For each loop

The for-each loop is an alternative to the for loop that is better suited to iterating over arrays and collections.

class HelloWorld {

public static void main(String[] args) {

int[] a = {10,20,30};

for(int x : a)

{

System.out.println(x);

}

}

}

OUTPUT

10

20

30

Switch Statement

Java Switch statement executes one statement from multiple conditions.

public class Main

{

static int sizeNumber = 34 ;

public static void main (String[]args)

{

switch (sizeNumber)

{

case 4 :

System.out.println(" its case One ");

case 2 :

System.out.println("It's case TWO ");

case 33 :

System.out.println(" its case 34 ");

case 35 :

System.out.println("It's case 35 ");

// break;

default:

System.out.println("Invalid size number");

}

}

}

OUTPUT

Invalid Size Number

In Switch statement,

1. duplicate cases not allowed.
2. The value of case must be of same data type as the variable in switch.
3. The break statement will use to terminate statement sequence.
4. The break statement is optional, if omitted, execution will continue on next statement.
5. Default is optional and can appear anywhere inside switch block.
6. If default statement is not at the end, then break statement must be kept after the default statement to omit execution of next statement.

Example : Finding weekday

public class Main

{

public static void main (String[]args)

{

int day = 7;

String weekday;

switch (5)

{

case 1:

weekday = "Sunday";

break;

case 2:weekday = "Monday";

break;

case 3:weekday = "Wednesday";

break;

case 4:weekday = "Thrusday";

break;

case 5:weekday = "Friday";

break;

case 6:weekday = "Saturday";

break;

case 7:weekday = "Sunday";

break;

default:weekday = "Invalid day";

}

System.out.println (weekday);

}

}

2. NESTED SWITCH STATEMENT

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

public class Main

{

public static void main (String[]args)

{

String Branch = "CSE";

int year = 2;

switch (year)

{

case 1:

System.out.println ("Year is 1");

break;

case 2:switch (Branch)

{

case "CSE":

System.out.println ("Branch is CSE");

break;

case "CCE":System.out.println ("Branch is CCE");

break;

default:System.out.println (" Nested switch working");

break;

}

}

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

3. SWITCH FOR – ENUM

public class Main

{

public enum Day

{ Sun, Mon, tue, Wed, Thrus, Fri, Sat };

public static void main (String[]args)

{

Day[]DayNow = Day.values ();

for (Day Now:DayNow)

{

switch (Now)

{

case Sun:

System.out.println ("It is Sunday");

break;

case Mon:System.out.println ("It is Monday");

break;

case tue:System.out.println ("It is Tuesday");

break;

case Wed:System.out.println ("It is Wednesday");

break;

case Thrus:System.out.println ("It is Thrusday");

break;

case Fri:System.out.println ("It is Friday");

break;

case Sat:System.out.println ("It is Saturday");

break;

}

} }

}

OUTPUT

It is Sunday

It is Monday

It is Tuesday

It is Wednesday

It is Thrusday

It is Friday

It is Saturday

Continue statement

Continue statement skips the current iteration of the project.

public class Main

{

public static void main(String[] args) {

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

{

if(i==2)

{

continue;

}

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

}

}

}

OUTPUT

i 0

i 1

i 3 skipped 2

i 4

i 5

i 6

i 7

i 8

i 9

|  |  |
| --- | --- |
| BREAK | CONITNUE |
| Break Statement is used to terminate loop  Immediately | Continue statement is used to skip current iteration in the loop |
| break keyword is used. | Continue keyword is used. |
| We can use break with switch statement | We cannot use continue in switch statement |
| It stops the execution of loops immediately | Continue does not stop execution of loop |
|  |  |
|  |  |