# Чем отличаются Java приложения и Java апплеты:

Java-апплет — прикладная программа, в то время как Java приложение это самостоятельное приложение, которое можно запустить в системе. Для обоих необходима виртуальная Java машина

# Какие основные части должны присутствовать в каждой Java-программе, их функции (назначение)?

Все Java программы содержат в себе 4 разновидности:

Классы – модель для создания объектов. Классы содержат в себе атрибуты (поля и методы)

Методы – функции работающие с объектами и находящимися в составе класса

Переменные – имя которое ссылается на область памяти в которой храниться определенные значения определенные значения

Пакеты – содержат в себе классы и помогают компилятору найти нужные классы для компиляции программы

# Что такое первичный класс приложения? Какой обязательный метод он должен содержать?

Первичный класс – класс, носящий название файла. Именно с первичного класса начинается работа приложения. Первичный класс обязательно должен содержать метод main так как данный метод является точкой входа в программу.

# Какие существуют виды переменных java, чем они отличаются друг от друга?

Переменные в Java делятся на 2 категории:

Примитивные типы – встроенные типы в яп, имеют определенный размер. Память для них выделяется заранее

Ссылочные типы – типы определенные пользователем или массивы. Память для них выделяется во время работы

# Какие примитивные типы определены в Java, особенности булевого типа.

В Java существуют переменные следующего типа:

Целочисленные типы: byte(1), short(2), int(4) long(8).

Дробные типы: float(4), double(8).

Логический тип: boolean(1).

Символьный тип: char(2).

Булевый тип может принимать только два значения (true или false)

# Что называют процессом реализации ссылочного типа

Реализация ссылочного типа – выделение памяти для переменной ссылочного типа с помощью оператора new

# Что делает конструктор класса? Должен ли он обязательно явно присутствовать в объявлении класса?

Конструктор – метод имеющий такое же название как и класс, выполняет определенные действия во время создания объекта.

Конструктор можно не объявлять в классе (тогда буде использоваться конструктор по умолчанию)

# Какие существуют виды ссылочных типов?

Существуют виды ссылочных типов:

Типы определенные пользователем

Тип String

Типы массива

# Что такое типы, определенные пользователем?

Типы созданные самим программистом и которые не входят в стандартный набор типов

# Что такое стандартные типы определённые пользователям

# чем массивы Java отличаются от массивов других языков их преимущества

Массивы в Java имеют определенный тип, массивы не изменяемые, могут быть многомерными и инициализируются с помощью {}

# как переменные различных видов передаются в качестве параметров методам

Переменные ссылочного типа передаются по ссылке а переменные примитивных типов передаются по значению (создается новый объект)

# как ведут себя строковые переменные при передаче их в качестве параметра в Java

Строковые переменные передаются по ссылке в метод. В метод передается не объект а ссылка

# Что такое совмещение методов

это механизм, позволяющий определить несколько методов с одним именем внутри одного класса, но с разными параметрами.

# Что такое элементы класса и элементы экземпляра класса чем они отличаются друг от друга как нужно указать что переменные элементы является элементом класса а не экземпляра

Элементы класса можно определить с помощью модификатора static. Если элемент не имеет модификатор static то он является элементом экземпляра класса

# Для чего используется модификаторы доступа Какие существуют модификаторы доступа как они ограничивают доступ к элементам

Модификаторы доступа в Java используются для определения уровня доступа к классам, методам, полям и конструкторам.

Существуют следующие модификаторы:

Private - Данные атрибуты могут использоватся только в классе

Protected- Данные атрибуты могут использоватся только в классе или в классах производным от данного.

Public - Данные атрибуты доступны в любой части программы

Default - Элементы с default доступом доступны только внутри того же пакета.

# что позволяет делать процесс наследования

Процесс наследования позволяет повторно использовать код, создать иерархию классов, использовать полиморфизм, расширять и специализировать базовый класс, поддерживать инкапсуляцию.

# Что такое суперкласс и подкласс

Суперкласс – класс от которого наследуются другие классы, подкласс – класс наследующийся от других классов.

# Что такое повторное использование кода

Повторное использование кода – использование написанного ранее кода

# Какие заранее определённые переменные содержат каждый класс Java/ Что можно сделать при помощи переменной This Что можно сделать при помощи переменной супер

Каждый класс java содержит следующие заранее определенные переменные:

This – данная переменная ссылается на текущий объект класса. С ее помощью можно ссылаться на элементы текущего объекта класса

Super – переменная ссылающаяся на объект суперкласса. С ее помощью можно ссылаться на элементы объекта суперкласса

# Что такое скрытие переменной затемнения переменной и замещения метода

Скрытие переменной - это когда переменная в подклассе имеет то же имя, что и переменная в суперклассе (из-за видимости нельзя обратится к базовой)

Затемнение переменной - это когда переменная в подклассе имеет то же имя, что и переменная в суперклассе, но может обратиться к переменной суперкласса через ключевое слово "super".

Замещение метода - это когда подкласс переопределяет метод суперкласса с тем же именем, сигнатурой и возвращаемым типом.

# Как импортировать классы из пакетов

Чтобы импортировать классы из пакетов в Java, используйте ключевое слово "import".

# Как использовать явные ссылки на классы из пакетов

Если вы хотите использовать класс из пакета без импорта, вы можете использовать полное имя класса, включая имя пакета. Это называется явной ссылкой на класс.

# Как добавлять класс в пакет

В Java, чтобы добавить класс в пакет, необходимо указать имя пакета в первой строке файла класса, перед объявлением класса. Например, чтобы добавить класс "MyClass" в пакет "com.example", файл класса должен выглядеть следующим образом:

package com.example;

public class MyClass {

// класс реализация здесь

}