1. Пример инкапсуляции, наследования, полиморфизма в Java
2. Модификаторы доступа: default, private, protected, public
3. Абстрактные классы (extends), интерфейсы (implements) – дефолтные методы в интерфейсах (?).
4. final методы, поля, классы.
5. static методы, поля, классы (вложенные).
6. Порядок инициализации объектов
7. Какие типы в данных можно использовать в улучшенном операторе for после двоеточия?

**for (String element: stringList) {}** коллекции, массивы

1. Что напечатает такая программа?

public class MM {  
 public static void main(String... args) {  
 Integer i;  
 System.*out*.println(i);  
 }  
}

1. Передача объектов в методы; ссылки  
   Чему будет равен ‘a’ после вызова meth(a):

package org.test.sample;  
  
public class MM {  
  
 String data = "sample data";  
  
 public static void main(String... args) {  
 MM mmValue = new MM();  
 int intValue = 42;  
 mmValue.meth(mmValue, intValue);  
  
 System.*out*.println(mmValue); // Что выведется здесь?  
 System.*out*.println(intValue); // Что выведется здесь?  
 }  
  
 private void meth(MM a, int b) {  
 a = null;  
 b = 100;  
 }  
  
 @Override  
 public String toString() {  
 return "MM{" +  
 "data='" + data + '\'' +  
 '}';  
 }  
}

1. Java Exceptions: checked vs unchecked
2. NPE
3. Try-catch-finally  
   Что выведет:

public class MM {  
  
 public static void main(String... args) {  
 int num = 0;  
 try {  
 num = 1;  
 throw new Exception();  
 } catch (Exception e) {  
 num = 2;  
 } finally {  
 num = 3;  
 }  
 System.*out*.println(num);  
 }  
}

Что вернет метод:

public class MM {  
  
 public static void main(String... args) {  
 System.*out*.println(*method*());  
 }  
  
 public static int method() {  
 try {  
 return 1;  
 } finally {  
 return 2;  
 }  
 }  
  
}

1. Сигнатуры методов, перегрузка методов
2. Коллекции: какие есть имплементации?
3. Что выведет эта программа?

public class MM {  
  
 private String brand;  
 private boolean empty;  
  
 public static void main(String... args) {  
 MM bottle = new MM();  
 if (!bottle.empty) {  
 System.*out*.println("brand = " + bottle.brand);  
 }  
 }  
}

1. Что быстрее ArrayList vs LinkedList?
2. equals(), hashCode()
3. Коллизии HashMap (?) – иногда спрашивает ☺
4. Многопоточность: synchronized на методе vs блок synchronized. Как сделать дедлок?  
   volatile.  
   Как треды создать и запустить? Классы Runnable и Callable?
5. Thread STATEs

Spring:

Для чего нужен Component Scan?

В чём разница между @Bean и @Component?

Можем ли мы применить @Autowired с не сеттерами и не конструкторами методами?

БД:

**What is a primary key?**

**What is a unique key?**

**What is a foreign key?**

**What is a join?**

**What are the types of join and explain each?**

* **Inner Join.**

Inner join return rows when there is at least one match of rows between the tables.

* **Right Join.**

Right join return rows which are common between the tables and all rows of Right hand side table. Simply, it returns all the rows from the right hand side table even though there are no matches in the left hand side table.

* **Left Join.**

Left join return rows which are common between the tables and all rows of Left hand side table. Simply, it returns all the rows from Left hand side table even though there are no matches in the Right hand side table.

* **Full Join.**

Full join return rows when there are matching rows in any one of the tables. This means, it returns all the rows from the left hand side table and all the rows from the right hand side table.

**What is normalization?**

**What is Denormalization.**

**What is a View?**

**What is an Index?**

**What is data Integrity?**

1. Join: inner/outer join?
2. Having

Unix:

1. netstat -lpn : что выведет? Что значит каждый флаг?
2. cat “file.txt” | cat – что выведет?
3. grep
4. Permissions: chmod 755 file.txt  
   пермишн на выполнение на директорию?