***Implementation of Constructor Types:***

**Constructor With No arguments:**

**code:**

package Constructor;

class main {

int i;

// constructor with no parameter

private main()

{

i = 5;

System.out.println("Constructor is called");

}

public static void main(String[] args) {

// calling the constructor with no parameter.

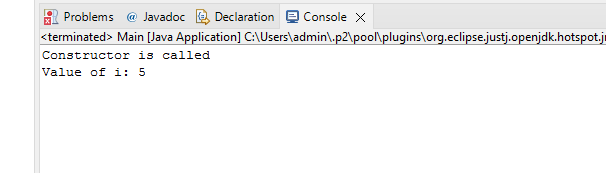
Main obj = new Main();

System.out.println("Value of i: " + obj.i);

}

}

**Output:**



**Constructors Overloading:**

**Code:**

package Constructors;

class Main {

String language;

// constructor with no parameter

Main() {

this.language = "Hello";

}

// constructor with parameter

Main(String greet) {

this.language = greet;

}

public void getName() {

System.out.println("Greeting: " + this.language);

}

public static void main(String[] args) {

// call constructor with no parameter

Main obj1 = new Main();

// call constructor with parameter

Main obj2 = new Main("Good morning!");

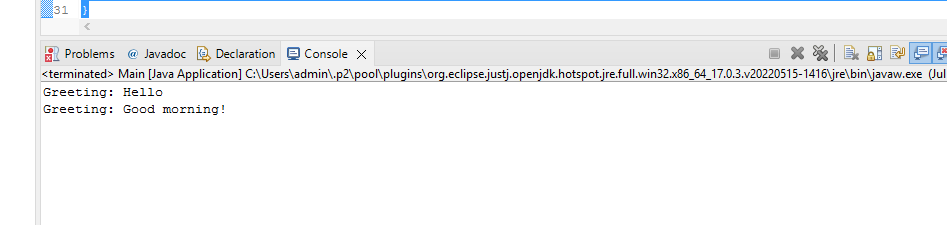
obj1.getName();

obj2.getName();

}

}

**Output:**



**Parameterized Constructor:**

Code:

package Constructor1;

class Main {

String flowers;

// constructor accepting parameters

Main(String flow) {

flowers = flow;

System.out.println( " The flower that you have choosen is "+flowers);

}

public static void main(String[] args) {

// call constructor by passing arguments

Main obj1 = new Main("Rose");

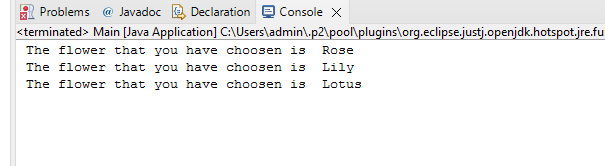
Main obj2 = new Main("Lily");

Main obj3 = new Main("Lotus");

}

}

**Output:**

****

**Default Constructor:**

**Code:**

package Constructor1;

class Main {

int a;

float b;

long c;

double d;

boolean e;

byte f;

public static void main(String[] args) {

// default constructor calling

Main obj = new Main();

System.out.println("Value of a: " + obj.a);

System.out.println("Value of b: " + obj.b);

System.out.println("Value of c: " + obj.c);

System.out.println("Value of d: " + obj.d);

System.out.println("Value of e: " + obj.e);

System.out.println("Value of f: " + obj.f);

}

}

**Output:**

