**ITM 311 Class Activity for Week 3**

**Part 1 Overloaded Constructor Demo 1**

To demonstrate the concept of Overloading, open Eclipse, copy the code below and execute the code. Make sure that you place appropriate comments statements.

When overloading a method: keep the same name as the method, but give it a different parameter list or different parameter datatype.

Place a snapshot of the output where indicated.

Source Code

//to demonstrate an overloaded constructor

**class** MyClass{

**int** x;

MyClass(){

System.***out***.println("Inside MyClass().");

x=0;

}

MyClass(**int** i) {

System.***out***.println("Inside MyClass(int).");

x=i;

}

MyClass(**double** d) {

System.***out***.println("Inside MyClass(double).");

x=(**int**) d;

}

MyClass(**int** i, **int** j){

System.***out***.println("Inside MyClass(int,int).");

x=i\*j;

}

}

**class** OverloadConsDemo{

**public** **static** **void** main(String args[]){

MyClass t1 = **new** MyClass();

MyClass t2 = **new** MyClass(88);

MyClass t3 = **new** MyClass(17.23);

MyClass t4 = **new** MyClass(2,4);

System.***out***.println("t1.x: " + t1.x);

System.***out***.println("t2.x: " + t2.x);

System.***out***.println("t3.x: " + t3.x);

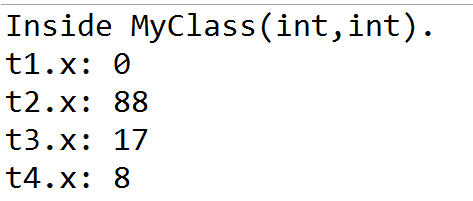
System.***out***.println("t4.x: " + t4.x);

}

}

Output

Place a Snapshot of Output for this code below this sentence.



Questions:

1. What is a constructor?
2. Why is overloading useful in Java?
3. What is a parameter?
4. Does Java use pass parameters by reference or by value?

**Part 2 Overriding Constructor Demo 2**

To demonstrate the concept of Overriding, open Eclipse, copy the code below and execute the code. DPlace a snapshot of the output where indicated below.

When overriding a method, preferences for the derived class change, but keep the same method.

Source Code

//Demo for method overriding

**class** Override {

**void** ovlDemo(){

System.***out***.println("No parameters");

}

//Override ovlDemo for one integer parameter.

**void** ovlDemo(**int** a) {

System.***out***.println("One parameter: " + a);

}

//Override ovlDemo for two integer parameters.

**int** ovlDemo(**int** a, **int** b){

System.***out***.println("Two parameters: " + a + " " + b);

**return** a + b;

}

//Overloadride ovlDemo for two double parameters.

**double** ovlDemo(**double** a, **double** b) {

System.***out***.println("Two double parameters: " + a + " " + b);

**return** a + b;

}

}

**class** OverrideDemo {

**public** **static** **void** main(String args[]) {

Override ob = **new** Override(); //instantiation of object

**int** resI;

**double** resD;

// call all versions of ovlDemo()

ob.ovlDemo();

System.***out***.println();

ob.ovlDemo(2);

System.***out***.println();

resI = ob.ovlDemo(4,6);

System.***out***.println("Result of ob.ovlDemo(4,6): " + resI);

System.***out***.println();

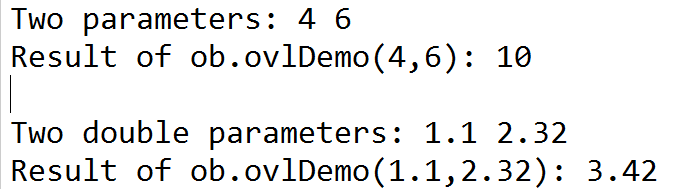
resD = ob.ovlDemo(1.1,2.32);

System.***out***.println("Result of ob.ovlDemo(1.1,2.32): " + resD);

}

}

Output



Modify the above code for part 2 to include three double parameters (values 3.4, 5.6, 7.8)

//Demo for method overriding

class Override {

void ovlDemo(){

System.out.println("No parameters");

}

//Override ovlDemo for one integer parameter.

void ovlDemo(int a) {

System.out.println("One parameter: " + a);

}

//Override ovlDemo for two integer parameters.

int ovlDemo(int a, int b){

System.out.println("Two parameters: " + a + " " + b);

return a + b;

}

//Overloadride ovlDemo for three double parameters.

double ovlDemo(double a, double b, double c) {

System.out.println("Three double parameters: " + a + " " + b + " " + c );

return a + b +c;

}

}

class OverrideDemo {

public static void main(String args[]) {

Override ob = new Override(); //instantiation of object

int resI;

double resD;

// call all versions of ovlDemo()

ob.ovlDemo();

System.out.println();

ob.ovlDemo(2);

System.out.println();

resI = ob.ovlDemo(4,6);

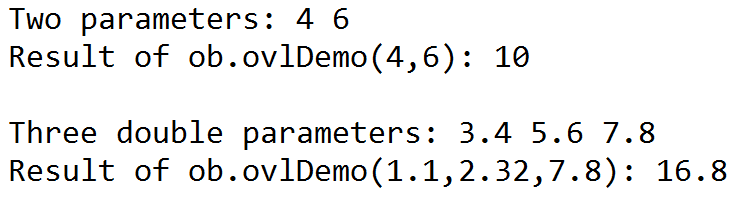
System.out.println("Result of ob.ovlDemo(4,6): " + resI);

System.out.println();

resD = ob.ovlDemo(3.4,5.6,7.8);

System.out.println("Result of ob.ovlDemo(1.1,2.32,7.8): " + resD);

}

}