Method Overloading :

- **It is the process of defining more than one method with the same name for different purpose**
- **In Methods Overloading Method name must be same, list of arguments should not be same , return type may or may not be same**
- **Eg: println(int) | println(String) | println(float) …..from PrintStream class in java.io package**

Ex 1:

```
class Demo
{
      void method1()   //non static mtd
      { System.out.println("Mtd-1 wout arg "); }

      void method1(int x) //instance mtd
      {System.out.println("Mtd-1 with int : "+x); }

      void method1(int x,int y)
      { System.out.println("Mtd-1 with 2 int  "+x+"  "+y); }

   public static void main(String args[ ])
      {
               Demo d=new Demo( );
                        d.method1(10);
                                 d.method1( );
                                 d.method1(10,20);
      }

}
```

Ex 2:

```
class TestIQ
{
	void method1(byte b)
	{ System.out.println("byte : "+b); }

	void method1(short s)
	{ System.out.println("short : "+s); }

	void method1(int i)
	{ System.out.println("int : "+i); }

	public static void main(String args[])
	{
		TestIQ t=new TestIQ();
			t.method1(10);
				t.method1(100);
				t.method1(10000);
	}
}
```

Eg 3:

```
class TestIQ
{
	void method1(byte b)
	{ System.out.println("byte : "+b); }

	void method1(short s)
	{ System.out.println("short : "+s); }
```

```
    void method1(int i)
    { System.out.println("int : "+i); }

    public static void main(String args[])
    {
        TestIQ t=new TestIQ();
            t.method1((byte)10);
                t.method1((short)100);
                t.method1(10000);
    }
}
```

Ex 4:

```
class IQ
{
    public static void main(String args[ ])
    {
        System.out.println("Hello Dear ");
          IQ i=new IQ( );
          int sq=i.main(5);
        System.out.println("Result is : "+sq);
    }

    public int main(int x)  //non static or instance mtd
    { int s;
      s=x*x;
      return s;
  }
}
```

Ex :

```
class Testing
{
      void method1(float x)
      { System.out.println("Mtd-1 Float : "+x); }

      public static void main(String args[])
      { Testing t=new Testing( );
                t.method1(12);
               t.method1('A');
            //byte->short-->int-> long -> float -> double
   //                  char -> int -> long -> float -> double
      }
}
```