**package** constructor;

**public** **class** PointMain {

**private** **int** x, y;

**public** PointMain() { //기본생성자

x = y = 10;

//생성자에는 절대 void를 쓰면 안된다. 클래스명과 똑같이 써주고 기본 생성자이므로 ()를 써준다.

}

**public** PointMain(**int** x, **int** y) { //외부로부터 x와 y를 받는다. setter와 같은 개념

**this**.x = x; //this.x에게 전달

**this**.y = y; //this.y에게 전달

//생성자와 setter는 같은 역할을 하지만 생성자는 new할 때 자동호출을 한다.(1번만 처리)

//setter는 메소드명은 마음대로 잡아주면 되고, 강제호출을 하되 원하는 시점에서 마음대로 불러다 쓸 수 있다.(수정)

//초기값을 잡을 때는 생성자, 값을 변경할 때는 setter.

}

**public** PointMain(PointMain aa, PointMain bb) { //객체형(클래스형)

**this**.x = aa.x + bb.x;

**this**.y = aa.y \* bb.y;

}

**public** **void** calcPlus() {

x += 10;

}

**public** **void** calcPlus(**int** x, **int** y) { //3과 -8 인수를 받는 메소드

**this**.x += x;

**this**.y += y;

}

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

PointMain aa = **new** PointMain();

System.***out***.println("x=" + aa.x + "\t y=" + aa.y);

System.***out***.println("\n x의 값을 +10 증가하는 메소드를 작성하시오 - calcPlus()");

aa.calcPlus();

System.***out***.println("x=" + aa.x + "\t y=" + aa.y);

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

PointMain bb = **new** PointMain(100, 200);

System.***out***.println("x=" + bb.x + "\t y=" + bb.y);

System.***out***.println("\n 3, -8 인수로 받아서 x는 증가, y는 감소하는 메소드를 작성하시오- calcPlus()");

bb.calcPlus(3, -8); //인수 받기

System.***out***.println("x=" + bb.x + "\t y=" + bb.y);

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

System.***out***.println("\n x는 aa의 x값과 bb의 x의 값을 더하고 y끼리는 곱하기");

PointMain cc = **new** PointMain(aa, bb); //new하면서 불려졌으므로 생성자

System.***out***.println("x=" + cc.x + "\t y=" + cc.y);

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

}

}// class PointMain

[결과]

x=10 y=10

x의 값을 +10 증가하는 메소드를 작성하시오 - calcPlus()

x=20 y=10

x=100 y=200

3, -8 인수로 받아서 x는 증가, y는 감소하는 메소드를 작성하시오- calcPlus()

x=103 y=192

x는 aa의 x값과 bb의 x의 값을 더하고 y끼리는 곱하기

x=123 y=1920