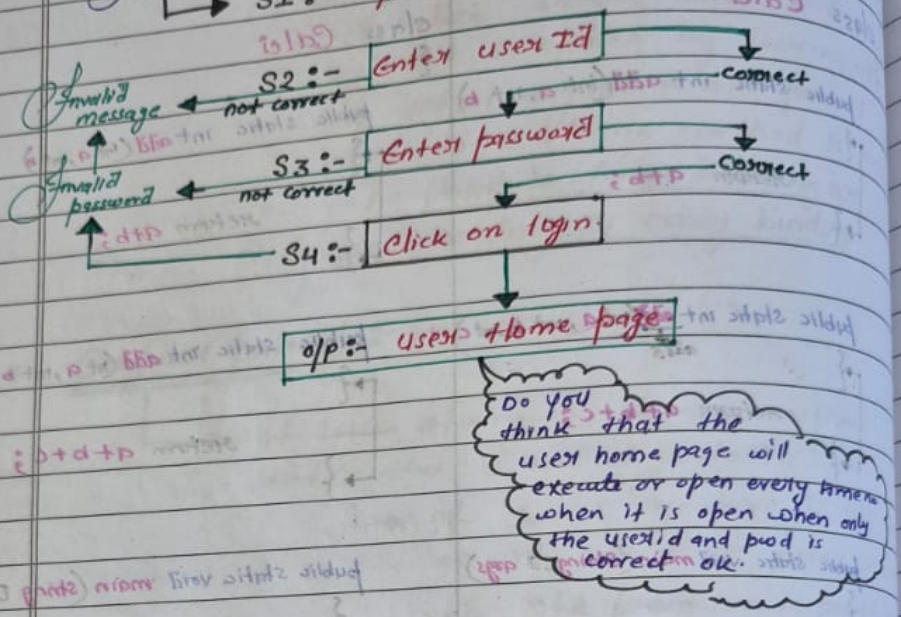
Topic :- Decision Statement :-

" Facebook Example "

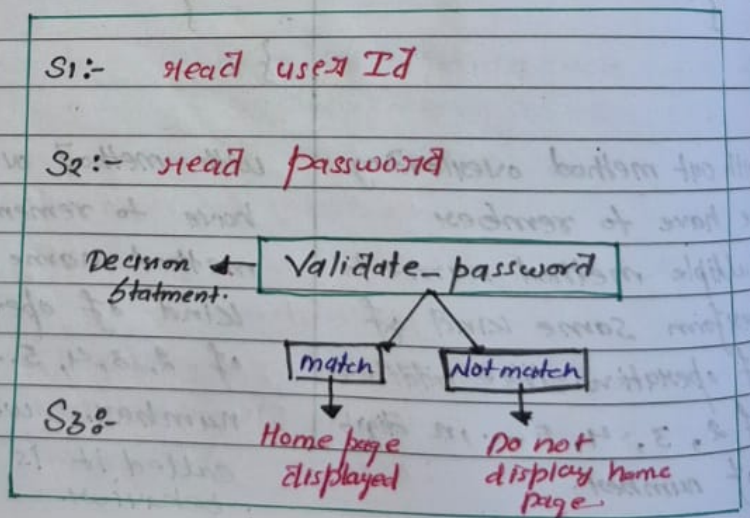
S1 :- open Fb



Do you think that the user home page will execute or open every time when it is open when only the userid and pwd is correct ok.

Q) why we need Decision Statement?

→ Decision statment helps the programer to decied whether a block of instructions has to be execute or skiped.



PAGE :
DATE : / /

Topic :- Types of Decision Statment.

(I) → if
(II) → if else
(III) → else if ladder
(IV) → Switch

if :-

it is a keyword, used as a decission statment.

Syntax :-

```
if(condition)
    statment;
```

→ if we have only one statment then we can write without curly braces.

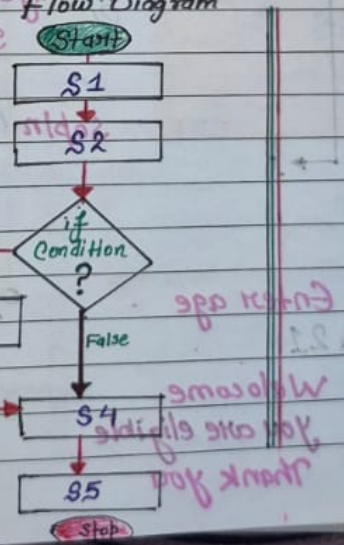
```
if(Condition)
{
    statment;
    statment;
}
```

→ Example :-

```
{
(I)   S1;
(II)  S2;
(III) if(Condition)
      {
(IV)    true  False
        S3;
      }
(V)   S4;
(VI)  S5;
}
```

S3 is skipped

Flow Diagram



PAGE :
DATE :

Topic :- Program on if :-

Case 1 :- age = 21

O/P :- Hello
you are eligible
Thank you

Case 2 :- age = 16

O/P :- Hello
Thank you

Note :- age >= 18

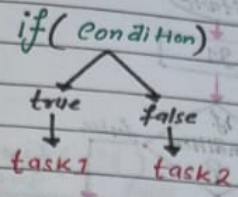
```
import java.util.Scanner;
class Program
{
    public static void main(String [] args)
    {
        Scanner sc = new Scanner();
        Sopln ("Enter age");
        int age = sc.nextInt();
        Sopln ("Welcome");
        if(age >= 18)
            Sopln ("you are eligible");
        Sopln ("Thank you");
    }
}
```

O/P :- Enter age
21
Welcome
you are eligible
Thank you

O/P :- Enter age
16
Welcome
Thank you

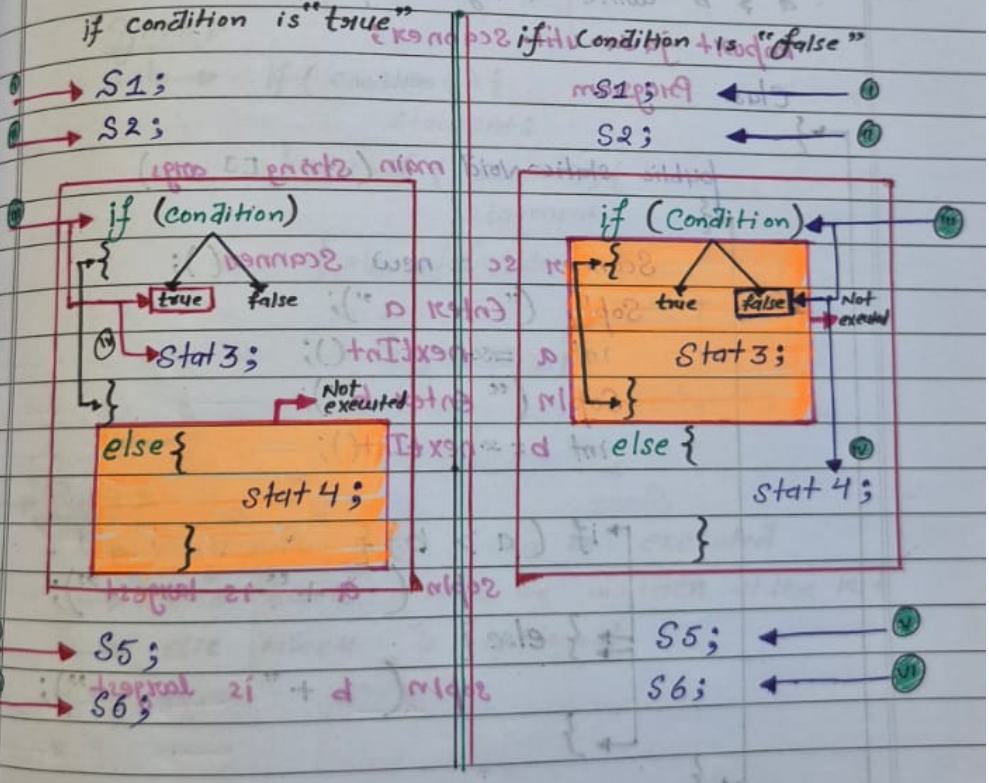
PAGE :
DATE : / /

if else :-



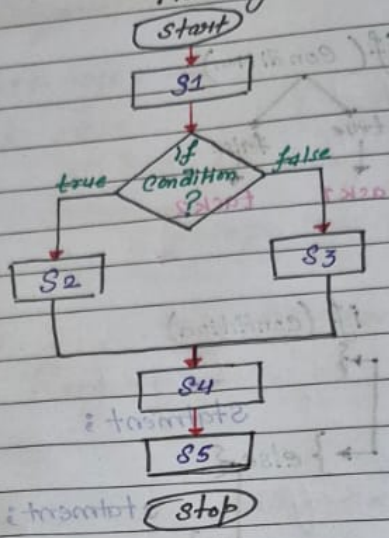
Syntax :-

```
if (condition)
{
    statment;
} else {
    statment;
}
```



PAGE :
DATE :

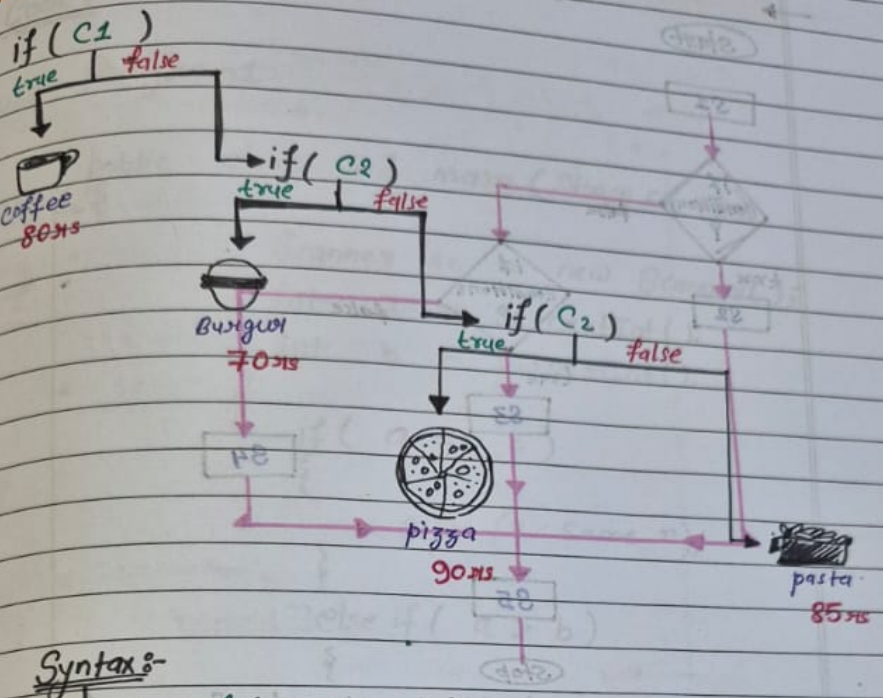
Flow Digram



→ Assume we have 2 numbers stored in container a & b write a logic to print largest no

```
import java.util.Scanner;
Class Program
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner();
        Sopln ("Enter a");
        int a =sc.nextInt();
        Sopln (" Enter b");
        int b= sc.nextInt();
        if (a > b){
            sopln( a + " is largest");
        } else {
            sopln( b + " is largest");
        }
    }
}
```

if else if ladder :-



Syntax :-

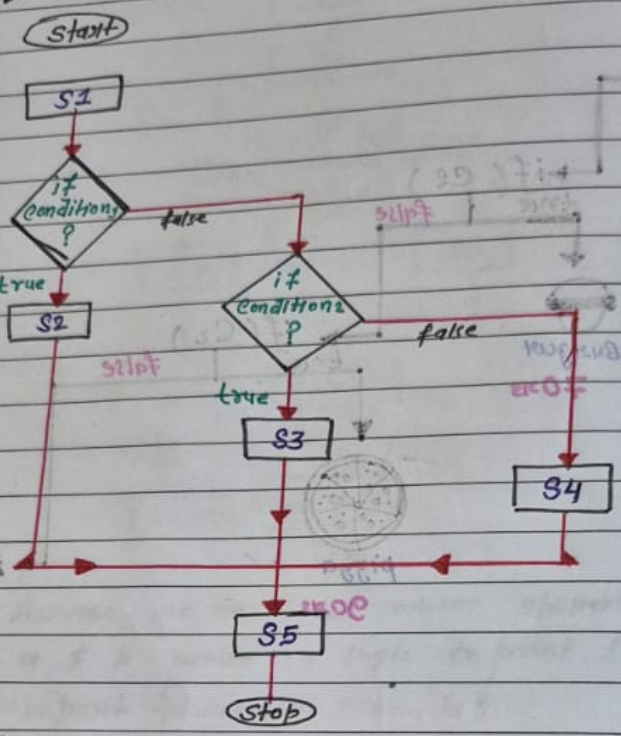
```
if ( condition 1 ) {
    statment ;
} else if ( condition 2 ) {
    statment ;
} else if ( condition 3 ) {
    statment ;
} else {
    statment ;
}
```

Rules :-

- Only one Block can be executed
- else block should be written at the last
- else block is optional

PAGE :
DATE : / /

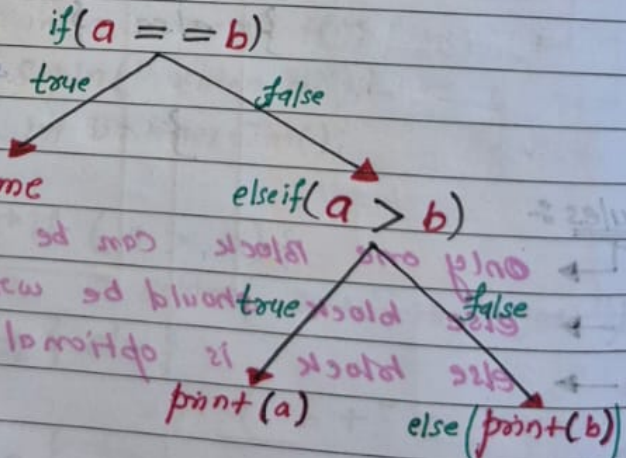
Flow Diagram



→ Assume we have 2 numbers stored in a & b, write a logic to check whether the data is same, if not print the largest.

Logic :-

a [ ] b [ ]



PAGE :
DATE : / /

Code 8-

```
class Program1
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner();
        int a = sc.nextInt();
        int b = sc.nextInt();
        if ( a == b )
        {
            System.out.println(" same ");
        }
        else if ( a > b )
        {
            System.out.println(" a is bigger a =" + a);
        }
        else
        {
            System.out.println(" b is bigger b =" + b);
        }
    }
}
```

o/p1:- a=10 b=10 o/p2:- a=20 b=10 o/p3:- a=10 b=20

Same a is bigger b is bigger
a = 20 b = 20

PAGE :
DATE : / /

Switch :-

Syntax :-

switch (value / variable / expression)
↳ we can't pass condition here.....

{
case value/exp : {
statments;
} [break;] → optional
↳ we can write inner block or out side the block

case value/exp :
statment;
[break;] →

default : {
statments; → optional
} [break;]

}

Rules :-

switch( )

→ Inside the switch paranthesis we can give the value of datatype such as byte, int, short, char, string, enum, pattern → long → error Lossy conversion long to int

case :

→ Inside the case we can pass only Constants. Variable is not allowed like 1 ✓, 1+2 ✓, a ✗, a+2 ✗, 'a' ✓, "hello" ✓

↳ if we give variable here.

error
↳ Constant expression required.

Topic: Work Flow of Switch Without Break Statment.

Example 1:-
→ Without break statment.

```
int a = ;
Switch ( a )
{
    case 1:
        Sopln ("Hello");
    Case 2:
        Sopln ("Hii");
}
```

O/P:- a = 2
Hii

O/P:- a = 1
Hello
Hii
} if a case is matching the case block will get executed and all the remaining case block present below the selected case get executed. including default block

* break:-
→ It is a keyword.
→ It is control transfer statment.
→ It can be used only inside
→ switch
→ loop

PAGE :
DATE : / /

Example2:- with break keyword.

```
int a = ;
switch (a)
{
    case 1:
        sopln("case1");
        break;
    case 2:{
        sopln("case 2");
        break;
    }
    case 3:{
        sopln("Case 3")
    }break;
    case 4: {
        sopln("last");
        sopln("case 4");
    }
    break;
}
```

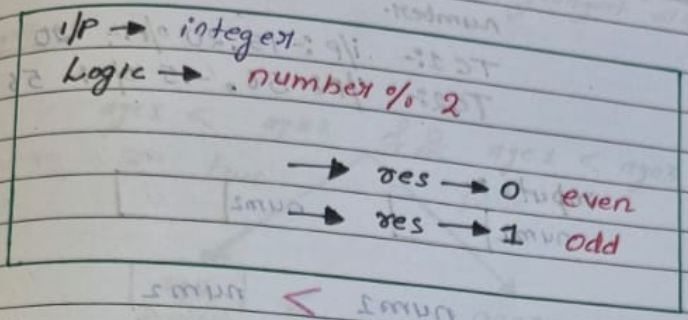
O/P:-
a = 1
case 1

O/P :-
a = 3
Case 3
last
case 4

O/P:-
a = 4

Topic:- Program on switch :-

* Designe a program to check whether a number is even or odd. using switch.



code:-

```
import java.util.Scanner;

Class Program1
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner();
        sopln("Enter a number");
            int num = sc.nextInt();

        switch (num%2)
        {
            case 0 : sopln("even");
                break;

            case 1 :
                sopln("odd");
        }
    }
}
```

It will work for only positive number

for negative number also use Math.abs(num%2)

It will work for both +ve as well as -ve.

Program 2.b :-

Suppose you want to create a program that greets the user based on time of the day. Implement a program that takes the current hour as input and outputs a greeting message accordingly.

Note : take the time in 24-hour format.
5 to 11 : "Good Morning!"
12 to 16 : "Good Afternoon"
17 to 20 : "Good Evening"
21 to 4 : "Good Night"

TC1:- i/p:- 11 o/p:- Good Morning.
TC2:- i/p:- 12 o/p:- Good Afternoon
TC3:- i/p:- 21 o/p:- Good Night
TC4:- i/p:- 25 o/p :- Invalid Time.

→Code:-

```
int hour = s.nextInt();
if (hour >= 5 && hour <= 11){
        sopln("Good Morning");
}
elseif (hour >=12 && hour <= 16){
        sopln("Good Afternoon");
}
else if (hour >= 17 && hour <= 20){
        sopln("Good Evening");
}
else if (hour >= 21 && hour <= 24){ // all hour>=0 && hour<=4
        sopln("Good Night");
}else{
        sopln("Invalid Hour");
}
```

PAGE :
DATE : / /

Program 3.a :-

A school follows a day wise time table for its student to play a specific game.

Time Table :-

Monday : Basket Ball
Tuesday : Tennis
Wednesday : Basket Ball
Thursday : Cricket
Friday : Tennis
Saturday : Exercise
Sunday : Holiday

Design a simple application which can read the day of week and suggest to the student which game is supposed to play today by following the prescribed time table designed by the School.

TC1 :- i/p :- Monday o/p :- Basket Ball
TC2 :- i/p :- Thursday o/p :- Cricket
TC3 :- i/p :- Tuesdya o/p :- Invalid Input.