Nested if statement

Nested if statement means if within if called as nested if statement.
When we have multiple conditions and some conditions are dependent on some another condition then we can use nested if statement.

Syntax of nested if statement

```
if(condition)
{
      if(condition)
      {
            if(   condition )
            {

            }
           else{

           }
      }
      else
      {

      }
}
else
{

}
```

Example: WAP to input the id of employee and if between 1 to 100 then input salary of employee and if salary amount is greater than 10000 then consider it is permanent employee otherwise consider it is temporary employee and if id not between 1 to 100 then candidate is not employee.


```
import java.util.*;
public class TestEmployeeApp
{
    public static void main(String x[])
    {
        int id,sal;
        Scanner xyz  = new Scanner(System.in);
        System.out.println("Enter id of employee");
        id=xyz.nextInt(); //10 1

        if(id>=1 && id<=100)
        {
            System.out.println("Enter salary of employee");
            sal=xyz.nextInt();

            if(      sal>=10000     )
            {
                System.out.println("Candidate is permenent");
            }
            else
            {
                System.out.println("Candidate is temporary");
            }
        }
        else
        { System.out.println("Candidate is not employee");
        }
    }
}
```

Example: WAP to input three values and check which is greater using nested if statement.

```
public class CTGNF
{     public static void main(String x[])
     {  int a,b,c;
        Scanner xyz  = new Scanner(System.in);
        System.out.println("Enter three values");
        a=xyz.nextInt();//10
        b=xyz.nextInt();//50
        c=xyz.nextInt();//80

        if(a>b)
        {
            if(a>c)
            { System.out.println("A is Greater");
            }
            else{
              System.out.println("C is Greater");
            }
        }
        else
        {   if( b > c)
            { System.out.println("B is Greater");
            }
            else{
               System.out.println("C is Greater");
             }
        }
     }
}
```

Switch statement

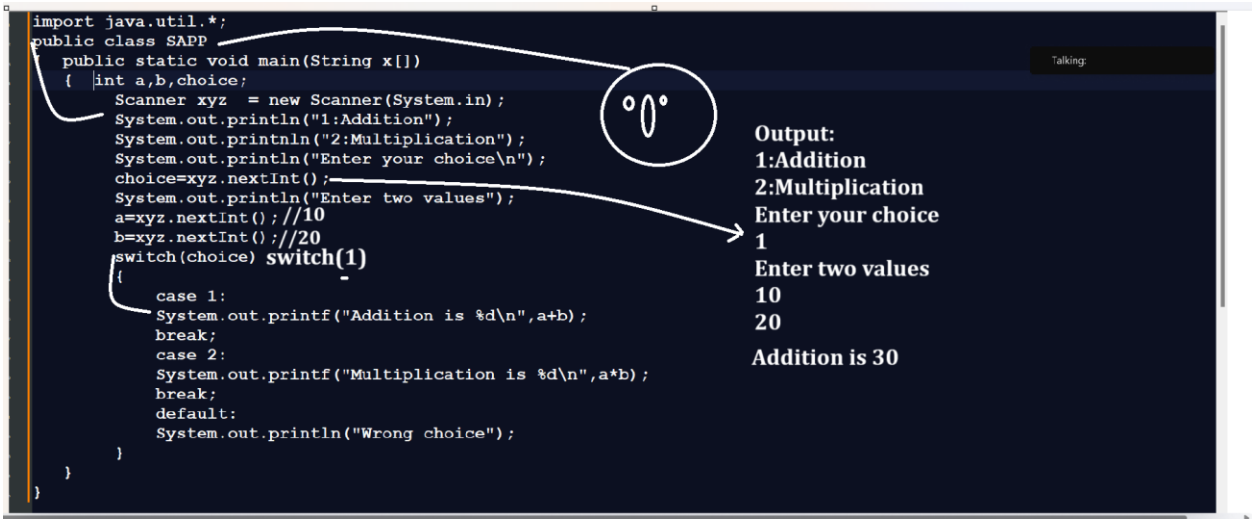
Switch statement is used for create menu driven application or choice based application

Syntax: switch(choice)

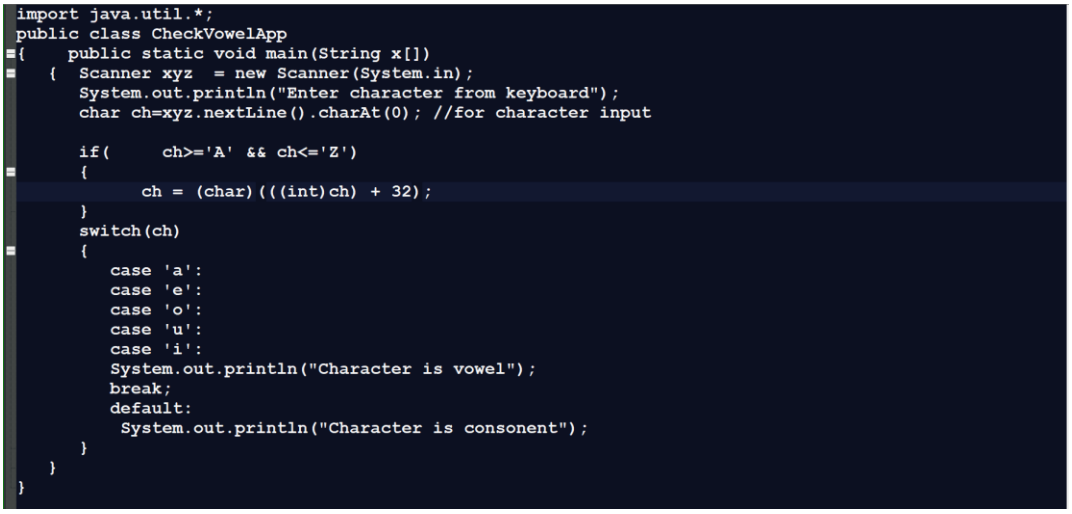
```
{
    case 1:
     write here your logics
     break;
    case 2:
    write here your logics
   break;
   case 3:
    write here your logics
   break;
  default:
   wrong choice
}
```

Note: you can give integer, character or string as choice but we cannot give float or boolean value as choice

Example: we want to input two values and when user press 1 then calculate addition and when user press 2 then calculate multiplication



Example: WAP to input character from keyboard and check character is consonant or vowels.



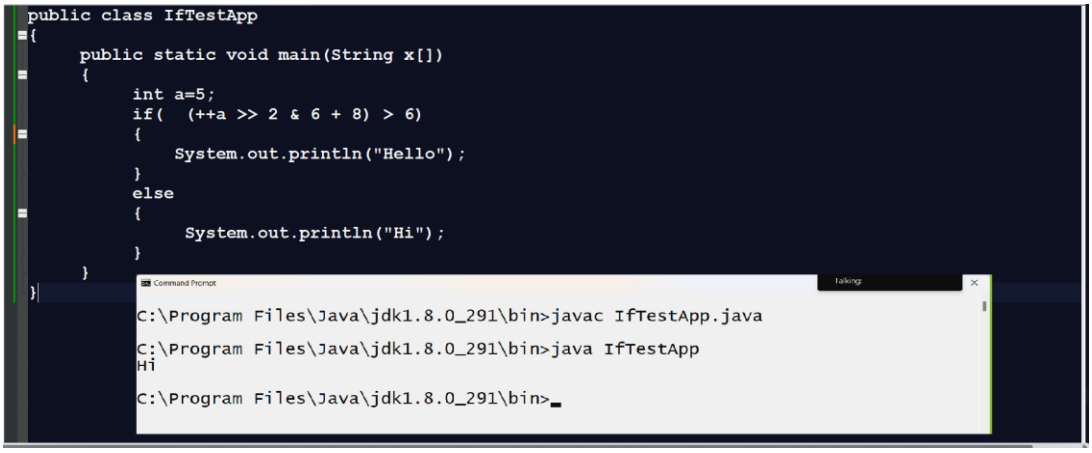
Example: WAP to input quantity and rate from keyboard and when user press 1 then calculate bill without gst and when user press 2 then calculate bill with gst 18%

Example with source code

```
import java.util.*;
public class BillWithSwitchApp
{  public static void main(String x[])
        {  Scanner xyz  = new Scanner(System.in);
           int qty,rate,bill,choice;
           System.out.println("1:Bill With GST");
           System.out.println("2:Bill Without GST");
           System.out.println("Enter your choice");
           choice=xyz.nextInt();
```

```
            System.out.println("Enter quantity and rate of product");
            qty=xyz.nextInt();
            rate=xyz.nextInt();
            switch(choice)
            {
                    case 1:
                             bill=(qty*rate)+((qty*rate)*18/100);
                            System.out.printf("Bill With GST %d\n",bill);
                            break;
                            case 2:
                            bill=qty*rate;
                            System.out.printf("Bill Without GST %d\n",bill);
                            break;
                            default:
                            System.out.println("Wrong choice");
            }
          }
}
```

Example: what will be output of given code?



Example: Solve the following problem statement

A washing machine works on the principle of Fuzzy System, the weight of clothes put inside it for washing is uncertain But based on weight measured by sensors, it decides time and water level which can be changed by menus given on the machine control area.For low level water, the time estimate is 25 minutes, where approximately weight is between 2000 grams or any nonzero positive number below that. //2-25
For medium level water, the time estimate is 35 minutes, where approximately weight is between 2001 grams and 4000 grams.
For high level water, the time estimate is 45 minutes, where approximately weight is above 4000 grams.

Assume the capacity of machine is maximum 7000 grams
Where approximately weight is zero, time estimate is 0 minutes.
Write a function which takes a numeric weight in the range [0,7000]
as input and produces estimated time as output is: “OVERLOADED”, and for
all other inputs, the output statement is
“INVALID INPUT”.
Input should be in the form of integer value –
Output must have the following format –
Time Estimated: Minutes
Example:

- Input value

2000

- Output value

Time Estimated: 25 minutes

Output

```
import java.util.*;
public class WMAPP
{ public static void main(String x[])
        { Scanner xyz  = new Scanner(System.in);
          int w;
          System.out.println("Enter weight from keyboard");
          w=xyz.nextInt();
          if(w>7000)
          { System.out.println("Machine is overloaded");
          }
          else if(w==0)
          { System.out.println("Machine required 0 minute");
          }
          else if(w>=1 && w<=2000)
          { System.out.println("Machine required 25 minutes");
          }
          else if(w>=2001 && w<=4000)
          { System.out.println("Machine required 35 minutes");
          }
          else if(w>=4001 && w<=7000)
          { System.out.println("Machine required 45 minutes");
          }
          else{
            System.out.println("Invalid inputs");
          }
        }
}
```