Limitation of if-else condition :

Example :

```
            int x = 90;

            if(x==10)
            {
            }
            else if(x==20)
            {
            }
            else if(x==30)
            {
            }
            else if(x==40)
            {
            }
            ...........
```

Limitation of if else:

The major drawback with if condition is, it checks the condition **again and again** so It increases the burden over CPU so we introduced **switch-case statement** to reduce the overhead of the CPU.

Switch Statement in java (Till JDK 17V) :

switch case syntax :

Old approach of writing switch case :

```
var variable = 101 ;

switch(variable)
{
  case 101 :
       //Statement to be executed;
  break;

  case 102 :
       //Statement to be executed;
  break;

  case 103 :
       //Statement to be executed;
  break;
  default :
      //Will be executed, if no matching
      //case is available
}
```

new approach of writing switch case :

```
switch(variable)
{
  case 101 ->  IO.println("Case 101");

  case 102 -> IO.println("Case 102");

  case 103 -> IO.println("Case 103");

  default ->  IO.println("Default case");

 }
```

: and break both are replace by arraw symbol

* It is a selective statement in java so, we can select one case among the available cases.

While writing the cases, break is optional but if we use break then the control will move from out of the switch body.[old style]. In switch case break is required.

We can write default so if any statement is not matching then default will be executed but It is **optional.**

In the new switch case break and : both are not required, We can replace with **->** symbol

In the new switch case, if we have multiple statements then {} are compulsory.

In the new switch case, default is compulsory, If we take switch statement as an **expression**

The label of switch must be constant expression. [100+2 ,   final int x = 101]

Types allowed while working with switch case :

byte
short
char
int
Byte
Short
Character
Integer
String [Allowed from JDK 1.7v]
enum [Allowed from JDK 1.5V]

Not allowed :

long
float
double
boolean

//Programs :

Basic old style of switch case program :

```
void main()
{
 var variable = 101 ;

switch(variable)
{
  case 101 :
       IO.println("Case 101");
  break;

  case 102 :
        IO.println("Case 102");
  break;
  case 103 :
       IO.println("Case 103");
  break;
  default :
       IO.println("Default case");
  }
  IO.println("Every time It will be executed");
}
```

Note : In the old style : and break both are required to execute a prticular statement.

New switch case :

```
 /* Write a switch case program where based on the first character
 select the color */

 void main()
 {

  String str = IO.readln("Enter the color name :");
  char color = str.toLowerCase().charAt(0);

  switch(color)
  {
       case 'r' -> IO.println("Red Color");
       case 'b' -> IO.println("Blue Color");
       case 'g' -> IO.println("Green Color");
       case 'w' -> IO.println("white Color");
       default -> IO.println("No color");
  }
  IO.println("This line will be executed everytime");
 }

  //Menu based switch case program :

  void main()
  {
           IO.println("\t\t**Main Menu**\n");
           IO.println("\t\t**100 Police**\n");
           IO.println("\t\t**101 Fire**\n");
           IO.println("\t\t**102 Ambulance**\n");
           IO.println("\t\t**139 Railway**\n");
           IO.println("\t\t**181 Women's Helpline**\n");

           IO.print("Enter your choice :");
           int choice = Integer.parseInt(IO.readln());

           switch(choice)
           {
           case 100 -> IO.println("Police Services");

           case 101 -> IO.println("Fire Services");

           case 102 -> IO.println("Ambulance Services");

           case 139 -> IO.println("Railway Enquiry");

           case 181 -> IO.println("Women's Helpline ");

           default ->    IO.println("Your choice is wrong");
           }
    }
```

Take String (immutable) as a switch case :

```
    void main()
    {
        IO.print("Enter the name of the season :");
        String season = IO.readln().toUpperCase();

        switch(season)
        {
             case "SUMMER" -> IO.println("Summer season");
             case "WINTER" -> IO.println("Winter season");
             case "SPRING" -> IO.println("Spring season");
             case "RAINY" -> IO.println("Rainy season");
             default -> IO.println("No valid season");
        }

    }
```