

Вопросы по теме Особенности Java

1. В чем особенность языка Java? Почему он такой распространенный?
2. Какие преимущества есть у языка Java перед другими языками программирования?
3. Что необходимо для запуска Вашего java-приложения на устройстве пользователя?
4. За что отвечает JVM?
5. Что такое байткод?
6. Что такое JDK и JRE?
7. Зачем нужна JIT-компиляция?

Теория

Особенности языка Java:

1. ООП
2. Кроссплатформенность (байткод + JVM)
3. Многопоточность
4. Строгая, статическая, явная типизация
5. Безопасность

Типизация – это особенность работы языка программирования с различными типами переменных.

Строгая (сильная) типизация – не позволяет смешивать несовместимые типы данных. Например, в JavaScript можно сложить `int + boolean` или `String + int`, в Java такого делать нельзя. Вы можете сказать, как так? Мы же можем спокойно смешивать `int + char`, в то время как `int` это число, а `char` - символ. Дело в том, что в `char` хранится код символа в виде числового значения и с этим кодом можно производить арифметические операции.

Статическая типизация – проверка типов данных производится на стадии компиляции.

```
int a = 1;
```

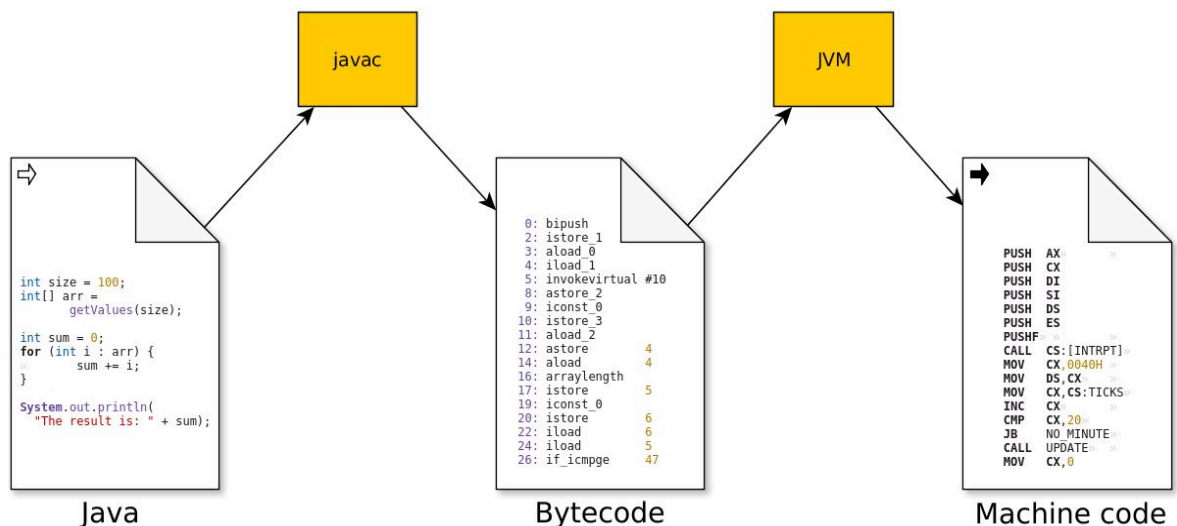
```
a = 1.1; // ошибка, присваиваем переменной типа int значение типа double
```

Явная типизация – нужно явно указывать, каким типом данных будет создаваемая переменная.

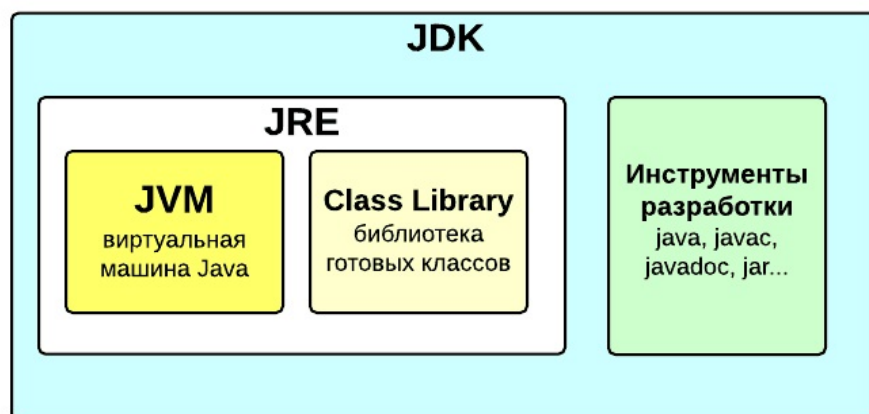
```
int a = 1;
```

ООП – Java использует объектно-ориентированную методологию программирования.

Система безопасности - 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.



Байткод - промежуточный код между кодом на языке Java и машинным кодом. Java код преобразуется в специальный байткод, который затем с помощью Java-машины (JVM) интерпретируется в понятный для текущей машины машинный код. Поэтому программа, написанная на Java, может работать на любой компьютерной архитектуре, для которой существует реализация виртуальной JVM.



JDK - комплект разработчика приложений, включающий в себя компилятор Java, стандартные библиотеки классов Java, примеры, документацию, различные утилиты и исполнительную систему Java (JRE).

JRE - минимальная (без компилятора и других средств разработки) реализация виртуальной машины, необходимая для исполнения скомпилированного Java-приложения. Состоит из виртуальной машины (JVM) и библиотеки Java классов.

JVM - виртуальная машина, которая запускает скомпилированный Java байткод. JVM не распознает исходный код Java. Поэтому сначала компилятор компилирует файлы .java в файлы с байткодом .class.

xxx.java → xxx.class

Загрузчик классов (ClassLoader) – часть JVM, которая загружает Java классы в память и делает их доступными для выполнения. Класс будет загружен только тогда, когда понадобится приложению. Существует 3 загрузчика классов:

1. Bootstrap classloader - написан на C++. Содержит все основные библиотеки.
2. Extension classloader - загружает классы расширений при запуске из папки jre/lib/ext.
3. System classloader - загружает классы, которые определены в переменной среды окружения CLASSPATH.

JIT-компиляция – технология, при которой компиляция программы происходит во время ее выполнения, а также происходит сохранение и переиспользование повторяющихся участков кода. Таким образом увеличивается производительность за счет увеличения потребления памяти и затрат времени на компиляцию.