1. ***Simple IF***

**if (test expression)**

**{**

**statement/s to be executed if test expression is true;**

**}**

**Statements-x;**



class IfDemo

{

public static void main(String args[])

{

int i = 10;

if (i > 15)

System.out.println("10 is less than 15");

// This statement will be executed as if considers one statement by default

System.out.println("I am Not in if");

}

}

*2)if-else* statement

if (test expression)

{

true-block statements;

}

else

{

false-block statements;

}

statement-x;



// Java program to illustrate if-else statement

class IfElseDemo

{

public static void main(String args[])

{

int i = 10;

if (i < 15)

System.out.println("i is smaller than 15");

else

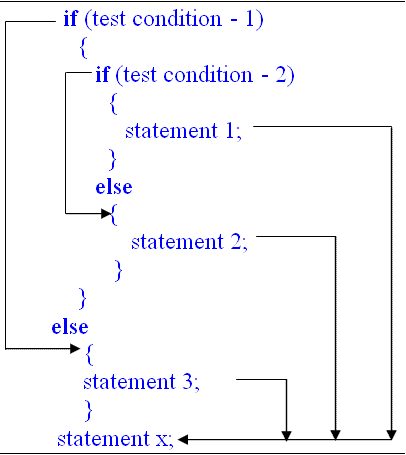
System.out.println("i is greater than 15");

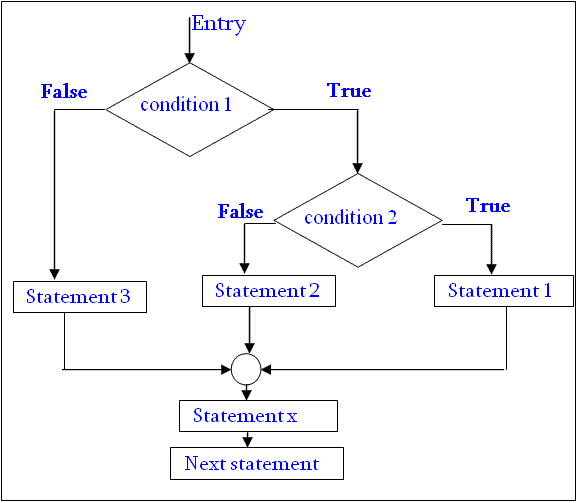
System.out.println(" Next statement");

}

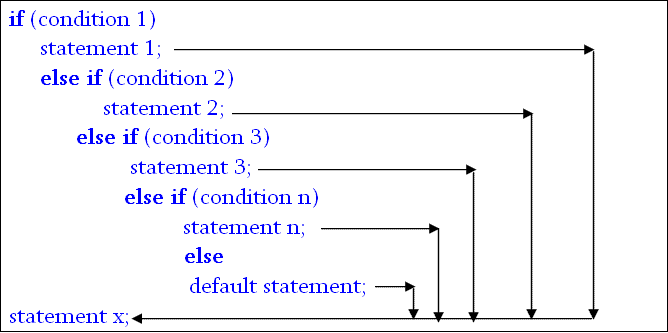
}

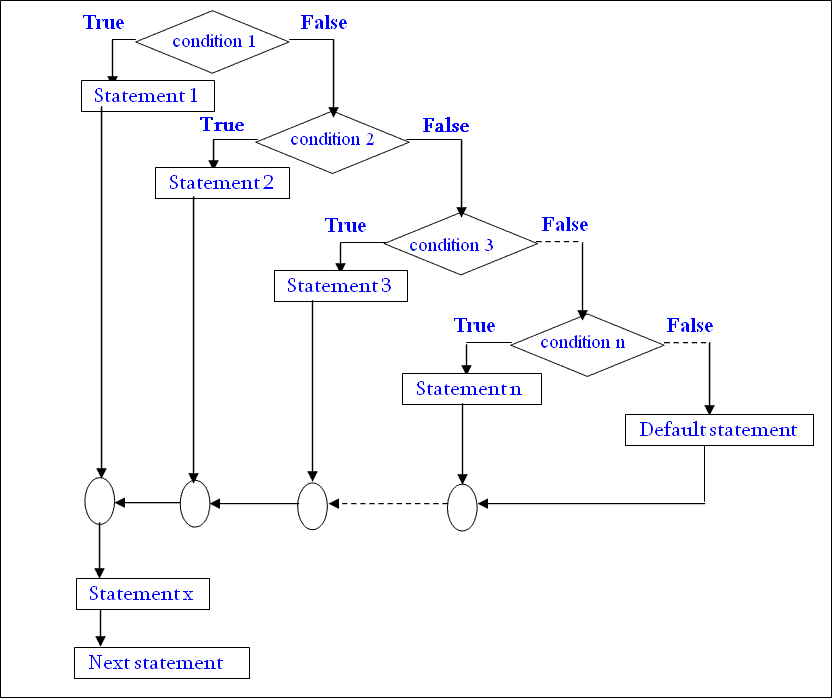
**3)Nested *if-else* statement**





4) Else…..If Ladder





// Java program to illustrate if-else-if ladder

class ifelseifDemo

{

public static void main(String args[])

{

int i = 20;

if (i == 10)

System.out.println("i is 10");

else if (i == 15)

System.out.println("i is 15");

else if (i == 20)

System.out.println("i is 20");

else

System.out.println("i is not present");

}

}

switch(**n**)

{

case value-1:

block-1;

break;

case value-2:

block-2;

break;

case value-3:

block-3;

break;

case value-4:

block-4;

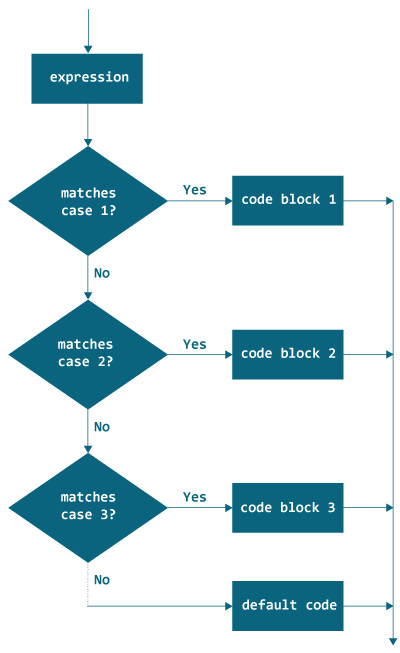
break;

default:

default-block;

break;

}



switch (*n*) {

case constant-1:

code/s to be executed if *n* equals to *constant1*;

break;

case *constant-2*:

code/s to be executed if *n* equals to *constant2*;

break;

case *constant -n*:

code/s to be executed if *n* equals to *constantn*;

break;

default:

code/s to be executed if *n* doesn't match to any cases;

}

// Java program to illustrate switch-case

class SwitchCaseDemo

{

public static void main(String args[])

{

int i = 9;

switch (i)

{

case 0:

System.out.println("i is zero.");

break;

case 1:

System.out.println("i is one.");

break;

case 2:

System.out.println("i is two.");

break;

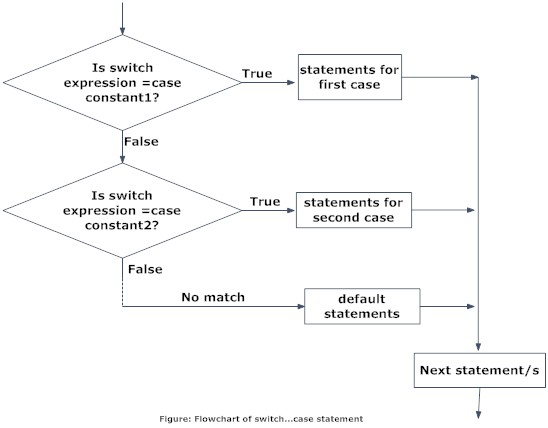
default:

System.out.println("i is greater than 2.");

}

}

}



while(condition)

{

statement(s);

}

Statement-x;



// Java program to illustrate while loop

class whileLoopDemo

{

public static void main(String args[])

{

int x = 1;

// Exit when x becomes greater than 4

while (x <= 4)

{

System.out.println("Value of x:" + x);

//increment the value of x for next iteration

x++;

}

}

}

for (initialization; condition;increment/decrement) {

statement(s);

}

// Java program to illustrate for loop

class forLoopDemo

{

public static void main(String args[])

{

// for loop begins when x=2

// and runs till x <=4

for (int x = 2; x <= 4; x++)

System.out.println("Value of x:" + x);

}

}



do {

statement-1;

} while( condition );



// Java program to illustrate do-while loop

class dowhileloopDemo

{

public static void main(String args[])

{

int x = 21;

do

{

//The line while be printer even

//if the condition is false

System.out.println("Value of x:" + x);

x++;

}

while (x < 20);

}

Nested For Loop:

for ( init; condition; increment )

{

for ( init; condition; increment )

{

statement(s);

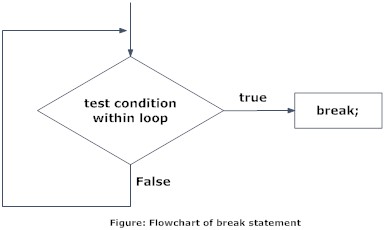
}

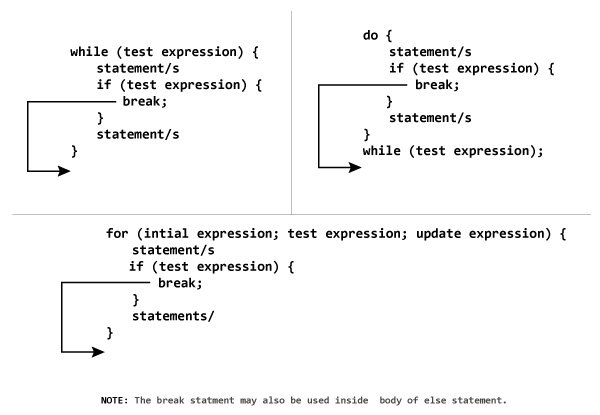
statement(s);

}

Syntax of break statement

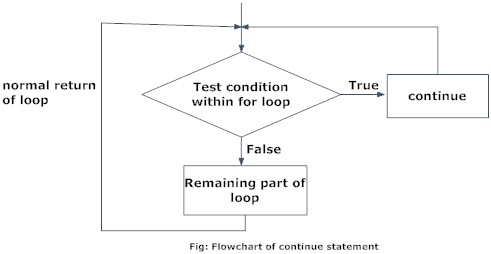
break;

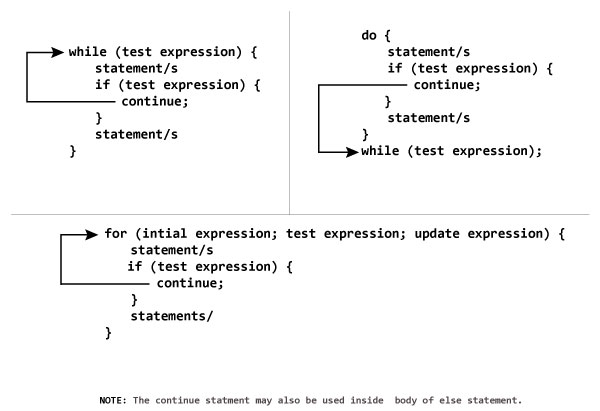




Syntax of continue Statement

continue;





**// Java program to illustrate using**

**// break to exit a loop**

**class BreakLoopDemo**

**{**

**public static void main(String args[])**

**{**

**// Initially loop is set to run from 0-9**

**for (int i = 0; i < 10; i++)**

**{**

**// terminate loop when i is 5.**

**if (i == 5)**

**break;**

**System.out.println("i: " + i);**

**}**

**System.out.println("Loop complete.");**

**}**

**}**

**// Java program to illustrate using**

**// continue in an if statement**

**class ContinueDemo**

**{**

**public static void main(String args[])**

**{**

**for (int i = 0; i < 10; i++)**

**{**

**// If the number is even**

**// skip and continue**

**if (i%2 == 0)**

**continue;**

**// If number is odd, print it**

**System.out.print(i + " ");**

**}**

**}**

**}**

Output:

1 3 5 7 9