Java
Interview Questions

Date
Page

① What does the static keyword mean in java? Explain the difference between static and non-static methods

→ Static keyword in java: The static keyword in java is used to defined class-level members that are shared among all instance of the class.
- A static variable or method belongs to the class itself rather than to any specific instance.

Static Method
① Can be called without creating an instance of the class.
② Can only directly access the other static members (variable and methods) of class
③ Cannot use instance variable or instance methods directly.

Non-static method
① Required an instance of the class to be called
② Can access both static and non-static members of the class
③ Typically operate on instance variables.

② What is the role of the static keyword in the context of memory management?

→ Memory management and static keyword:
Static variables are allocated memory only once when the class is loaded, and they persist for the lifetime of the application

- Non-static variables are allocated memory when an instance of the class is created and are deallocated when the instance is destroyed
- This means that static variables are shared across all instances, while non-static variables are unique to each instance

(3) Can static methods be overloaded and overriden in java? How are static variables shared across multiple instance of a class?

→ Static methods
- Overloading: Yes. Static methods can be overloaded. Overloading refers to defining multiple methods with the same name but different parameter lists within the same class.
- Overriding: No, static methods cannot be overridden. Static methods are bound to the class, not to instance. However, you can hide a static method by defining a static method with the same name in a subclass.

Static Variables:
- Static variables are shared across all instance of a class. All instance of the class have access to the same static variable, meaning that if one instance changes value of a static variable, the change is reflected across all instance

(4) What is the significance of the final keyword in java?

→ Final keyword in java:

Final Variable:
- A variable declared as final can only a initialized once
- Once assigned, its value cannot be changed
Final method:
- A final method declared as final cannot be overridden by subclasses.
- This helps prevents altering the behavior of the method in subclasses.
Final class:
- A class declared as final cannot be subclassed. This ensures that the class's implementation cannot be extended or modified by inheritance

(5) What are narrowing and widening conversion in java?

→ ① Narrowing Conversion:
Converting a larger data type to a smaller data type. This can potentially result in data loss.
Example: Converting ~~int to~~ double to int
② Widening Conversion:
Converting a smaller data types to a larger data type. This is usually safe and does not result in data loss.
Example: Converting int to double

(6) Provide examples of narrowing and widening conversion between primitive data types:

→ Widening Conversion:
Int to long
float to double

Narrowing Conversion
double to int
long to short

(7) How does java handle potential loss of precision during narrowing Conversion?

→ Java handles potential loss of precision during narrowing conversion by explicitly requiring the programmer to perform the Conversion using a cast. This cast indicates that the programmer is aware of the potential loss of precision and accepts it.

Example:

```
double d = 9.78;
int i = (int) d;
```

(8) Explain the concept of automatic widening Conversion in java.

→ Automatic Widening Conversion:

Occurs when Converting a smaller data type to a larger data types. Java performs this Conversion automatically, without the need for an explicit cast.

Example:

Assigning an int value to a double variable is handled automatically by java

```
int num = 10;
double d = num;
```

(9) What are the implications of narrowing and widening Conversion on type compatibility and data loss!

→ Widening Conversion:
Generally safe and does not result in data loss.
Type compatibility is maintained since a larger type can represent all possible values of a smaller type

Narrowing Conversion:
Can result in data loss or precision loss.
Type compatibility may be lost since a smaller type may not be able to represent all values of a larger type
Requires explicits casting to indicates that the programmer acknowledges and accepts the risk of data loss.