* **modifier:** Modifiers are java keywords that you add to those definitions to change their meanings. Java language has a wide variety of modifiers, which is classified in two categories –
* [**Access Modifiers**](https://www.tutorialspoint.com/java/java_access_modifiers.htm) **(controls the access level):** public, private, default and protected.
* [**Non-Access Modifiers**](https://www.tutorialspoint.com/java/java_nonaccess_modifiers.htm) **(do not control access level, but provides other functionality):** final, static, abstract, transient, synchronized and volatile.
* **Instance method (non-static method):** Instance method are methods which require an object of its class to be created before it can be called. To call a instance method, we have to create an object of the class in which the method is defined.

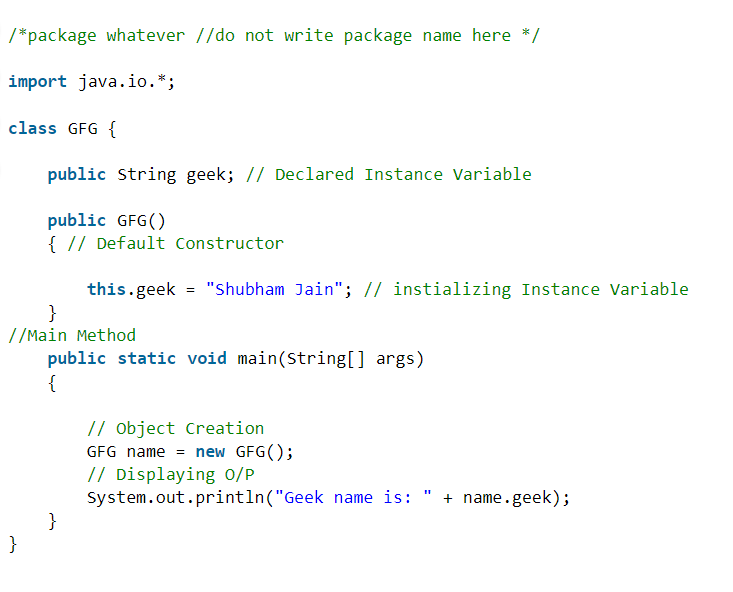
1. All methods except static methods (i.e., methods having static keyword) are instance methods.
2. Instance methods not have static keyword in it. It declared without static keyword.

* **Static method**: Static methods are the methods in java that can be called without creating an object of class.
* **Variable in java:** There are 3 types of variable in java-

1. Local var.
2. Instance var.
3. Static var.

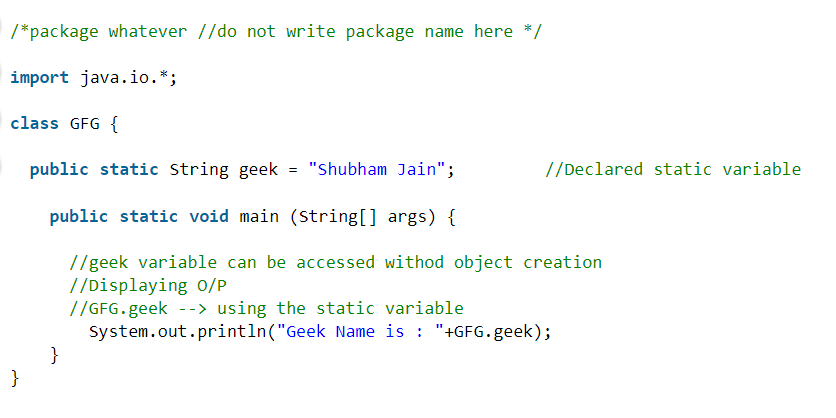
* **Instance variable (non-static variable / member variable/ instance field / field):**

1. It declared in a class outside any method, constructor, or block.
2. Variables declared without using static keyword are instance variables.
3. These variables are created when an object of the class is created and destroyed when the object is destroyed.
4. Initialization of instance variable is not mandatory. Its default value is 0.
5. Declaration-



* **Static variable (class variable / static field / field) :**

1. These variables are declared similarly as instance variables. The difference is that static variables are declared using the static keyword within a class outside any method constructor or block.
2. Variables declared using static keyword are static variables.
3. Initialization of static variable is not mandatory. Its default value is 0.
4. Declaration-



* **Local variable:**

1. Local variables are declared in methods, constructors, or blocks.
2. Local variables are created when the method, constructor or block is entered and the variable will be destroyed once it exits the method, constructor, or block.
3. Access modifiers cannot be used for local variables.
4. Local variables are visible only within the declared method, constructor, or block.
5. There is no default value for local variables, so local variables should be declared and an initial value should be assigned before the first use.