

②

Create a package CIE which has two classes - Students and Internals. The class Personal has members like usn, name, sem. The class Internals has an array that stores the internal marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.

→ package CIE;

```
public class Student {
    public String sem usn;
    public String name;
    public int sem;
}
```

→ package CIE;

```
public class Internal extends Student {
    public int[] internalMarks = new int[5];
}
```

→ package SEE;

```
import CIE.Student;
```

```
public class External extends Student {
    public int[] externalMark = new int[5];
}
```

→ import CIE.Internal;

import SEE.External;

import java.util.Scanner;

public class Main {

public static void main(String args[]) {

~~int~~ ~~no~~ ~~args~~

```
Scanner s = new Scanner(System.in);
```

```
int n = s.nextInt();
```

```
int[] internal = new int[n];  
External[] secStudents = new External[n];
```

```
for (int i = 0; i < n; i++) {  
    System.out.print("USN: ");  
    String usn = s.next();  
    System.out.print("Name: ");  
    String name = s.next();  
    System.out.print("Sem: ");  
    int sem = s.nextInt();
```

```
int[] creMarks = new int[n];  
for (int j = 0; j < n; j++) {  
    creMarks[j] = s.nextInt();  
}
```

```
creStudents[i] = new Internal(usn, name, sem,  
                                creMarks);  
}
```

```
for (int i = 0; i < n; i++) {  
    String usn = s.next();  
    String name = s.next();  
    int sem = s.nextInt();
```

```
int[] secMarks = new int[n];  
for (int j = 0; j < n; j++) {  
    secMarks[j] = s.nextInt();  
}
```

see Students [i].new External (usr, name, sem,
see Marks);

for (int i = 0; i < N; i++)

System.out.println("see Students [i] user");

System.out.println("see Students [i] name");

System.out.println("see Students [i] sem");

System.out.println("see Marks: ");

for (int j = 0; j < N; j++)

System.out.println("see Students [i] internal

Marks [j] + " ");

}

System.out.println("Set Marks: ");

for (int j = 0; j < N; j++)

System.out.println("see Students [i] see Marks [j] + " ");

}

}

}

}

29/1/29

O/P

QSN IBM 22CS150 name Jee Sem 3

Internal marks 1: 43

2: 45

3: 47

4: 46

5: 41

Internal Marks: 1: 90

2: 87

3: 65

4: 99

5: 43

- ① Write a program to demonstrate handling of exceptions in inheritance too. Create a base class called 'Father' and derived class called 'Son' which extends the base class. In Father class, implement a constructor which takes the age and throws the exception 'WrongAge!' when the input age < 0 .
- ② Write a program which creates two threads, one thread displaying 'BMS Collg. of Engineering' once every 10 sec and another displaying 'CSE' once every two seconds.

```

class WrongAge extends RuntimeException {
    public WrongAge() {
        super("Age cannot be negative");
    }
}

```

```

class Father {
    private int age;

    public Father(int age) throws WrongAge {
        if (age < 0)
            throw new WrongAge();
        this.age = age;
    }

    public int getAge() {
        return age;
    }
}

```

```

class Son extends Father {
    private int sonAge;

    public Son(int fatherAge, int sonAge) throws
        WrongAge, IllegalArgumentException {

```

```
super(fatherAge);
```

```
if (sonAge > fatherAge) {
```

```
    throw new IllegalArgumentException {
```

```
        "Son's age must be less than father's &
```

```
        this. sonAge - sonAge;
```

```
    }
```

```
public int getSonAge() {
```

```
    return sonAge;
```

```
}
```

```
}
```

```
public class Main {
```

```
    public static void main(String args[]) {
```

```
        try {
```

```
            Father father = new Father(50);
```

```
            Son son = new Son(50, 25);
```

```
            System.out.println("Father's age: " + father
```

```
                .getAge());
```

```
            System.out.println("Son's age: " + son.getSonAge());
```

```
        } catch (WrongAge e) {
```

```
            System.out.println("Exception: " + e.getMessage());
```

```
        } catch (IllegalArgumentException e) {
```

```
            System.out.println("Exception: " + e.getMessage());
```

```
        } catch (IllegalArgumentException e) {
```

```
            System.out.println("Exception: " + e.getMessage());
```

```
        }
```

```
    }
```

```
}
```


24

Fathers Age 30

Sons age 25