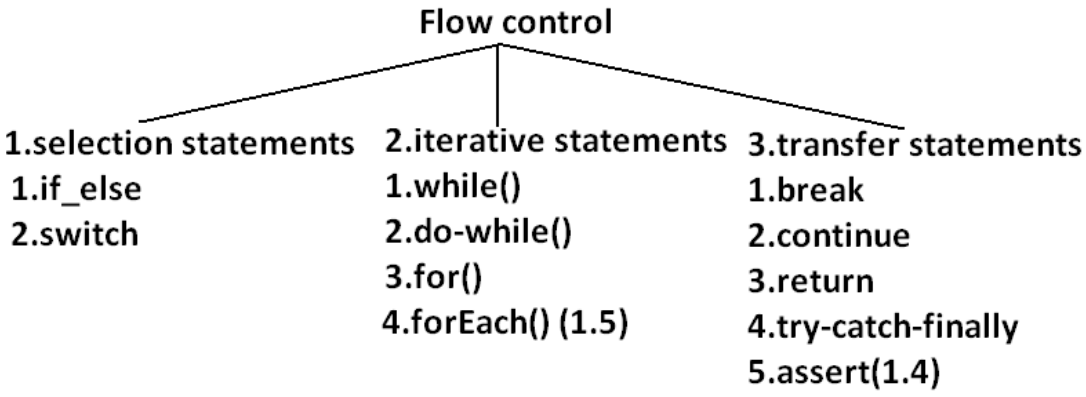
Control Flow

Control flow indicates the sequence in which the statements in a java program will be executed in run time.

Control flow statements in Java:



11- WAP to demonstrate simple if

```
public class ExampleIf {
   public static void main(String[] args) {
       int testScore = 95;
       if (testScore >= 40) {
           System.out.println("You passed the exam!");
       }
   }
}
```

Output:

You passed the exam!

11- WAP to demonstrate if else

```
public class ExampleIfElse {
    public static void main(String[] args) {
        int testScore = 25;
        if (testScore >= 40) {
            System.out.println("You passed the exam!");
        } else {
            System.out.println("You failed the exam!");
        }
    }
}
```

Output:
You failed the exam!

11- WAP to demonstrate if else if

```
public class ExampleIfElseIf {
    public static void main(String[] args) {
        int testScore = 35;
        int graceMarks = 5;
        int passMarks = 40;
        if (testScore >= passMarks) {
            System.out.println("You passed the exam!");
        } else if ((testScore + graceMarks) >= passMarks) {
            System.out.println("You passed the exam with grace marks!");
        } else {
            System.out.println("You failed the exam!");
        }
    }
}
```

Output:
You passed the exam with grace marks!

Rule:

1. Curly braces are optional if there is only 1 statement after if or else
2. The statement should not be declarative statement

12- WAP to demonstrate if else if rules

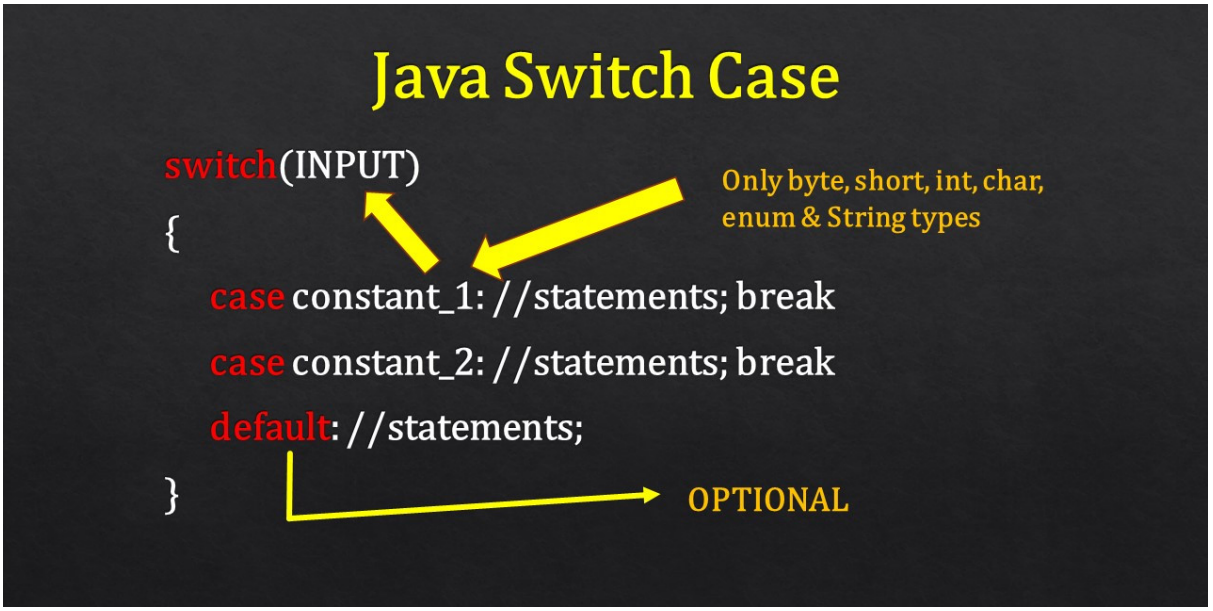
```
public class ExampleIfElseIfRules {
   /*Rule 1 : curly braces "{ }" are optional if only
               single statement after if or else
   Rule 2 : statement should not be declarative statement*/
   public static void main(String[] args) {
       int testScore = 35;
       int graceMarks = 5;
       int passMarks = 40;
       if (testScore >= passMarks)
           System.out.println("You passed the exam!");
       else if ((testScore + graceMarks) >= passMarks)
           System.out.println("You passed the exam with grace marks!");
       else
           System.out.println("You failed the exam!");
   }
}
```

12- WAP to demonstrate switch case

```
public class ExampleSwitch {
   public static void main(String[] args) {
       int monthNumber = 9;
       String month = "";
       switch (monthNumber) {
           case 1:
               month = "Jan";
               break;
           case 2:
               month = "Feb";
               break;
           case 3:
               month = "Mar";
               break;
           case 4:
               month = "Apr";
               break;
           case 5:
               month = "May";
               break;
           case 6:
               month = "Jun";
               break;
           case 7:
               month = "Jul";
               break;
```

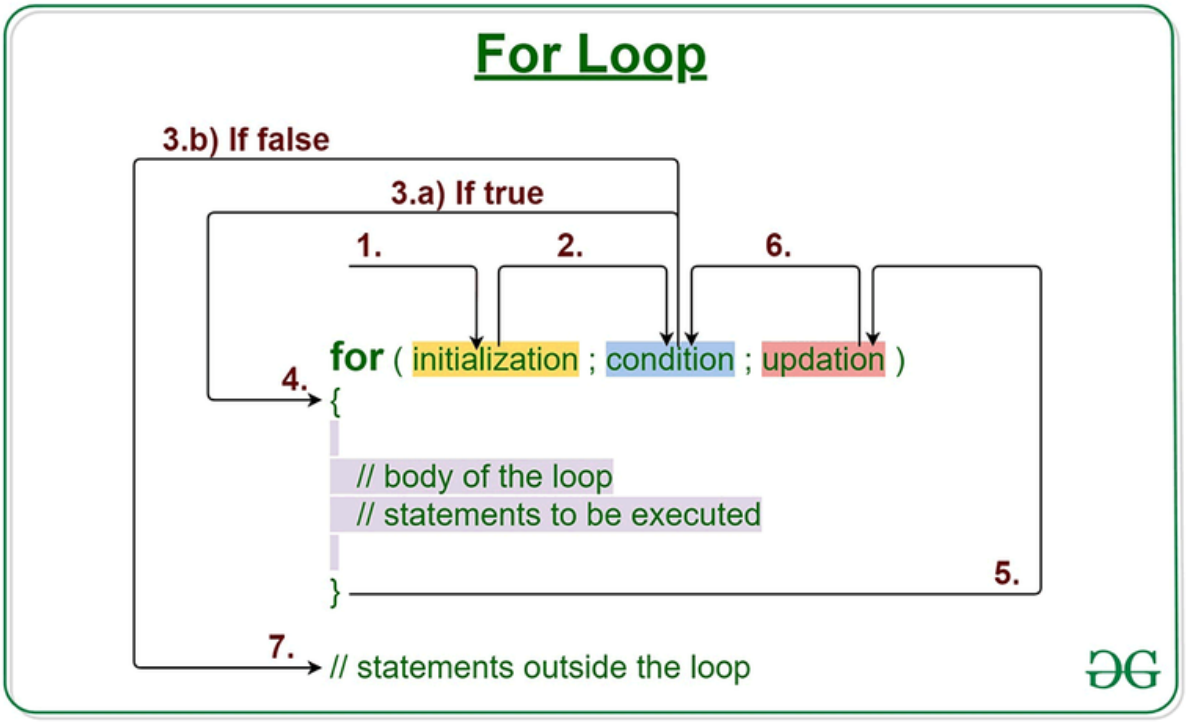
```
            case 8:
                month = "Aug";
                break;
            case 9:
                month = "Sep";
                break;
            case 10:
                month = "Oct";
                break;
            case 11:
                month = "Nov";
                break;
            case 12:
                month = "Dec";
                break;
            default:
                System.out.println("Invalid Input!");
                break;
        }
        System.out.println("Month :: "+month);
    }
}
```

Rules:



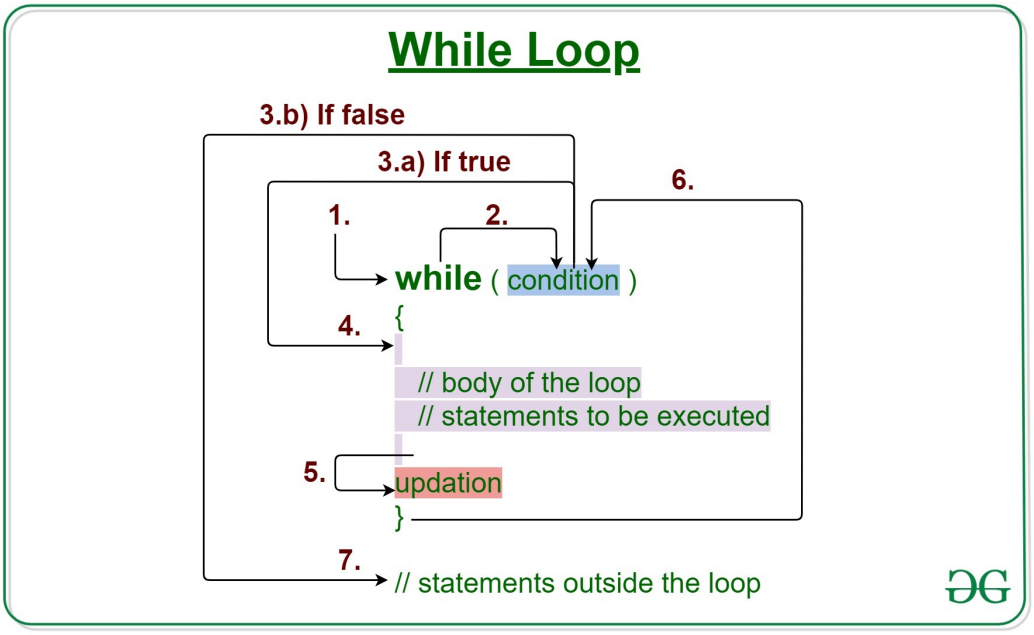
13- WAP to demonstrate for

```
public class ExampleFor {
    public static void main(String[] args) {
        int number = 10;
        int sum = 0;
        for (int i = 1; i <= number; i++) {
            sum += i;
        }
        System.out.println("sum :: " + sum);
    }
}
```



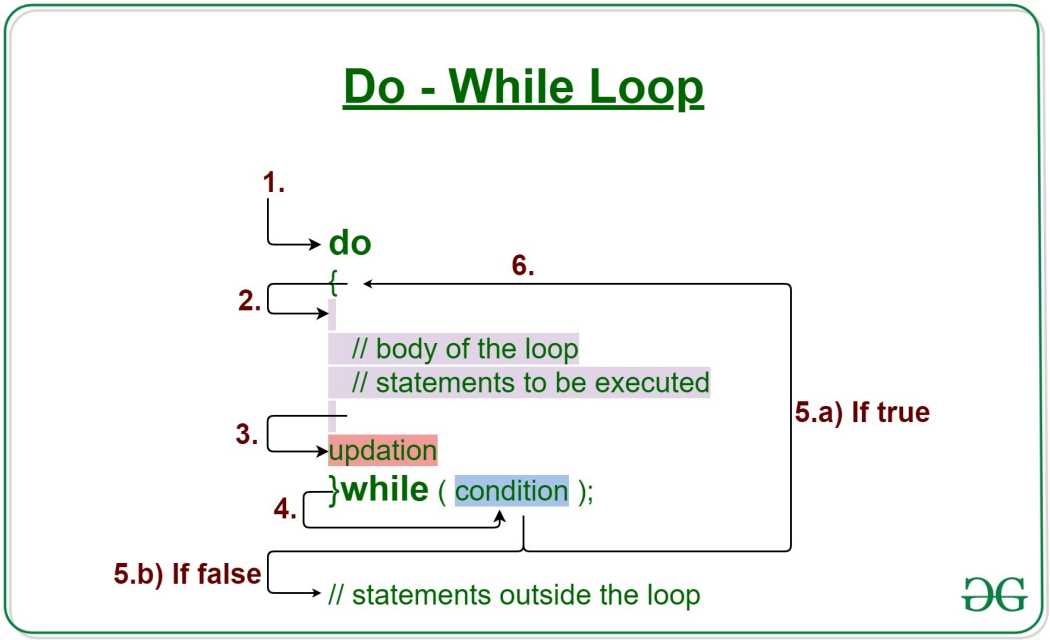
14- WAP to demonstrate while

```
public class ExampleWhile {
    public static void main(String[] args) {
        int number = 10;
        int sum = 0;
        while (number >= 1) {
            sum += number;
            number--;
        }
        System.out.println("sum :: " + sum);
    }
}
```



15- WAP to demonstrate do while

```
public class ExampleDoWhile {
   public static void main(String[] args) {
       int count = 10;
       do {
           System.out.println("I am learning java");
           count--;
       } while (count >= 1);
   }
}
```



16- WAP to demonstrate for each (enhanced for loop)

```
public class ExampleForEach {
    public static void main(String[] args) {
        int[] numbers = {1,2,3,4,5,6,7,8,9,10};
        int sum = 0;
        for(int num : numbers){
            sum+=num;
        }
        System.out.println("sum :: "+sum);
    }
}
```

17- WAP to demonstrate for each (enhanced for loop)

```
public class ExampleForEach2 {
    public static void main(String[] args) {
        List<String> fruitsList = List.of("Apple","Banana","Grape","Orange");
        for(String fruit : fruitsList){
            System.out.println("I am eating :: "+fruit);
        }
    }
}
```

Output:

I am eating :: Apple
I am eating :: Banana
I am eating :: Grape
I am eating :: Orange

Break Statement:

1. Used in switch to stop fall through
2. Used in loops to break upon some condition

17- WAP to demonstrate break

```
public class ExampleBreak {
    public static void main(String[] args) {
        String signalColor = "RED";
        switch (signalColor) {
            case "RED":
                System.out.println("Signal is RED.STOP!");
            case "YELLOW":
                System.out.println("Signal is YELLOW.BE READY!");
            case "GREEN":
                System.out.println("Signal is GREEN.GO!");
            default:
                System.out.println("ERROR. Invalid Colour!");
        }
    }
}
```

Output:
Signal is RED.STOP!
Signal is YELLOW.BE READY!
Signal is GREEN.GO!
ERROR. Invalid Colour!

```
public class ExampleBreak {
   public static void main(String[] args) {
        String signalColor = "YELLOW";
        switch (signalColor) {
            case "RED":
                System.out.println("Signal is RED.STOP!");
                break;
            case "YELLOW":
                System.out.println("Signal is YELLOW.BE READY!");
                break;
            case "GREEN":
                System.out.println("Signal is GREEN.GO!");
                break;
            default:
                System.out.println("ERROR. Invalid Colour!");
                break;
        }
   }
}
```

Output:
Signal is YELLOW.BE READY!

18- WAP to demonstrate break in for loop

```
public class ExampleBreakInLoop {
   public static void main(String[] args) {
        int number = 10;
        int sum = 0;
        for (int i = 1; i <= number; i++) {
            if(i==5){
                break;
            }
            sum += i;
        }
        System.out.println("sum :: " + sum);
   }
}
```

Output:
sum :: 10

19- WAP to demonstrate break in for loop

```
public class ExampleBreakInWhileLoop {
   public static void main(String[] args) {
        int number = 10;
        int sum = 0;
        while (number >= 1) {
            if(number==5){
                break;
            }
            sum += number;
            number--;
        }
        System.out.println("sum :: " + sum);
   }
}
```

Output:
sum :: 40

Continue Statement:
Used to skip current iteration and continue the next iteration

20- WAP to demonstrate continue in for loop

```
public class ExampleContinueFor {
   public static void main(String[] args) {
        int number = 10;
        int sum = 0;
        for (int i = 1; i <= number; i++) {
            if(i==5){
                continue;
            }
            sum += i;
        }
        System.out.println("sum :: " + sum);
   }
}
```

Output:
sum :: 50

21- WAP to demonstrate continue in while loop

```
public class ExampleContinueWhile {
    public static void main(String[] args) {
        int number = 10;
        int sum = 0;
        while (number >= 1) {
            if (number == 5) {
                number--;
                continue;
            }
            sum += number;
            number--;
        }
        System.out.println("sum :: " + sum);
    }
}
```

Output:
sum :: 50

22- WAP to demonstrate return statement

```
public class ExampleReturn {
    public static void main(String[] args) {
        System.out.println(performOperation("sum",20,30));
    }

    private static int performOperation(String operation, int num1, int num2) {
        if (operation.equals("sum")) {
            return num1 + num2;
        } else if (operation.equals("difference")) {
            return num1 - num2;
        } else {
            return num1 * num2;
        }
    }
}
```

Output:
sum :: 50

23- WAP to demonstrate try-catch-finally

```
public class ExampleTryCatchFinally {
   public static void main(String[] args) {
        int numerator = 100;
        int denominator = 2;
        System.out.println("result :: " + (numerator / denominator));
   }
}
```

Output:

result:: 50

```
public class ExampleTryCatchFinally {
   public static void main(String[] args) {
        int numerator = 100;
        int denominator = 0;
        System.out.println("result :: " + (numerator / denominator));
   }
}
```

Output:

Exception in thread "main" java.lang.ArithmeticException: / by zero
        at org.learjava.a2z.control_flow.ExampleTryCatchFinally.main(ExampleTryCatchFinally.java:7)

```
public class ExampleTryCatchFinally {
   public static void main(String[] args) {
        int numerator = 100;
        int denominator = 0;
        try {
            System.out.println("result :: " + (numerator / denominator));
        } catch (ArithmeticException e) {
            System.out.println("division by 0 is not possible");
        }
   }
}
```

Output:

division by 0 is not possible

```
public class ExampleTryCatchFinally {
   public static void main(String[] args) {
        int numerator = 100;
        int denominator = 0;
        try {
            System.out.println("result :: " + (numerator / denominator));
        } catch (ArithmeticException e) {
            System.out.println("division by 0 is not possible");
        } finally {
            System.out.println("I am executed if exception occurred or not");
        }
   }
}
```

Output:
division by 0 is not possible
I am executed if exception occurred or not