**FINAL METHOD**

What is final method

Why final method

1. To prevent to override superclass methods in subclass.

* We **cannot override** final method instead we can inherit in subclass, but we **can overload** final method.
* We can override non final method of superclass as final method in subclass. But vice versa gives an error.
* We can declare main method in subclass as final. But we can’t override in subclass. So to execute subclass static and non static members, we have to use static block in sub class.

|  |  |
| --- | --- |
| **Private methods** | **Final methods** |
| Cannot override to subclass, if override, compiler won’t throw error, but it will be treated as separate method. | Cannot override to subclass, if override, compiler throw error |
| These **will not inherit to subclass** | **inherit to subclass** |
| **Can’t call private methods in sub class. Gives CE** | **Call final methods in subclass** |

**FINAL CLASS**

What is final class?

A class declared with **final keyword**.

When to declare?

1. If we don’t want to extend all functionality
2. If we don’t want to override all methods in subclass

Rules:

1. In final class, all members of class are not final, we have to explicitly declare them as final.
2. We cannot override non-final member because we cannot create sub class.
3. We can change non final variable values.
4. We can instantiate final class.

**How many ways we can prevent a class to inherit?**

1. By declaring class as final : here u can create a object of class
2. By declaring private constructor : here you can neither create a object nor create a subclass.