Assignment No. 02

1) Implement math operations using constructor and instance methods.

```
class MathOperations{

	int num1;//instance
	int num2;//instance

	{
		System.out.println("init block invoked!!");

	}

	MathOperations(int num1, int num2){
		System.out.println("Constructor invoked!!");
		this.num1 = num1;
		this.num2 = num2;

	}

	static int result;

	void add(){
		result = num1 + num2;
		System.out.println(num1+ " + " +num2+ " = "+result);

		sub();
	}

	void sub(){

		result = num1 - num2;
		System.out.println(num1+ " - " +num2+ " = "+result);

		mul();
	}

	void mul(){

		result = num1 * num2;
		System.out.println(num1+ " x " +num2+ " = "+result);
		div();
	}

	void div(){

		result = num1 / num2;
		System.out.println(num1+ " / " +num2+ " = "+result);
```

```
                mod();
        }
        void mod(){

                result = num1 % num2;
                System.out.println(num1+ " % " +num2+ " = "+result);

        }
}

class Q1{

        public static void main(String args[]){

                MathOperations mo1 = new MathOperations(10, 20);
                mo1.add();

                MathOperations mo2 = new MathOperations(30, 40);
                mo2.add();

                MathOperations mo3 = new MathOperations(100, 20);
                mo3.add();
        }
}
```

2) Ref variable assignment demo

```
class StaticRefVar{
    static int i;
    static int j;
    int a;
    int b;

    static StaticRefVar svar;
    static A avar;
}

class A{
    static int k;
    int c;
    int d;
```

```
    static B bvar;

}

class B{
    static int l;
    int e;
    int f;

    static A bavar;

}

class InstanceRefVar{

    int m;
    int n;
    C cvar;



}

class C{

    int x;
    int y;
}


class Q2{
    public static void main(String[] args){
    //Static Reference Variable:
```

```
StaticRefVar.svar = new StaticRefVar();
StaticRefVar.avar = new A();

A.bvar = new B();

B.bavar = new A();

System.out.println("Before Initialization using Static Reference Variable:");
System.out.println("Static i = " +StaticRefVar.i);
System.out.println("Static j = " +StaticRefVar.j);

System.out.println("svar a = " +StaticRefVar.svar.a);
System.out.println("svar b = " +StaticRefVar.svar.b);

System.out.println("avar c = " +StaticRefVar.avar.c);
System.out.println("avar d = " +StaticRefVar.avar.d);
System.out.println("Static k = " +StaticRefVar.avar.k);

StaticRefVar.i = 10;
StaticRefVar.j = 20;
StaticRefVar.svar.a = 30;
StaticRefVar.svar.b = 40;
StaticRefVar.avar.c = 50;
StaticRefVar.avar.d = 60;
StaticRefVar.avar.k = 70;

System.out.println("");

System.out.println("After Initialization using Static Reference Variable:");
System.out.println("Static i = " +StaticRefVar.i);
System.out.println("Static j = " +StaticRefVar.j);
```

```
System.out.println("svar a = " +StaticRefVar.svar.a);
System.out.println("svar b = " +StaticRefVar.svar.b);

System.out.println("avar c = " +StaticRefVar.avar.c);
System.out.println("avar d = " +StaticRefVar.avar.d);
System.out.println("Static k = " +StaticRefVar.avar.k);

System.out.println("");

//Instance Reference Variable:
InstanceRefVar irv = new InstanceRefVar();
irv.cvar = new C();

System.out.println("Before Initialization using Instance Reference Variable:");
System.out.println("irv m = " +irv.m);
System.out.println("irv m = " +irv.n);

System.out.println("cvar x = " +irv.cvar.x);
System.out.println("cvar y = " +irv.cvar.y);

irv.m = 80;
irv.n = 90;
irv.cvar.x = 100;
irv.cvar.y = 200;

System.out.println("");
System.out.println("After Initialization using Instance Reference Variable:");
System.out.println("irv m = " +irv.m);
System.out.println("irv m = " +irv.n);

System.out.println("cvar x = " +irv.cvar.x);
```

```
                System.out.println("cvar y = " +irv.cvar.y);


        }


}
```