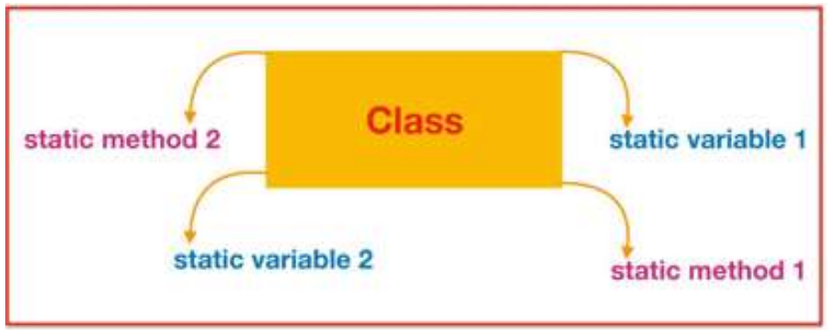
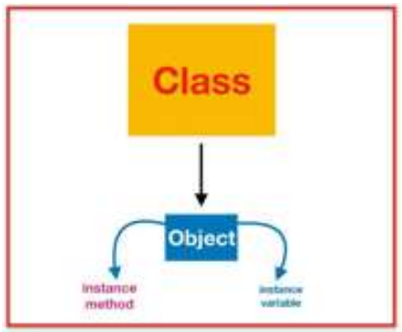
Static Keyword



Static variables / methods



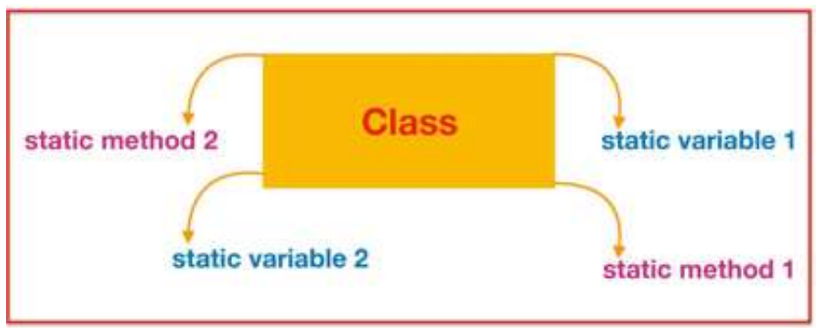
Non-static variables / methods

static kelimesi bir variable'i veya Method'u Class'a baglamak icin kullanilir.

Bir variable veya Method static olarak etiketlendiginde o artik class'in elemani olur ve ona ulasmak icin object olusturmamiza gerek kalmaz.

Instance variable'a ulasmak icin ise MUTLAKA object olusturmaliyiz

Static Variables



Static variables / methods

1) Class yuklendiginde, memory'de static variable'lar olusturulur.

2) Static variable'lar bir tane olusturulur ve class'daki tum objeler onu gorur ve kullanir.

3) Memory kullanimi static variable'lar icin sadece bir kere olur.

4) Static variable'lar static veya static olmayan methodlarin icinde kullanilabilir.

5) Static variable'lara baska classlardan sadece class ismi kullanilarak ulasilabilir, obje olusturmaya gerek yoktur.

Static Variables

Asagidaki class calistirildiginda output ne olur ?

```
public class Deneme {

    static int count=0;

    public void increment() {
        count++;
    }


    public static void main(String[] args) {
        Deneme obj1=new Deneme();

        Deneme obj2=new Deneme();

        obj1.increment();

        obj2.increment();

        System.out.println("Obj1: count is="+ obj1.count);
        System.out.println("0bj2: count is="+obj2.count);
    }}
```

Static Variables

Asagidaki class calistirildiginda output ne olur ?

```
public class Deneme {

    int x;
    static int y;

    Deneme(int i){
        x+=i;
        y+=i;
    }

    public static void main(String[] args) {

        new Deneme(2);

        Deneme dnm=new Deneme(3);

        System.out.println(dnm.x + "," + dnm.y);
    }}
```

Static Methods

1) Return Type'dan once static keyword kullanarak, static method olusturabiliriz

```
public class Deneme {

    public static void main(String[] args) {

    }
    public static void add() {
    }
    }
```

Static Methods

2) Static Method'lar static variable (class variables) lari direk kullanabilirler

```
public class Deneme {

    static int sayi1=10;
    int sayi2=20;

    public static void main(String[] args) {
        System.out.println(sayi1);

        System.out.println(sayi2);

    }

    public static void add() {
        System.out.println(sayi1);

        System.out.println(sayi2);
    }

    }
```

ama static olmayanlari object olusturmadan kullanamazlar

Static Methods

3) Static Method'lar static ve non-static method'lardan cagrilabilir.

```
public class Deneme {

    public static void main(String[] args) {
        add();
    }

    public static void add() {

    }

    public void concat() {
        add();
    }

}
```

Static Methods

Asagidaki class calistirildiginda output ne olur ?

```
public class Counter {
    int count;
    static int stCount;
    public Counter() {
        count ++ ;
        stCount ++ ;
    }
    public int getCount(){
        return count;
    }
    public static int getStCount(){
        return stCount;
    }

    public static void main(String[] args) {

        Counter cs1 = new Counter();
        Counter cs2 = new Counter();
        Counter cs3 = new Counter();
        Counter cs4 = new Counter();
        Counter cs5 = new Counter();
        Counter cs6 = new Counter();
        System.out.println("count is: " + cs6.getCount());
        System.out.println("stCount is: " + cs6.getStCount());
    }
    }
```

Static Methods

Asagidaki class calistirildiginda output ne olur ?

```
public class StaticMember {
    static int x;
    int y;

    StaticMember(){
        x+=2;
        y++;
    }
    static int getSquare(){
        return x * x;
    }
    public static void main(String[] args) {
        StaticMember sm1 = new StaticMember();

        StaticMember sm2 = new StaticMember();

        int z = sm1.getSquare();

        System.out.print("-x" + z + "-y" + sm2.y);

    }
}
```

Static Methods

Asagidaki class calistirildiginda output ne olur ?

```
class Counter {
        int count=0;

        Counter(){
        count++;
        System.out.println(count);
        }

        public static void main(String args[]){

        Counter c1=new Counter();
        Counter c2=new Counter();
        Counter c3=new Counter();
        }
}
```

Static Methods

Asagidaki class calistirildiginda output ne olur ?

```
class Student{

  int number;
  String name;
  static String college ="ITS";

  Student(int r, String n, String college){
        this.number = r;
        this.name = n;
        this.college = college;
  }

  public static void main(String args[]){

     Student s1 = new Student(111,"Karan", "MIT");
     Student s2 = new Student(222,"Aryan", "Harvard");

     System.out.println(s1.number);
     System.out.println(s2.number);

     System.out.println(s1.name);
     System.out.println(s2.name);

     System.out.println(s1.college);
     System.out.println(s2.college);

  }
}
```

Static Methods

What is the result of the following program?

```
1: public class Squares {
2:   public static long square(int x) {
3:     long y = x * (long) x;
4:     x = -1;
5:     return y;
6:   }
7:   public static void main(String[] args) {
8:     int value = 9;
9:     long result = square(value);
10:     System.out.println(value);
11:   } }
```

A. -1

B. 9

C. 81

D. Compiler error on line 9.

E. Compiler error on a different line.

Instance Variables vs Static Variables

Instance Variable

1) instance variables .....'in icinde ama .....'in disinda olusturulur

2) Instance variables bir ..... 'e baglidir. Dolayisiyla, bir ..... olusturuldugunda olusur ve ..... silindiginde silinirler.

3)Instance variables .....ismi ile cagrilabilirler.

4) instance variable icin ilk deger atamasi yapmak .....dir. Eger ilk atama yapilmazsa default deger alir.

5) Her yeni obje olusturuldugunda, instance variables ilk atanan degere esit olur.True / False

6) Bir class'i kullanarak 2 instance variable'a sahip 6 obje olusturursak, 12 instance variables olusturmus oluruz. True / False

Static Variable

1) Static variables .....'in icinde ama .....'in disinda olusturulur

2) Static variables bir ..... 'a baglidir. Dolayisiyla, bir ..... olusturuldugunda olusur ve ..... silindiginde silinirler.

3) Static variables .....ismi ile cagrilabilirler.

4) Static variable icin ilk deger atamasi yapmak ..... dir. Eger ilk atama yapilmazsa default deger alir.

5) Class variable'a her yeni deger atamasi oldugunda, degeri tum objeler icin degisir.True / False

6) Bir class'i kullanarak 2 static variable'a sahip 6 obje olusturursak, 2 static variables olusturmus oluruz. True / False

Static Blocks

1) Static block static variable'lara deger atamasi yapmak icin kullanilir.

2) Static block, class ilk calistirilmaya baslandiginda calisir ve static variable'lara ilk deger atamasi yapar (initialize)

3) Static block'lar constructor'lardan, tum method'lardan ve main method'dan once calisir.

4) Eger 1'den fazla static block varsa ustteki blok daha once calisir.

```
public class StaticBlock {
    public static int age;

    static {
        System.out.println("Static block 2 calisti");
        age = 24;
    }

    static {
        System.out.println("Static block 1 calisti");
        age = 23;
    }

    public StaticBlock() {
        System.out.println("Constructor calisti");
        System.out.println(++age);
    }

    public static void main(String[] args) {

        System.out.println("Main method calisti 1");
        System.out.println(++age);
        StaticBlock obj = new StaticBlock();
        System.out.println("Main method calisti 2");
    }
}
```