

- Interfaces are used for defining set of rules to the implementing classes
- We create method definitions in interfaces. Before java 8 we do not have any facility to add method body in interface method.
- A method without having body is called abstract method.
- By default all methods are abstract in interface.
- We cannot create instance for interface.
- we must use "implements" keyword to implement a class for interface
- We must develop methods with body in implementing classes.
- If a class is implementing interface it must write body for the interface methods or it must be abstract class
- A class can implement multiple interfaces
- We can create instance for class and store it as interface reference
- In that case only the methods defined in interface can be accessible
- If you create instance for class and take it as the same class reference then you can access all methods of that class
- We can implement methods with body inside of interface from java 8 onwards.
- For that the method must be static or default.
- If a method is static or default then the implementing class may not need to implement the body for that method.
- When a class implementing multiple interfaces then we can create instance using any interface or we can create instance using one interface and cast to other interface
- `Interfacename1 i1 = new classname();`
- `Interfacename2 i2 = (Interfacename2) i1;`
- In this case only those methods defined in respective interface will be accessed.
- Create Interface: Right click on project -> new -> Interface
- `public class className implements InterfaceName{ }`