



UI/UX 전문가 과정 비트캠프

2021년 01월 13일

장 해 솔

이메일 : wkdgothf@gmail.com

github : <https://github.com/legossol>

목표

1. 오늘 배운 것들
 - 상속자
2. 질문 노트 작성
 - Tenth(Matrix) 질문

*상속자

: 자신이 가지고있는 private 를 extends를 통해서 다른 클래스로 넘겨줄 수 있는 특성을 가지고 있다

Ex

아래와같이 넘겨주는 부모에 해당하는 person2가 가지고있는 프라이빗 나이와 이름을

```
1 package Eleventh;
2
3 class Person2 {
4     private String name;
5     private int age;
6
7     public Person2(String name, int age) {
8         this.name = name;
9         this.age = age;
10    }
11
12    public String getName() {
13        return name;
14    }
15
16    public int getAge() {
17        return age;
18    }
19
20    @Override
21    public String toString() {
22        return "Person{" +
23            "name=" + name + '\'' +
24            ", age=" + age +
25            '\'' +
26        }
27    }
```

자식에 해당하는 student2가 상속을 받아 이름과 나이를 따로 적지 않아도 사용할 수 있다.

```
InheritanceTest.java
1/janghaesol/java_work/src/Eleventh

28
29 // extends 키워드는 상속 키워드입니다.
30 // class Student2 extends Person2 는
31 // Student2 클래스에 Person2 클래스의 정보를 상속하겠다는 의미
32 class Student2 extends Person2 {
33     private String major;
34
35     public Student2(String name, int age, String major) {
36         // super() 는 부모 클래스를 의미한다.
37         // this 는 자기 자신을 의미한다.
38         super(name, age);
39         this.major = major;
40     }
41
42     // reference method
43     public void test() {
```

사용 이유 : 유지보수와 메모리 절약.

*궁금한 점.

```
private boolean checkDivideElement(int len, int row) {
    if(len % row == 0) {
        this.row = row;
        this.col = len / row;
    } else {
        System.out.printf("행렬로 변환할 수 없습니다.\n");
        System.out.printf("올바른 차원을 입력하세요.\n");
        System.out.printf("혹은 적절한 숫자(행)를 입력하세요\n");
        return false;
    }
    return true;
}

private boolean checkAvailable(int[] arr, int row, int col) {
    int len = arr.length;
    boolean res = (len == row * col ? true : false);
    if(res) {
        this.row = row;
        this.col = col;
    }
    return res;
}

private boolean checkAvailable(int[][] arr, int num) {
    int row = arr.length;
    int col = arr[0].length;
    int len = row * col;
}
```

Q1 : 위 코드에서 마지막줄 int len = row * col;이면 checkDivideElement로 가서 검사를 시작한다

여기서 궁금한게 아래와 같이 row값을 이미 입력 받은 상태이기에 그냥 boolean에서 같은 int num으로 바꾸고 row==num을 하는 것과 차이가 있는 걸까?

```
System.out.println("Matrix B:");
Matrix B = new Matrix(arr2, row: 3);
```

```
public Matrix(int row, int col) {
    this.row = row;
    this.col = col;
    mat = new int[row][col];
}
```

Q2 : 11회차 수업 숙제중 2번째의 풀이를 위해 아래와 같이 작성하였하였다

```
package jan0113;

class Transport{
    private final String size;
    private final String nic;
    private final String species;
    private final String enginePA;

    public Transport(String Size,String nic,String species,String enginePA) {
        this.size = Size;
        this.nic = nic;
        this.species = species;
        this.enginePA = enginePA;
    }

    public String getSize() { return size; }
    public String getnic() {
        return nic;
    }

    public String getspecies() {
        return species;
    }

    public String getenginePA() { return enginePA; }

    @Override
    public String toString() {
        return "Transport{" +
            "size='" + size + '\'' +
            ", nic='" + nic + '\'' +
            ", species='" + species + '\'' +
            ", enginePA='" + enginePA + '\'' +
            '}';
    }
}

class overseas extends Transport {
    private String canFishingOrNot;

    public overseas(String size, String nic, String species, String enginePA, String canFishingOrNot) {
        super(size, nic, species, enginePA);
        this.canFishingOrNot = canFishingOrNot;
    }
}
```

이와같이 상속이 한명이 아니라 두명이상으로 하여서
 공통으로 나타내야 하는 부분을 클래스Transport로 만들고
 그 Transport를 상속받아 특정한 내용들을 넣어야하는 overseas와 land를 만들었는데
 이와같이 활용하는 것이 맞는 것인가.

Q3 : 위의 문제 풀이하던 중 overseas 와 land에서 toString을 사용하려 했지만 자동적으로
 Transport클래스의 잡히지 않았다. 안잡힐 수 있다는 것은 알지만 바로 전 문제인
 homework1에서는 잡히는 것이 여기서는 안잡히는 이유가 궁금하다.

