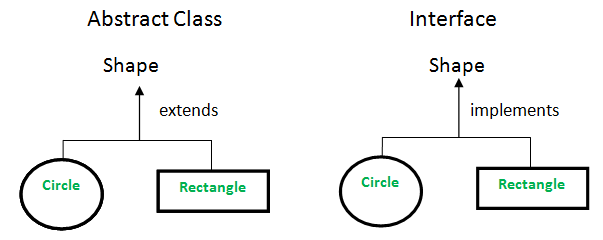
**Difference between Abstract Class and Interface in Java**

* Difficulty Level : [Easy](https://www.geeksforgeeks.org/easy/)
* Last Updated : 28 Feb, 2022

As we know that abstraction refers to hiding the internal implementation of the feature and only showing the functionality to the users. i.e. what it works (showing), how it works (hiding). Both [abstract class](https://www.geeksforgeeks.org/abstract-classes-in-java/) and [interface](https://www.geeksforgeeks.org/interfaces-in-java/) are used for abstraction, henceforth Interface and Abstract Class are required prerequisites



**Abstract class vs Interface**

* **Type of methods:** Interface can have only abstract methods. An abstract class can have abstract and non-abstract methods. From Java 8, it can have default and static methods also.
* **Final Variables:** Variables declared in a Java interface are by default final. An abstract class may contain non-final variables.
* **Type of variables:**Abstract class can have final, non-final, static and non-static variables. The interface has only static and final variables.
* **Implementation:** Abstract class can provide the implementation of the interface. Interface can’t provide the implementation of an abstract class.
* **Inheritance vs Abstraction:** A Java interface can be implemented using the keyword “implements” and an abstract class can be extended using the keyword “extends”.
* **Multiple implementations:** An interface can extend another Java interface only, an abstract class can extend another Java class and implement multiple Java interfaces.
* **Accessibility of Data Members:** Members of a Java interface are public by default. A Java abstract class can have class members like private, protected, etc.