1. CONDITIONAL OPERATORS:

They are used when a condition comprises more than one Boolean expression. For instance, if we want to print a number Only if it is greater than 2 and less than 5, then we will use conditional operators to combine the 2 expressions we have 3 types of conditional operators- logical-and, logical-or and ternary operator.

2. TYPES OF OPERATORS

- Arithmetic operator
- Relational operator
- logical operators
- Assignment operators
- Unary operators
- Bitwise [operators

3. SWITCH CASE

It is like an if-else ladder with multiple conditions, where we check for equality of a variable with various values.

```
switch (expression)
  {
    case x:{ //CODE}
    break;
    case y:{//CODE}
     break;
    default:{//CODE}
  }
```

4. Priority levels of arithmetic operation

Precedence	Operator	Type	Associativity
15	() [] .	Parentheses Array subscript Member selection	Left to Right
14	++ --	Unary post-increment Unary post-decrement	Right to left
13	++ -- + - ! ~ (type)	Unary pre-increment Unary pre-decrement Unary plus Unary minus Unary logical negation Unary bitwise complement Unary type cast	Right to left
12	* / %	Multiplication Division Modulus	Left to right
11	+ -	Addition Subtraction	Left to right
10	<< >> >>>	Bitwise left shift Bitwise right shift with sign extension Bitwise right shift with zero extension	Left to right
9	< <= > >= instanceof	Relational less than Relational less than or equal Relational greater than Relational greater than or equal Type comparison (objects only)	Left to right
8	== !=	Relational is equal to Relational is not equal to	Left to right
7	&	Bitwise AND	Left to right
6	^	Bitwise exclusive OR	Left to right
5	|	Bitwise inclusive OR	Left to right
4	&&	Logical AND	Left to right
3	||	Logical OR	Left to right
2	? :	Ternary conditional	Right to left

1	= += -= *= /= %=	Assignment Addition assignment Subtraction assignment Multiplication assignment Division assignment Modulus assignment	Right to left

5. CONDITIONAL STATEMENTS

Java follows suit by using conditional statements to manage the program's flow.This is crucial because at some time, in order to advance with our code, we must fulfil requirements. For example, the console's yes or no input will determine whether the programme is to be continued or cancelled.

- If statement
- else statement
- If-else Ternary
- Nested if-else
- Switch statement

6. SYNTAX OF IF ELSE

```
If(condition)
   {
      //code
    }
Else
   {
     //code
   }
```

7. THREE TYPES OF ITERATIVE STATEMENTS

- FOR LOOP
- WHILE LOOP
- DO WHILE

8. FOR LOOP

```
for(initialisation;condition_check;update)
   {
     //code
   }
```

DO WHILE LOOP

```
do
  {
    //code
  }
 While(condition)
```

In, “for loop” first the condition will be checked if it matches then only the body will be executed otherwise not

But, in “do while” loop once the body will be executed then it matches the condition.

9. class Print

```
public static void main(String[] args)
    {
       for(int i=1;i<=10;i++)
          {
            System.out.println(i)
          }
     }
```