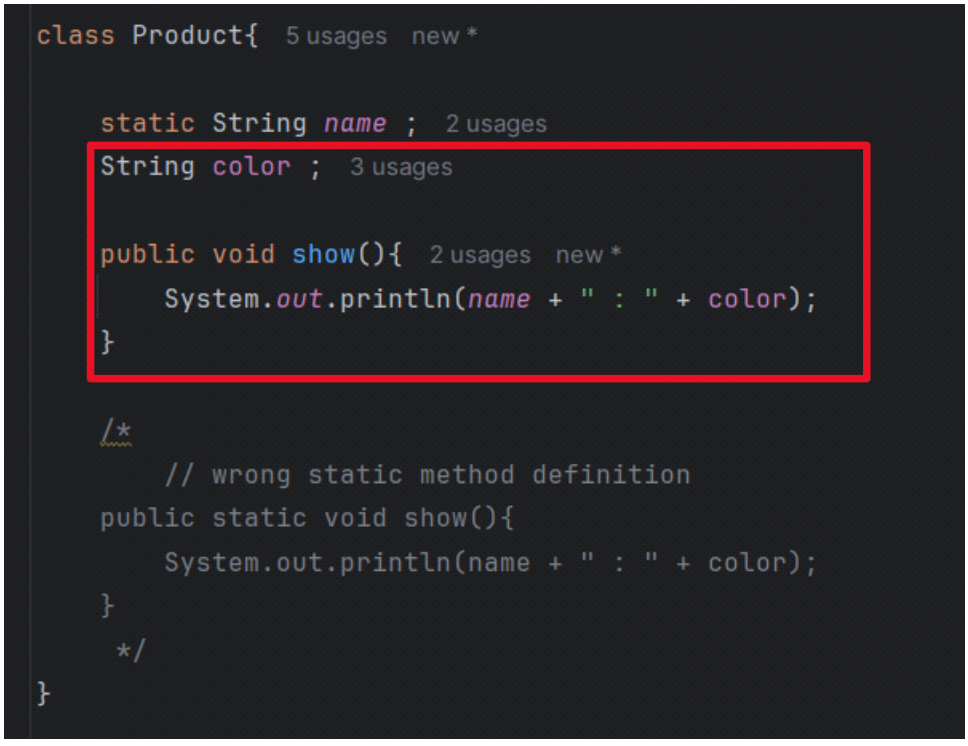
Instance and Static

Monday, August 18, 2025 3:25 PM

Instance Variables and Methods

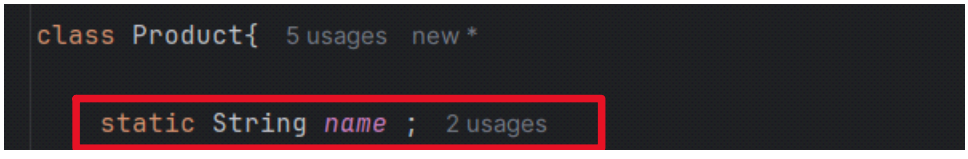
- **Instance Variable**:
  - Belongs to an **object** (instance) of the class.
  - Each object gets its **own copy** of instance variables.
  - Declared **without static** keyword.
  - Exists as long as the object exists.
- **Instance Method**:
  - Works on **instance variables**.
  - Can be called only through an **object**.
  - Can access both **instance** and **static** variables/methods.
- Eg.



```
class Product{  5 usages  new *

    static String name ;  2 usages
    String color ;  3 usages

    public void show(){  2 usages  new *
        System.out.println(name + " : " + color);
    }

    /*
        // wrong static method definition
    public static void show(){
        System.out.println(name + " : " + color);
    }
     */
}
```

◇ Static Variables and Methods

- **Static Variable**:
  - Belongs to the **class**, not to objects.
  - **Shared among all objects** (only one copy exists in memory).
  - Declared using static keyword.
- **Static Method**:
  - Belongs to the **class**.
  - Can be called **without creating an object**.
  - Can access only **static variables** and other **static methods** (cannot directly access instance variables).
- Eg.



```
class Product{  5 usages  new *

    static String name ;  2 usages
```

```
    static String name ;  2 usages
    String color ;  3 usages

    public void show(){  2 usages  new *
        System.out.println(name + " : " + color);
    }

    /*
        // wrong static method definition
    public static void show(){
        System.out.println(name + " : " + color);
    }
     */
}
```

```
public class Instance_static {  new *
    public static void main(String[] a){  new *
```

Notes :

1. A instance variable cannot be called inside a static method .
   - Cause the value of the instance variable is different for different object , thus when called it creates conflict
   - Eg.
     -
```
// wrong static method definition
public static void show(){  no usages  new *
    System.out.println(name + " : " + color);
}
```
2. Why main is a static method
   - as if main is instance method then we need an object to call it
3. Initialization of static variable is done using class name
   -
```
// initialization of static variable
Product.name = "pen" ;

// initialization of instance variable among objects
Product pen1 = new Product() ;
pen1.color = "black" ;

Product pen2 = new Product() ;
pen2.color = "blue" ;
```