

SP Tutorials
Questions on Core Java Part I

Q1 What is the fundamental concept upon which Java is built?

A The class forms the basis for object-oriented programming in Java. It is the logical construct upon which the entire language is built because it defines the shape and nature of an object.

Q2 What is meant by instance variables?

A The variables defined within a class are called instance variables.

Q3 Why are instance variables so named?

A Variables defined within a class are called instance variables because each instance of the class (i.e., each object of the class) contains its own copy of these variables.

Q4 Is it compulsory for all classes to have a main() method?

A Java classes do not need to have a main method. We specify the main method in a class only if that class is the starting point of our program.

Q5 What role does the new operator play in object creation?

A The new operator dynamically allocates memory for an object and returns a reference to it. This reference is the address in memory of the object allocated by new. This reference is then stored in the variable.

Q6 Differentiate between objects and classes.

A 1) Class creates a logic framework that defines the relationship amongst its members. Object is an instance of a class.

2) A class is created using the keyword class.
An object is created using the keyword new.

3) Creating a class does not allocate any memory, i.e., it is a logical entity.
Memory is allocated when an object is instantiated, i.e., it is a physical entity.

Q7 Differentiate between parameter and argument.

A A parameter is a variable defined by a method that receives a value when the method is invoked.

An argument is a value that is passed to a method when that method is invoked.

For example,

```
int square(int number){  
    return number*number;  
}
```

Here, number is a parameter. A call to square() such as square(3) passes 3 as an argument.

Q8 What is a constructor?

A Constructor is a special member function that initializes an object immediately upon creation. It has the same name as the class in which it resides. A constructor has no return type, not even void.

Q9 Why do constructors have no return type?

A Constructors have no return type, not even void. This is because the implicit return type of a class' constructor is the class type itself.

Q10 What do you know about default constructors?

A When we do not explicitly define a constructor for a class, Java provides us with a default constructor. This initializes all instance variables to a default initial value, which maybe 0 for integers, 0.0 for floats, null for Strings and so on. Once we define our own constructor, the default constructor is no longer used.

Q11 What is the this keyword?

A To allow a method to refer to the object that invoked it, Java defines the this keyword. It can be used inside any method to refer to the current object.

Q12 What is instance variable hiding? How is the this keyword used to overcome instance variable hiding?

A It is illegal in Java to declare two local variables with the same name inside the same or enclosing scopes. However, we can have local variables, including parameters to methods, which overlap with the names of the class' instance variables. When a local variable has the same name as an instance variable, the local variable hides the instance variable. This is known as instance variable hiding.

Since the this keyword allows us to refer directly to the object, it is used to resolve any namespace collisions that might occur between instance variables and local variables.