеки Java-классов.

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

**Что такое Java SE/EE/ME? -** Это разные редакции Java.

**Java Standart Edition (SE)** - это стандартная редакция Java, используемая для разработки простых Java приложений. Используя данную редакцию можно создавать аплеты, консольные приложения, приложения с графическим интерфейсом пользователя.

**Java Enterprise Edition (EE)**- редакция для разработки распределнных приложений масштаба предприятия. Включает в себя технологию Enterprise Java Beans (EJB), Java Server Pages (JSP) и сервлеты (Servlets). Кроме того, Java EE включает в себя спецификацию на разработку корпоративных приложений, согласно которой следует строить подобные приложения. Java EE и .Net сейчас два основных соперника на рынке решений для разработки корпоративных приложений.

**Java Micro Edition (ME)** - компактная версия платформы Java SE представляет собой надежную и настраиваемую среду для приложений, работающих на мобильных и встраиваемых устройствах. Сейчас получила широкую популярность, в связи с развитием мобильных микропроцессорных устройств. В нее входят "облегченные" стандартные классы и классы для написания мидлетов (Midlets). Мидлеты - это аналоги аплетов, но только приспособленные специально для небольших устройств. В них так-же поддерживается графика, звук, реакция на события (нажатие кнопок и т.д.). Java ME наверно наиболее полно соответствует начальному предназначению Java - платформы для написания программ для бытовых устройств.

1. **Перечислите основные свойства и преимущества платформы Java.**

Java — это язык программирования общего назначения, который следует парадигме объектно-ориентированного программирования и подходу «Написать один раз и использовать везде» . Java используется для настольных, сетевых, мобильных и корпоративных приложений. Java можно найти везде. Это основной язык [разработки для Android](https://www.altexsoft.com/blog/engineering/top-20-tools-for-android-development/). Он используется в веб-приложениях, правительственных веб-сайтах и ​​технологиях обработки больших данных, таких как Hadoop и Apache Storm.

#### Java — это язык высокого уровня. Независимость от платформы. Язык для распределенного программирования и комфортной удаленной совместной работы. Его код сначала транслируется в специальный байт-код, независимый от платформы. А затем этот байт-код выполняется виртуальной машиной JVM. В то же время Java не является и чисто компилируемым языком, как С или С++. Подобная архитектура обеспечивает кроссплатформенность и аппаратную переносимость программ на Java.

#### Java C-подобный язык, что упрощает переход на него с C# и C++.

#### Кроме того, Java - полностью объектно-ориентированный язык, даже в большей степени, чем С++. Все сущности в языке Java являются объектами, за исключением немногих основных типов (primitive types), например чисел. В свое время объектно-ориентированное программирование (ООП) заменило структурное программирование.

#### Важно и то, что разрабатывать на Java программы, которые не содержат ошибок, значительно легче, чем на С++.

#### Все дело в том, что разработчиками языка Java из компании Sun был проведен фундаментальный анализ программ на языке С++. Анализировались "узкие места" исходного кода, которые и приводят к появлению трудновыявимых ошибок. Поэтому было принято решение проектировать язык Java с учетом возможности создавать программы, в которых были бы скрыты наиболее распространенные ошибки.

#### Является ли язык Java компилируемым или интерпретируемым? –интерпретируемый

#### Его код сначала транслируется в специальный байт-код, который выполняется виртуальной машиной JVM. В то же время Java не является и чисто компилируемым языком.

#### Что такое механизм автоматической сборки мусора (garbage collector)

#### Это программа, которая работает на [виртуальной машине Java](http://en.wikipedia.org/wiki/Java_Virtual_Machine), который избавляется от объектов, которые больше не используются Java-приложением. Это форма автоматического управления памятью.

Garbage Collector должен делать всего две вещи:

- Обнаруживать мусор

- Очищать память от мусора

## Как Garbage Collector обнаруживает мусор?

Существует два подхода к обнаружению мусора:

- Reference counting

- Tracing

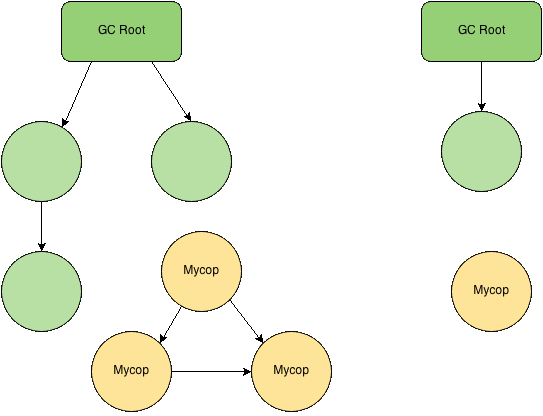
**Reference counting**

Суть подхода состоит в том, что каждый объект имеет счетчик. Счетчик хранит информацию о том, сколько ссылок указывает на объект. Kогда ссылка уничтожается, счетчик уменьшается. Если значение счетчика равно нулю, - объект можно считать мусором и память можно очищать.

**Tracing**

В “Tracing” главная идея состоит в мысли: “Живые объект - те до которых мы можем добраться с корневых точек (GC Root), все остальные - мусор. Все что доступно с живого объекта - также живое”.

Если мы представим все объекты и ссылки между ними как деревоб то нам нужно пройти с корневых узлов по всем узламю. При этом узлы, до которых мы сможем добраться - не мусор, все остальные - мусор.



Что такое корневая точка (GC Root)?

Литература говорит что существует 4 типа корневых точек:

* Локальные переменные и параметры методов
* Java Потоки
* Статические переменные
* Ссылки из JNI

Самое простое java приложение будет иметь такие корневые точки:

* Локальные переменные внутри main метода, параметры main метода.
* Поток который выполняет main.
* Статические переменные класса, внутри которого находится main метод.

1. **В чем сходства и различия Java, C/C++, С#?  
   Сходства:**  
     
   - C# похож на Java. В него включены языковые свойства Java, такие как единичное наследование, интерфейсы, близкий синтаксис и компиляция в промежуточный формат. Но C# отличается от Java использованием свойств, позаимствоннах из Delphi (Андерс Хейлсберг кстати придумал), непосредственной интеграцией с COM (Component Object Model) моделью.  
     
   - Для обоих языков компиляция выполняется в машинно-независимый и независимый от языка программирования программный код, который запускается в рамках управляемой среды выполнения.   
     
   -Концепция JVM (Java Virtual Machine) очень подобна подсистеме CLR (Common Language Runtime) языка C#.  
     
   -Сборщик мусора.  
     
   -Не существует глобальных функций или констант, все они должны принадлежать какому-либо классу.  
     
   -С# и Java не используют концепцию заголовочных файлов.  
     
   -Оба языка поддреживают внутренние классы.
2. **Почему Java является платформой, а не языком программирования?**

Java как платформа - это программное обеспечение, представляющее собой рабочую среду для работы программ, написанных на Java (и не только). Она состоит из Java API и Java виртуальной машины (JVM).

1. **Перечислите основные категории программ Java.**

- Самостоятельные программы (назовём их приложения Java), работающие независимо на локальном компьютере.

- Апплеты (applets), работающие в Internet.

1. **Какие существуют IDE для разработки программ?**- NetBeans  
   - IntelliJ IDEA  
   - Eclipse  
   - JDeveloper   
   - Android Studio  
   - DrJava  
   - MyEclipse  
   - Java N-IDE