Conditional Statement-> It is not possible to store in a variable.

//Wrong na to if ko na hee switch ko store kar sakte hai kisi varible me
int a= if(condition){
// your code
}

Conditional Statement-> To run the code based on the conditional Statement.

Types-> IF AND SWITCH.

IF(Normal If, If-else, If-else ladder)

//Normal If
if(Condition){
// your code/ logic jo condition true hone par aap run karna chahte hai
}

//If-else
// else should be immediate of if(if ka block end hote hee aapka else aana chaheaye)
if(Condition){
// your code/ logic jo condition true hone par aap run karna chahte hai
}
else{
//agar condition wrong hui us case me else vala code run hota hai
}

//If-else-ladder(Nested If)
if(Condition){
// your code/ logic jo condition true hone par aap run karna chahte hai
if(Condition){
// your code/ logic jo condition true hone par aap run karna chahte hai
}
else{
}

}
else if(Condition){
// your code/ logic jo condition true hone par aap run karna chahte hai

}
else{

}

switch (Normal)->
//Code readable banana ke leaye aae hai

// do not use any decimal value
switch(character/numeric){
//Write your cases
case caseNo: //your logic

break;//optional

default : // jab koi bhi case match nhi hoga tab aapka default vala case run hoga

}

Ternary Expression-> to evaluate one line expression result.

int/float/double/string variableName=condition ? (your value it can be anything jo condition true hone par vbarible store hogi (no., string, character, Boolean):(your value it can be anything jo condition wrong hone par kisi variable ke andar store hogi (no., string, character, Boolean)

Condition done ke leaye conditional operator hote hai->
<,>,<=,>=,==,!=(Boolean result date hai that means true(1)/ false(0)

agar multiple condition dena hai if ke andar us us time par aapke pass logical operator hote hai

5==10// ek condition ke base me if ko chala sakta hu

6<=90 || 7>=80// more than one condition hai to yha par logical operator ka use keaya jata hai
Logical Operator(AND(&&), OR(||), Not(!)

Loops -> which is used to perform a specific code in metion iteration
(loop jo hota hai vo kisi kaam ko jitne aapne iteration/round deaye hai utne bar karata hai)

//LOOP me 3 cheej hoti hai
initialization
condition
updation

Syntex of for loop

for(initialize; condition; updation){

// jo kaam aapko repeatly karana hai
}
ex.
for(int i=1;i<=6;i++){
cout<<"Dsa "<<i<<"Round"<<endl;
}

Types of Loops-
for
while
do while

for(int i=0;i<=4;i++){

}

1. only1st time hee initialization vala part chalta hai
2. condition hamesha check hoti hai(chahe vha 1st time ya last time)
3. updation condition ko false karne ke leaye likha jata hai(esme aap
increment(pre/post increment) ya decrement(pre/post decrement karte hai)

for(int i=0;i<4;i++){
cout<<" :-"<<i<<endl;
}

i
0

FOR LOOP/WHILE LOOP (ke case me ese excute hote hai)

1st round me initialization vala part check hota hai uske bad condtion check hoti hai
a. i=0

b. i<4 (0<4) agar ye condition shi hui to hee aap loop ke andar jayenge
(true)// loop ke andar aa gaye

c. aap loop me i ko print kra rhe hai so yha par 0 print ho jayegaya

d. eske bad for loop ka block end ho jayega lekin bahar nikalne se pahle vo condition ko dekhega agar vo wrong nhi hai to vah updation vale part me jayega

(i++) so aapne i++ keaya hai i=i+1 ab i kee value 1 ho gayi hai eske bad direct vo 2nd round jayega

or 2nd round me codition ko check karega ab kabhi vah initialization vale part ko check nhi kearega

If condtion true so loop ke anadar or code excute karke same again and again karta rhega

same for 3, 4 or every jab tak condition galt na ho jaye

Nested Loop ke case me bhi same steps hoti hai same bahar loop ke leaye hota hai and same step andar loop ke leaye hoti hai.

Kisi bhi code kee time complexity us code ke andar likhe gaye loops me depend kartI hai

1. agar ek loop and other loop same level me hai (means nested nhi hai us case me time complexity add hoti hai)

2. agar ek loop ke andar dusra loop hai (means nested hai us case me time complexity multiply hoti hai)