

Day 3

📅 Created	@Mar 10, 2021
👤 Created by	ⓧ VIGNESH J
⋮ Tags	Java

Problem

Variables

Scope

Code

Class

Problem

- Onetime
- Recurring
- Analyze the problem
- 3D Thinking
- Keep Analyze
- Object Oriented Analyze
- Object Oriented Design
- Object Oriented Programming

Best Free Online Video Recording & Screen Capture Software | Fluvid

Fluvid is the best all-in-one online screen capture & video recording software that is available for free. Fluvid helps you record, edit, communicate & share your video messages in the most simplest way.

 <https://fluvid.com/videos/detail/VBpQeinLdkHBxvQkX#.YEHV4g4TgLc.link>

10:06 AM - 10:44 AM

Variables

Integers

- long
- int
- short
- byte

Floating - Numbers

- double
- float

Characters

- char

Boolean

- bool

Class - Plan

- Methods
 - Name
 - Return Type
 - Void if no return
- Variables
 - instance variables...those variables which are declared inside a class
 - by default instance variables are initialized to there default values..

Object - Real Thing

Scope

- Public - Global Scoped
- Private - Class scoped
- Protected - Package & Relation (inheritance) scoped
- NOMOD - Package scoped

Access Protection

Aa #	▼ No Modifier	▼ Private	▼ Protected	▼ Public
<u>Same package sub class</u>	YES	No	YES	YES
<u>Same package non subclass</u>	YES	No	YES	YES
<u>Same Class</u>	YES	YES	YES	YES
<u>Different package sub class</u>	NO	No	YES	YES
<u>Different package non sub classs</u>	NO	No	NO	YES

Code

```
package day3;

public class AccessDemo {
    // instance variables...those variables which are declared inside a class
    // by default instance variables are initialized to there default values..
    public int pub;
    // global scope
    private int pri;
    // class scope
    protected int pro;
    // relation (inheritance) scope
```

```

int nomod;
// package scope

public void display() {
    System.out.println(pub);
    System.out.println(pri);
    System.out.println(pro);
    System.out.println(nomod);
}
}

class AccessDemoSub extends AccessDemo {
    public void display() {
        System.out.println(pub);
        // System.out.println(pri);
        System.out.println(pro);
        System.out.println(nomod);
    }
}

class AccessDemoNonSub {
    AccessDemo obj = new AccessDemo();

    // obj is not the object - it is just reference or name
    // what is object here - the object is AccessDemo object.
    public void display() {
        System.out.println(obj.pub);
        // System.out.println(obj.pri);
        System.out.println(obj.pro);
        System.out.println(obj.nomod);
    }
}

```

```

package day2;

import day3.AccessDemo;

/*
 * private - class scope
 * nomod- package scope
 * protected - relation (inheritance) scope
 * public - global scope
 */
public class AccessDemoAnotherPackNonSubClass {
    AccessDemo obj = new AccessDemo();

    // obj is not the object - it is just reference or name
    // what is object here - the object is AccessDemo object.
    public void display() {
        System.out.println(obj.pub);
        // System.out.println(obj.pri);
        // System.out.println(obj.pro);
        // System.out.println(obj.nomod);
    }
}

```

```

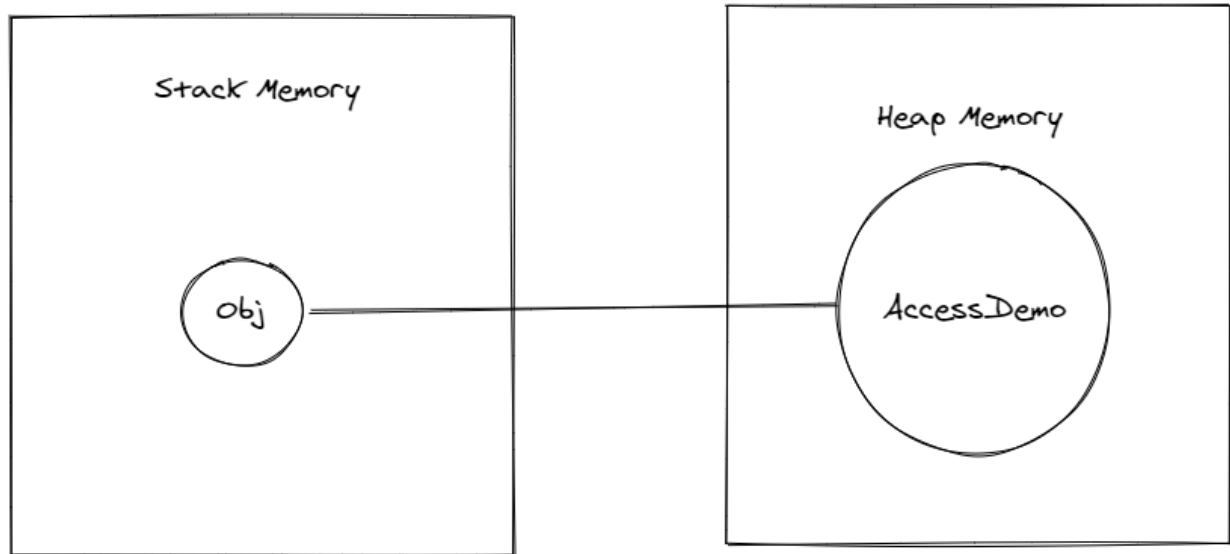
package day2;

import day3.AccessDemo;

//private - class scope
//nomodifier - package scope
public class AccessDemoAnotherPackSubClass extends AccessDemo {
    public void display() {
        System.out.println(pub);
        // System.out.println(pri);
    }
}

```

```
System.out.println(pro);  
// System.out.println(nomod);  
}  
}
```



How object stored in memory

Best Free Online Video Recording & Screen Capture Software | Fluvid

Fluvid is the best all-in-one online screen capture & video recording software that is available for free. Fluvid helps you record, edit, communicate & share your video messages in the most simplest way.

https://fluvid.com/videos/detail/AZX-wTr7qXFm4_Rky

11 AM - 11:14 AM

Class

- Proper Name
- CamelCase
- Body
- Properties
 - variables
 - always start with lower case
 - instance variables
- Methods
 - With or Without Access Modifier

- Can have Return Type Or Void
- If variable in a method should return something
- Can have parameter
- Can not return more than one value
- Method should start with lowercase

Constructor

- class can have constructor
- called when at the time of object
- name of the constructor and class name should be same
- can be called only once
- used to what to and what not to initialize
- can be many constructor
- can have overloading constructor, same name with different parameter

Method Overload

- Method can be overloaded
- With different no of parameter
- Data changes behavior changes
- Can have different return type
- Can have different Access Modifier

```
package ClassExample;

public class Employee {
    // constructor overload or method overload...
    public Employee() {
        System.out.println("cons called..");
    }

    public Employee(String s) {
        System.out.println("Parametric constructor called...");
    }

    String name; // instance variables

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name; // this always refers to instance variable...
    }

    // overloaded methods...
    // overloaded methods access specifier can change...
    // overloaded method return type can change...
    // you cannot change the method name-if you change it is not overloading then
    // parameter compulsory to be different
    int met() {
        System.out.println("no param method called..");
    }
}
```

```

        return 1;
    }


    public void met(int i) {
        System.out.println("param method called....");
    }

    public static void main(String[] args) {
        Employee emp; // no object is created, only variable reference is created....
        new Employee(); // orphan object with no variable reference...
        Employee e = new Employee("hjhjhjhj");
        e.setName("VIGNESH J");
        System.out.println(e.getName());
        // dynamically java identifies
        // VMI - virtual method invocation or dynamic method invocation - polymorphism
        e.met();
        e.met(100);
    }
}

```

Best Free Online Video Recording & Screen Capture Software | Fluvid

Fluvid is the best all-in-one online screen capture & video recording software that is available for free. Fluvid helps you record, edit, communicate & share your video messages in the most simplest way.

 https://fluvid.com/videos/detail/R_4m9Sa1q8TMAp11A#.YEiW-_9snhl.link

2:49 PM - 3:20 PM

Method Overriding

- Inheritance
 - Kind of relation

```

package ClassExample;

public class ClassesAndObject {
    public static void main(String[] args) {
        LeatherShoe shoe = new LeatherShoe();
        shoe.met(2);
        shoe.leetcodeMethod();
    }
}

class A {
    public A() {
        System.out.println("a cons called...");
    }
}

class Shoe extends A {
    public Shoe() {
    }

    public Shoe(int i) {
        System.out.println("shoe cons called...");
    }

    // final declared methods cannot be overridden
    // final
    protected void met(int i) {
        System.out.println("met of shoe called...");
    }
}

```

```

//inheritance - Kind of Relationship - java does not support multiple inheritance but it supports multi level inheritance
class LeatherShoe extends Shoe {
    public void leatherMethod() {

    }

    public LeatherShoe() {
        super(10); // this super will invoke the parent class constructor and remember this should
        // be the first line
        System.out.println("leather shoe cons called...");
    }

    // overriding
    // changing the method name or method parameters will not amount to overriding
    // the parameters and the method name both should be same
    // you cannot change the return type
    // you can increase the visibility but you cannot decrease the visibility
    // override annotation checks whether the method below is declared in the parent
    // class or not
    // if not declared it will throw a error.
    @Override
    public void met(int i) {
        System.out.println("met of leathershoe called...");
        super.met(10); // super is a keyword - use to invoke the parent method...
    }
}

```