**Class in Java**

A class is a group of objects which have common properties. It is a template or blueprint from which objects are created. It is a logical entity. It can't be physical.

A class in Java can contain:

• fields

• methods

• constructors

• blocks

• nested class and interface

Syntax to declare a class:

class <class\_name>{

field;

method;

}

**Instance variable in Java**

A variable which is created inside the class but outside the method, is known as instance variable. Instance variable doesn't get memory at compile time. It gets memory at run time when object(instance) is created. That is why, it is known as instance variable.

3 Ways to initialize object

There are 3 ways to initialize object in java.

1. By reference variable

2. By method

3. By constructor

**Object and Class Example: main function within class**

In this example, we have created a Student class that have two data members id and name. We are creating the object of the Student class by new keyword and printing the objects value.

Here, we are creating main() method inside the class through reference variables

package basics;

//main class

public class SimpleClass {

int id;//field or data member or instance variable

String name;

public static void main(String args[]){

SimpleClass s1=new SimpleClass ();//creating an object of Student

s1.id=101;

s1.name="Ram";

System.out.println(s1.id);//accessing member through reference variable

System.out.println(s1.name);

}

}

o/p

101

Ram

**How to create a project and java file**

**Steps to create project and java file.**

File → New → Java project and give project name **basics** which you want to create

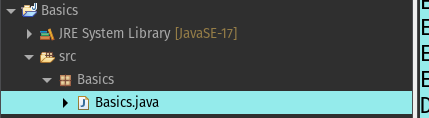
and press **don’t create module**

Click **Package explorer and select the project you have created** ( in our case it is B**asics**)

and right click on **src** folder and type **new → Package** and type **Basics**

**Right click** on the **Basics** package folder → **New → Class → Class name ( Basics.java)**

**pick** the check box **public static void main ()**

****

**Object and Class Example-initialization through reference variables:outside main class**

In real time development, we create classes and use it from another class. It is a better approach than previous one. Let's see a simple example, where we are having main() method in another class.

We can have multiple classes in different java files or single java file. If you define multiple classes in a single java source file, it is a good idea to save the file name with the class name which has main() method.

package basics;

class Student{

int id;

String name;

}

public class MainOutClass {

public static void main(String args[]){

Student s1=new Student();

s1.id=101;

s1.name="Ram";

System.out.println(s1.id);

System.out.println(s1.name);

}

}

o/p

101

Ram

**new keyword in Java**

The new keyword is used to allocate memory at run time. All objects get memory in Heap memory area.

**Object and Class Example: Initialization through method**

In this example, we are creating the two objects of Student class and initializing the value to these objects by invoking the insertRecord method. Here, we are displaying the state (data) of the objects by invoking the displaydata() method.

// out side class

class Student{

private int rollno;

private String name;

void setdata(int r, String n){

rollno=r;

name=n;

}

void dispdata(){

System.out.println(rollno+" "+name);}

}

// main class

class TestStudent{

public static void main(String args[]){

Student s1=new Student();

Student s2=new Student();

s1.setdata(111,"Karan");

s2.setdata(222,"Aryan");

s1.dispdata();

s2.dispdata();

}

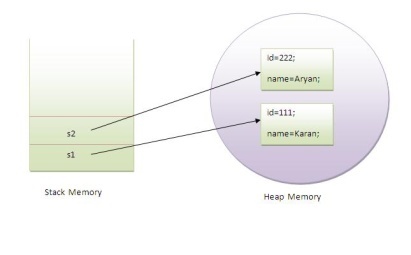
}

o/p

Output:

111 Karan

222 Aryan



**Object and Class Example: Initialization through constructor**

class Student{

int id;

String name;

// constructor

Student(int i, String n)

{

id=i;

name=n;

}

public void Display()

{

System.out.println("Empno "+id);

System.out.println("Ename "+name);

}

}

public class MainOutClass {

public static void main(String args[]){

Student s1=new Student(101,"Ram");

s1.Display();

}

}

o/p

Empno 101

Ename Ram