**Day 11**

**Constructors in Java**

* A constructor in Java is a special method that is used to initialize objects.
* The constructor is called when an object of a class is created.
* JVM will automatically provide constructor internally when we specifically don’t create the constructor.

**Rules for Constructions**

* The name of the constructor must be same as that of class name.
* Constructor must not have an explicit return type.
* Constructor cannot be abstract, static, final and synchronized.
* While creating constructors, access modifiers should be specified.

**Types of Constructors**

There are two types of constructors in Java. They are:

1. **Default Constructors**
2. **Parametrized Constructors**

**Default Constructors**

This constructor is implemented by default by the Java compiler if there is no explicit constructor implemented by the user for the class.

Syntax

<class Sample> ()

{

//statements

}

Example

class Sample

{

Sample()

{

System.out.println(“Constructor created”);

}

public static void main(String args[])

{

Sample obj = new Sample();

}

}

**Why Constructors are used ???**

Default constructors will give default values. That is, for int 0, float 0.0, string null.

class Student

{

int id;

String name;

void display()

{

System.out.println(id+ “ ” +name);

}

public static void main(String args[])

{

Student ob1 = new Student();

Student ob2 = new Student();

ob1.display();

ob2.display();

}

}

This output will be

0 null // since default constructor is being invoked.

**Parametrized Constructors**

A parameterized constructor accepts parameters with which you can initialize the instance variables. Using parameterized constructor, you can initialize the class variables dynamically at the time of instantiating the class with distinct values.

Class Student

{

int id;

String name;

Student (int i, String n)

{

Id =i;

name = n;

}

void display()

{

System.out.println(id+” “+name);

}

public static void main(String args[])

{

Student ob1 = new Student(12, “Anoopa”);

Student ob2 = new Student(13, “Rohit”);

ob1.display();

ob2.display();

}

}

**Constructor Overloading**

class Student

{

int id;

String name;

int age;

Student(int i, String n)

{

id =I;

name=n;

}

Student (int i, String n, int a)

{

Id =i;

name=n;

age=a;

}

Void display()

{

System.out.println(id+” “+name);

}

public static void main(String args[])

{

Student ob1 = new Student(12,”Anoop”);

Student ob2 = new Student(15,”Jitty”,20);

ob1.display();

ob2.display();

}

}