Decision making and branching

- normally programs are executed sequentially.
- there are situations where we need to change the order of execution.
- then program breaks sequential execution and jump to a new section, this is called branching.

Decision making with IF statements.

- it is a two way statement.
- used in conjunction with an expression.

Form 1:-

```
if (<expression>){
        <statement >;
        <statement >;
        …..
        }
else {
        <statement >;
        <statement >;
        …..
        }
```

Form 2:- Nesting of IF statements

```
if (<expression 1>){
        if (<expression 2>){
                <statement >;
                <statement >;
                …..
                }
        else {
                <statement >;
                <statement >;
                …..
                }
        }
else {
        if (<expression 3>){
                <statement >;
                <statement >;
                …..
                }
        else {
                <statement >;
                <statement >;
                …..
                }
        }
```

Form 3:- IF-ELSE-IF ladder

```
if (<expression 1>){
        <statement >;
        <statement >;
        …..
        }
else if (<expression 2>){
        <statement >;
        <statement >;
        …..
        }
        else if (<expression 3>){
                <statement>;
                }
        else{
                <statement>;
                }
```

Decision making with SWITCH statements.

- used when one of the many alternatives is to be stated.
- can be converted into IF-ELSE structure, but the complexity will be increased.

```
switch (<expression>){
        case <val 1>:
                <statement >;
                <statement >;
                …..
                break;
        case <val 2>:
                <statement >;
                <statement >;
                …..
                break;
        case <val 3>:
                <statement >;
                <statement >;
                …..
                break;
        case <val 4>:
                <statement >;
                <statement >;
                …..
                break;
        default:
                <statement >;
                <statement >;
                …..
                }
```

eg:-

```
int x = 10;
switch (x){
        case 35:
                System.out.println("lpo");
                System.out.println("lponihbhbhu");
                break;
        case 110:
                System.out.println("lpooijoj");
                System.out.println("lponihbhbhu");
                break;
        case 15:
                System.out.println("lainipo");
                System.out.println("lponihbhbhu");
                break;
        case 25:
                System.out.println("lpojoijo");
                System.out.println("lponihbhbhu");
                break;
        default:
                System.out.println("lpjnhibherbhwo");
                System.out.println("lpihbhjbhjbonihbhbhu");
                }
```

note that:

- <expression> is an integer or a character
- <val 1>, <val 2>,… are constants
- break statement prevents the SWITCH statement getting cascaded

Home work:

1. write a program to find the number of integers and sum of integers those are greater than 100 and less than 200, and that are divisible by 7.
2. given a list of marks ranging from 0-100, write a program to compute and print the number of students who have obtained marks in the range 100-81,80-61,60-41 and 40-0.