

Final keyword



① Primitive data type
final int a = 2;

a = 3 // X

Error → a is final variable

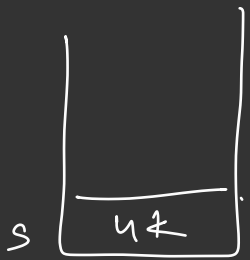
② Non Primitive

final Student s = new Student();

s = null

s = new Student] X Error

s.name = "Tanmay" ✓



Wrapper Class

ArrayList < Integer >

int

Integer
Character
Boolean

Integer a = 2;

Integer b = new Integer(2);

Static keyword

↳ ① Variables

② functions

③ blocks

④ class

```
class Student {  
    int rollno;  
    String name;  
    static int count;
```

```
    Student() {  
        count++;
```

```
    }
```

```
}
```

```
main () {
```

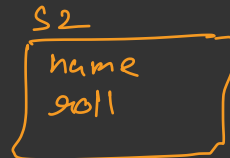
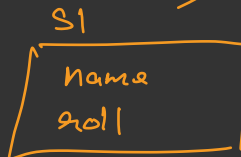
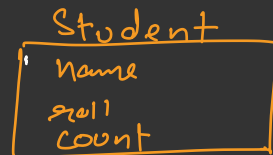
```
    Student s1 = new Student();
```

```
    Student s2 = new Student();
```

```
    syso (s1.count)
```

```
    syso (s2.count)
```

student.count



② Static functions

↳ called from class name
doesn't require any instance

Static Variable / function

↳ does not depend on instance

static function] → You cannot have a non static member

```
class Student {
```

```
    String name;  
    static int count;
```

```
    static void printName() {
```

```
        sysout(count)
```

```
        sysout(name) // X Error
```

```
        Student s = new Student();
```

```
        s.name
```

Q Longest Common Prefix (LCP)

→ [geeks, geeks for geeks, geeky, geek]
⇒ geek

sort

Arrays.sort(arr);

→ [geek, geeks, geeks for geeks, geeky]

g e e k -
↑ ↑ ↑
g e e k y

s. substring(0, 5)

length = ~~0~~ ~~1~~ ~~2~~ 3 4

② No Sorting

LCP(geeks, geek, geeky) ⇒ geek

LCP(LCP(geeks, geek), geeky) ⇒ geek

↓
geek

```

b s String LCPUtil (String arr[]) {
String result = LCP (arr[0], arr[1])
for (i = 2; i < arr.length; i++) {
    result = LCP (result, arr[i]);
}
return result;
}

```

```

b s String LCP (String a, String b) {
int l = Math.min (a.length(), b.length())
int count = 0;
for (i = 0; i < l; i++) {
    if (a.charAt(i) != b.charAt(i)) {
        break;
    }
    count++;
}
}

```