

Week 9 : ADT Design

- How to download Java in linux(Ubuntu)

1. Check java version

java -version

```
dcs1ab@computer1:~/java$ java -version
java version "1.8.0_40"
Java(TM) SE Runtime Environment (build 1.8.0_40-b26)
Java HotSpot(TM) 64-Bit Server VM (build 25.40-b25, mixed mode)
```

2. if not, check OS version and start download jdk

file /sbin/init

if you want to delete existed java, use this command : sudo
apt-get purge openjdk-~~W~~*

3. make a new folder

sudo mkdir -p /usr/local/java

4. click the accept button and download exe file using 'wget'
command

copy linked url or copy url on attribution

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

5. copy downloaded file

sudo mv jdk-8u40-linux-x64.tar.gz /usr/local/java/

6. decompress

cd /usr/local/java;

sudo tar xvfz /usr/local/java/jdk-8u40-linux-x64.tar.gz

7. Register environment

```
sudo vi /etc/profile
export JAVA_HOME=$(readlink -f /usr/bin/java | sed "s:bin/java::")
sudo source /etc/profile
```

8. Register commands

```
sudo update-alternatives --install "/usr/bin/java" "java" "/usr/local/java/jdk-8u40-linux-x64/bin/java";
sudo update-alternatives --install "/usr/bin/javac" "javac" "/usr/local/java/jdk-8u40-linux-x64/bin/javac";
sudo update-alternatives --install "/usr/bin/javaws" "javaws" "/usr/local/java/jdk-8u40-linux-x64/bin/javaws";
sudo update-alternatives --set java /usr/local/java/jdk-8u40-linux-x64/bin/java;
sudo update-alternatives --set javac /usr/local/java/jdk-8u40-linux-x64/bin/javac;
sudo update-alternatives --set javaws /usr/local/java/jdk-8u40-linux-x64/bin/javaws;
```

9. check a installation of java

```
java -version
```

Part1. ADT Design

Define as class.

Every objects are allocated in heap space.

Encapsulation : Data representation + Operation

Information Hiding : Object's representation part hides, and user access object by operation.

- Form of class

```
public class Person {
    public String name;
    public int age;
    public Person() {
    }
    public Person(String s) {
        name = s;
    }
    public String getName() {
        return name;
    }
}
```

```
}  
}
```

Constructor : The method that has same name with class name.

Method : Possibly execute. an object behavior is implemented in Method.

Method Overloading :

- ① Same method name
- ② Different number of the parameter or different type of the parameter.
- ③ Error : Only different return type.

- Generating Object

```
public static void main (String args[]) {  
    Person aPerson;  
    aPerson = new Person("홍길동");  
    aPerson.age = 30;  
    String s = aPerson.getName();  
}
```

After declaring reference variable about object 'aPerson', Person object is generated by 'new' keyword.

- Member Access Modifier

Accessing class	Member Access Modifier			
	default	private	protected	public
Same package class	O	X	O	O
Different package class	X	X	X	O

```
class Person {  
    public String name;  
    private int age;
```

```
    int num;

    public int getAge() {
        return age;
    }

    public void setAge(int value) {
        b = value;
    }
}

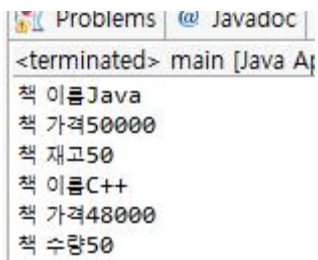
public class Access {
    public static void main(String[] args) {
        Person aPerson = new Person();
        aPerson.name = "홍길동";
        aPerson.setAge(20);
        aPerson.num = 10;
    }
}
```

You can define public declared name instantly. When private you define private declared age instantly, then an error occurs. Because age is declared at different class. so, you have to make a get/set method to access private variable.

[Exercise]

1. Make a 'Goods' class which has name, price, stock. And print out that information at main method.

```
public class Goods {  
    /*blank*/  
    Goods(String name, int price, int numberOfStock)  
    {  
        this.name = name;  
        this.price = price;  
        this.numberOfStock = numberOfStock;  
    }  
    /*blank*/  
}
```



```
<terminated> main [Java A]  
책 이름Java  
책 가격50000  
책 재고50  
책 이름C++  
책 가격48000  
책 수량50
```