

Yash Raj Rawat  
243920037  
Batch 15B D.S

## Assignment 2.3

Input java.util.\*;

try InvalidMarksException extends Exception {

public InvalidMarksException (String message) {  
super(message);  
}

class Student {

private int rollNumber;  
private String studentName;  
private int marks[];

public Student (int rollNumber, String studentName, int[] marks)

this.rollNumber = rollNumber;  
this.studentName = studentName;  
this.marks = new int[3];  
System.arraycopy(marks, 0, this.marks, 0, 3);

public void validateMarks () throws InvalidMarksException {

for (int i = 0; i < marks.length; i++) {  
if (marks[i] < 0 || marks[i] > 100) {  
throw new InvalidMarksException ("Marks [" + i + "] > 100");  
}  
}

```
+ ( i + 1 ) + " : " + marks [ i ] ) ;
```

```
}
```

```
} public double calculate Average () {
```

```
double sum = 0.0 ;  
for ( int m : marks )  
sum += m ;  
return sum / marks . length ;
```

```
} public String get Result String () {
```

```
double avg ; calculate Average () ;
```

```
for ( int m : marks ) {
```

```
if ( avg > m )  
return " Paid " ;
```

```
}
```

```
return ( avg > 70.0 ) ? " Pass " : " Fail " ;
```

```
} public int get RollNumber () {  
return RollNumber ;
```

```
}
```

```

public void displayResult() {
    System.out.print("Roll number : " + rollnumber);
    System.out.println(" Student name : " + studentname);
    System.out.print(" Marks : ");
    for (int i = 0; i < marks.length; i++) {
        System.out.print(marks[i] + (" " + marks.length));
        System.out.println();
    }
    System.out.println(" Average : " + calculateAverage());
    System.out.println(" Result : " + getFinalGrade());
}

public class Student StudentManagement {
    private Student > students;
    private int studentCount;
    private Name Name;
    public Student Manager (int capacity) {
        this.students = new Student[capacity];
        this.Name = new Name(capacity);
    }
}

```

plus studentName : D  
thisName : new Name (by user input),

public void addStudent(), throws InvalidMarksException {

try {  
 System.out.print("Enter Roll number: ");  
 int roll = scanner.nextInt();

Scanner mention();

System.out.print("Enter Student Name: ");  
String name = scanner.nextLine();  
int id = new int[3];

for (int i = 0; i < 3; i++) {

System.out.print("Enter marks for subject " + (i + 1) + ": ");

marks[i] = scanner.nextInt();

},  
student S = new Student (roll, name, marks);

S.validateMarks();

```

if (studentCount < students.length) {
    students[studentCount] = System.out.println("Student number " + studentCount + " has been added");
    System.out.println("Current number of students: " + studentCount);
}

else {
    System.out.println("Student already exists");
}
}

catch (Exception e) {
    e.printStackTrace();
}

}

public void showStudentDetails() {
    System.out.print("Enter Roll number: ");
    int roll = scanner.nextInt();
    for (int i = 0; i < students.length; i++) {
        if (students[i].getRollNumber() == roll) {
            System.out.println(students[i].toString());
        }
    }
}

```

void() display(){

font("serif");

stroke;

stroke

public void draw(){

if ( scanner == null ) {

Scanner scan =

new

Scanner();

while ( ! scan.hasNextString( "string" ) ) {

String str = scan.nextLine();

try {

Message message =

freely S

Message box =

capture( str );

System.out.println( "Name: " + message );