1. public class ThisDemo {
3. int a;
4. int b;
6. ThisDemo(int a, int b){
8. a=a;
9. b=b;
11. }
13. public static void main(String[] args) {
15. ThisDemo obj = new ThisDemo(10, 20);
17. System.out.println(obj.a);
18. System.out.println(obj.b);
19. }
21. }

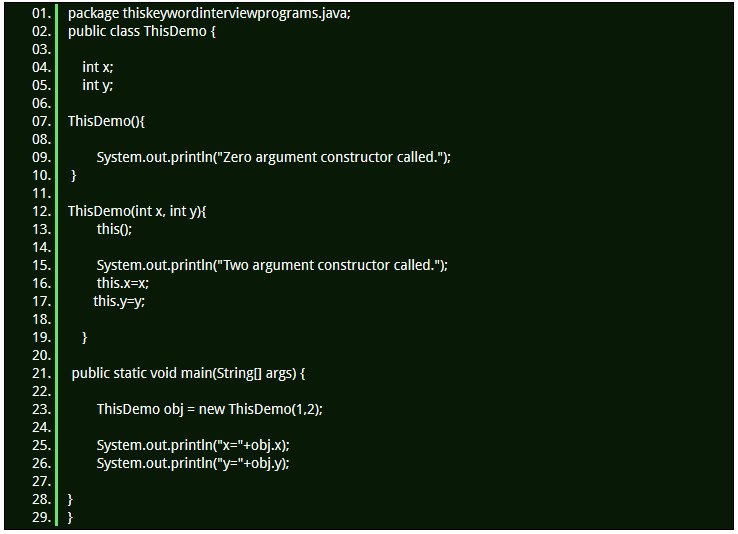
0 0

1. public class ThisDemo {
3. int a;
4. int b;
6. ThisDemo(int a, int b){
8. this.a=a;
9. this.b=b;
11. }
13. public static void main(String[] args) {
15. ThisDemo obj = new ThisDemo(10, 20);
17. System.out.println(obj.a);
18. System.out.println(obj.b);
19. }
21. }

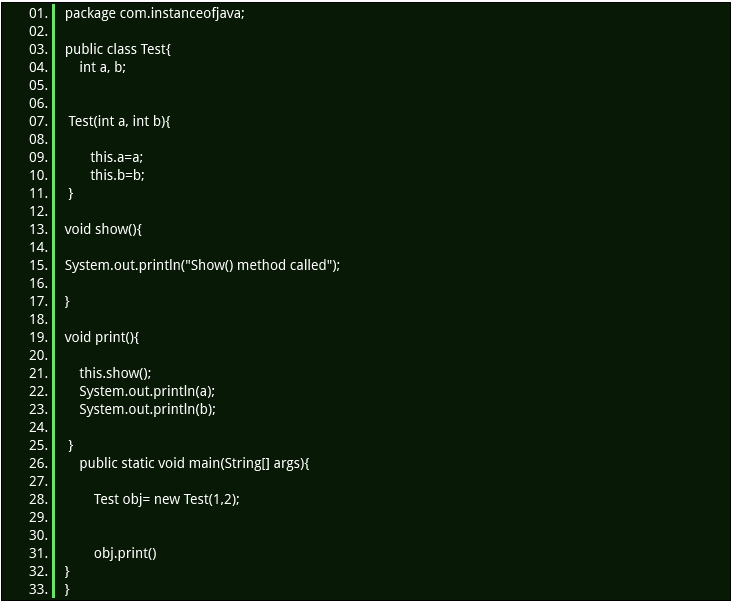
10 20

1. public class ThisDemo {
3. int a;
4. int b;
6. ThisDemo(int a, int b){
8. this.a=a;
9. this.b=b;
11. }
13. public static void main(String[] args) {
15. ThisDemo obj = new ThisDemo();
17. System.out.println(obj.a);
18. System.out.println(obj.b);
19. }
21. }

Compilation Error



1. Zero argument constructor called.
2. Two argument constructor called.
3. x=1
4. y=2



1. Show() method called
2. 1
3. 2

**Can we call non static method from constructor using this?**



1. B class Constructor
2. method

**Can we assign something to this ?**

1. public class ThisDemo {
3. int a;
4. int b;
6. ThisDemo(int x, int y){
8. this= new ThisDemo();
9. System.out.println("Two argument constructor called.");
10. this.x=x;
11. this.y=y;
13. }
15. public static void main(String[] args) {
17. ThisDemo obj = new ThisDemo(10, 20);
19. System.out.println(obj.a);
20. System.out.println(obj.b);
21. }
23. }
24. **Compile Time Error at line number 9: left hand side of assignment must be a variable**
25. **We can not modify or assign something to this it leads to compile time error**

**Can we use this as return statement in a method?**

1. public class B{
2. int a;//
4. public int getA() {
5. return a;
6. }
8. public void setA(int a) {
9. this.a = a;
10. }
12. B show(){
13. return this;
14. }
16. public static void main(String[] args) {
18. B obj = new B();
20. obj.setA(10);
22. System.out.println(obj.getA());
23. B obj2= obj.show();
24. System.out.println(obj2.getA());
26. }
27. }

10

10