Contents

[**Java Overview** 1](#_Toc509958054)

[**Access Modifiers** 1](#_Toc509958055)

|  |
| --- |
| **Java Overview** |
| **Access Modifiers** |
|  |
| What do you mean by Modifier? |
| Modifiers are keywords that you add to class, method, constructor, variables to change their meanings or way of working.  Java language has a wide variety of modifiers, including the following:   * Access Modifiers * Non Access Modifiers |
| What do you mean by Access Modifier? |
| Modifiers are keywords that you add to class, method, constructor, variables to set access levels.  The four access levels are:   * Private * Default(No modifier mentioned) * Protected * Public  |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Visibility** | **Private**  **(within Class)** | **Default**  **(within Package)** | **Protected**  **(within Same Package and Sub classes of different package)** | **Public**  **(Everywhere within the Java project)** | | **Same Class** | **Accessible** | **Accessible** | **Accessible** | **Accessible** | | **Same Package** |  | **Accessible** | **Accessible** | **Accessible** | | **Different Package - Sub Class** |  |  | **Accessible** | **Accessible** | | **Different Package - Non Sub Class** |  |  |  | **Accessible** |  * Interface:  1. Can only be public or default. 2. Variables declared in interface can be public, static and final. Even if we don’t write in the code, it will be considered public, static and final by default. 3. All the interface methods can be and by default abstract and public. Even if we don’t write in the code.      * Inheritance:  1. Methods declared public in a superclass also must be public in all subclasses. 2. Methods declared protected in a superclass must either be protected or public in subclasses; they cannot be private. 3. Methods declared private are not inherited at all, so there is no rule for them. |
| Can I have a private class?  Can I have a protected class? |
| * Class:  1. Can only be public or default, unless it’s inner class. |
| Can I have a private interface?  Can I have a protected interface? |
| * Interface:  1. Can only be public or default. |
| What is the access level of interface fields if I do not append any modifier for them?  Can I have interface fields as private?  Can I have interface fields as protected? |
| * Variables declared in interface can be public, static and final. Even if we don’t write in the code, it will be considered public, static and final by default. |
| What is the access level of interface method if I do not append any modifier for them?  Can I have interface method as private?  Can I have interface method as protected? |
| * All the interface methods can be and by default abstract and public. Even if we don’t write in the code. |
| I have a Methods declared protected in a superclass can I declare it as private in subclasses? |
| * No * Inheritance:  1. Methods declared public in a superclass also must be public in all subclasses. 2. Methods declared protected in a superclass must either be protected or public in subclasses; they cannot be private. 3. Methods declared private are not inherited at all, so there is no rule for them. |
| How can I access private variables? |
| Variables that are declared private can be accessed outside the class using multiple ways, most popular are:   * 1. Using Non-private setter and getter methods are present in the class.   2. Using Constructor to set the value. |