Condional and loops :

Condition :- It provides check for the statement

- 1) If -else statement.
- 2) else if statement

Loops :- Loops are used to iterate a part of program several times.

1) ~~for~~ loop :- used when we know how many times the loop will iterate.

2) While loop :- used when we dont know no. of times the loop will iterate.

3) Do while loop :- used when we want to execute ~~our~~ code at lease ones.

Difference b/w While loop & Do while loop

While loop	Do while loop
- Used when no. of iteration is not fixed	- Used when we want to execute at least ones
- entry controlled loop	- exit controlled loop
- no semicolon required at the end of while.	- semicolon required at the end of while.

Loop :- A loop is a control structure that allows to repeate a block of code multiple times as long as ~~condit~~ given a condition is true.

1.) for loop :- used when no/. of iderations are known.

Syantax :-

```
        for (int i (initialization); i<=5 (condition),
                                   i++ (update))
        {
           //(code block)
        }
```

2) While loop :- used when no% of Iterations is unknown but condition is provided.

Syantax :-

```
     while (condition) {
         // code
     }
```

⇒ Do-while loop :- Used when loop is requred to run atleast once

Syantax :-

```
     do {
        // code
     } while (condition);
```

Date ______
Page ______

Switch :

- Switch is a control flow statement used to execute one block of code anon among many case based on value of (variable/expression).
- Used as an alternative of if-else-if statement

Syantax :-

```
switch (expression) {
        case one :
              // code
              break;   ⟶  terminate the sequence
     case two :
            // code
            break;
     default :              ]→ default will be executed when
           // code:             none of the above case is
   }                            satisfied.
```

→ If break is not used then it will continue with other cases.

→ duplicate cases are not allowed.

★ x.equals ("word") ⟶ it checks value not reference

★ x == "word" ⟶ here it checks references.

New Syantax :—

```
switch (expression) {
        case one → //do this ;
        case two → // do this;
        case three → // do this;
        defaut → //do this;
        }
```

Nested Switch case :— placing switch statement inside another. It allow decision making at multiple

Syantax :—

```
switch (expression) {
    case one :
          //code block
          break ;
    case two :
          switch (expressien){
                case 1:
                      // code block
                      break;
                case 2:
                      // code block
                      break;
                default;
          }
          break;
case three :
        // code block
        break;
defaut :
        //code block
};
```