**Use of “this” keyword**

The keyword **this** identifies a special type of reserved word in java. The keyword **this** means **“for the instance of the class”** or within a method called for an object it means for “myself”.

* When a non-static method is called, the **this** reference is a constant reference to the object for which the method is called.
* One cannot declare the this reference or make assignments to it. Attempts to reassign a value to **this** reference in a non-static method will result a compilation error (It will state something like: **Cannot assign a value to final variable this**)
* For the static method, **this** reference does not exist.
* The this reference is passed as an extra argument (hidden) to all non-static methods. Member data is addressed by evaluating the expression this member-name (implicitly done).
* But sometimes, we need to use the keyword this explicitly, to create difference between an instance of a class’s data members and normal variable. Check the following example:-

class Person

{

…………………………………………..

…………………………………………..//implementations of other methods

public Person(final String Name,final int Age,final char Gender)

//parameterized constructor of the base class

{

this.Name=Name;

this.Age=Age;

this.Gender=Gender;

}

…………………………………………..

…………………………………………..//implementations of other methods

}

Her, we use the **this** keyword to make difference between a class instance’s data members name and the parameters name in a non-static method of a class.

**Another usage of this keyword:**

The method call this() refers to the default constructor (i.e. it invokes a class’s default constructor) when it need to instantiated through another constructor of through another non-static method of the class.