Q1. What is an interface in java?

Ans. An interface in java is a mechanism that is used to achieve complete abstraction. It is basically a kind of class that contains only constants and abstract methods.

Q2. Which modifiers are allowed for methods in an interface? Explain with an example.

Ans. Only abstract and public modifiers are allowed for methods in interfaces.

Q3. What is the use of interface in java?

Ans. There are many reasons to use interfaces in java as follows:
> An interface is used to achieve full abstraction.
> Using interfaces is the best way to expose our project’s API to some other project.
> Programmers use interfaces to customize features of software differently for different objects.
> By using interfaces, we can achieve the functionality of multiple inheritance.

Q4. What is the difference between abstract class and interface in java?

Ans.

Abstract class	Interface
Abstract class can have abstract and non-abstract methods	Interface can have only abstract methods. Since JAVA 8, it can have default and static methods also.
It doesn’t support multiple inheritance	It supports multiple inheritance
It can have final, non-final, static and non-static variables	It can have only final and static variables
It can provide the implementation of interface	It can not provide the implementation of abstract class
The abstract keyword is used to declare abstract class	The interface keyword is used to declare the interface
An abstract class can extend another java class and implement multiple java interfaces	An interface can extends another java interface only.
An abstract class can be extended using the keyword “extends”	An interface can be implemented using keyword “implements”
A java abstract class can have class members like private, protected etc.	Members of a java interface are public by default.

Example:
public abstract class Shape
{
public abstract void draw( );
}
	Example:
	public interface Drawable{
	Void draw( );
	}