Constructor

Assignment

Question 1- What is a constructor?

Ans-

A constructor is a special method that is used to initialize objects when they are created.

It has the same name as the class and is called automatically when an object is created . The constructor is used to set the initial values of the object’s properties or instance variables.

Question 2- What is constructor chaining?

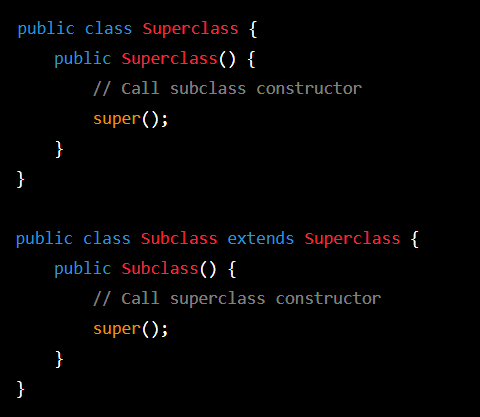
Ans-

Constructor chaining refers to the process of calling one constructor from another constructor within the same class or from a subclass. This is done using the keyword “this” or “super” to refer to the constructor being called. By chaining constructors, a class can reuse code from another constructor, reducing duplication and promoting code reuse. This can be especially useful when dealing with complex class hierarchies or when working with classes that have multiple constructors with different parameters.

Question 3- Can we call a subclass constructor from a superclass constructor?

Ans

Yes, we can call a subclass constructor from a superclass constructor, we use the following syntax:



In the above example, the ‘Subclass’ constructor calls the ‘Superclass’ constructor using ‘super()’. This ensures that the ‘Superclass’ is properly initialized before the ‘Subclass’ is initialized.

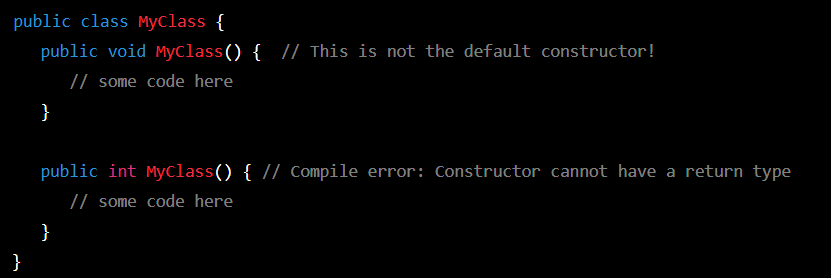
Question 4- What happens if you keep a return type for the default constructor?

Ans-

The default constructor is a constructor that is automatically generated by the compiler if we don’t define any constructor in your class. It is used to initialize the object with default values.

The default constructor doesn’t have a return type, not even ‘void’, and it returns a reference to the object being constructed automatically. so , if we try to define a return type for the default constructor in Java, we will get a compilation error.

Here is an example of what happens if you try to define a return type for the default constructor:



As you can see, the second constructor definition with an ‘int’ return type is invalid, and the compiler will report an error when you try to compile the code.

Question 5- What is No-arg constructor?

Ans-

A No-arg constructor in java is a constructor that takes no arguments or parameters. It is also known as a default constructor, as it is provided by default if no other constructors are defined in a Java class. The purpose of a no-arg constructor is to initialize the object’s instance variables with default values. If a Java class does not define any constructor explicitly, the Java compiler automatically creates a no-arg constructor for the class.

Question 6- How is a No-argument constructor different from the default construction?

Ans-

A No-argument constructor is a construction that is explicitly defined in a class and takes no parameters, while a default constructor is a no-argument constructor that is automatically generated by the compiler if no other constructor are explicitly defined in the class. The behavior A no-argument constructor and a default constructor can be identical if the no-argument constructor defined in the class is empty.

Question 7- When do we need constructor overloading?

Ans-

Constructor overloading in Java allows us to create multiple constructor with different parameters lists in a class. This can be useful in several situation, including:

1. Creating objects with different initial values: if a class has multiple fields, each with a different default value, we can provide different constructors that allows the user to specify which values.
2. Providing flexibility to users: By providing multiple constructors with different parameter lists, we give users of our class the flexibility to choose the constructor that best fits their needs.
3. Implementing polymorphism:constructor overloading can be used in conjunction with method overloading to implement polymorphism in Java. By providing multiple constructors with different parameters lists, we can create objects of different types, each with its own set of behaviors.

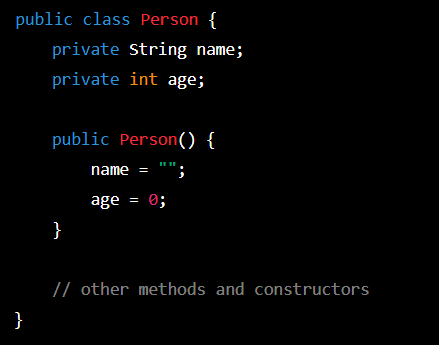
Overall, constructor overloading allows us to provide more flexibility and functionality to our Java classes, and is often used in conjunction with other-oriented programming concepts like encapsulation and inheritance.

Question 8- What is Default constructor Explain with an Example

Ans-

A default constructor is a special type of constructor that is automatically generated by the compiler if a class does not have any other constructor defined. It has no parameters and its purpose is to initialize the instance variables of the class with default values.

Here is an example of a class with a default constructor:



In the above example, the ‘person’ class has a default constructor which initializes the ‘name’ and ‘age’ instance variables with default values of an empty string and zero, respectively. This constructor can be called using the ‘new’ keyword to create a new instance of the ‘Person’ class without providing any arguments, like this:



This will create a new ‘Person’ object with default values for its ‘name’ and ‘age’ properties.