4.8 Conditions, Loops and Repetition

Control structure is the essential part of any programming language which is required to control the logic of your program. Commonly used control structures are -

- if statement
- while
- do-while
- for
- switch case

- break
- continue

We will discuss these control structures with the help of illustrative JavaScripts.

4.8.1 Selection Statement

As a selection statement we use various types of if statements. These statements are.

1. **if** The syntax of if is

```
if(condition)
  statement if the condition is true
```

2. **if...else** The syntax of if...else is

```
if(condition)
  statement if the condition is true
else
  statement if the condition is false
```

3. **if...else if** The syntax of if...else if is

```
if(condition)
  statement if the condition is true
else if(condition)
  statement if another condition is true
else if(condition)
  statement if another condition is true
...
else
  statement
```

Some times we can have **nested** if...statements, which work similar to C or C++. Here is a sample JavaScript

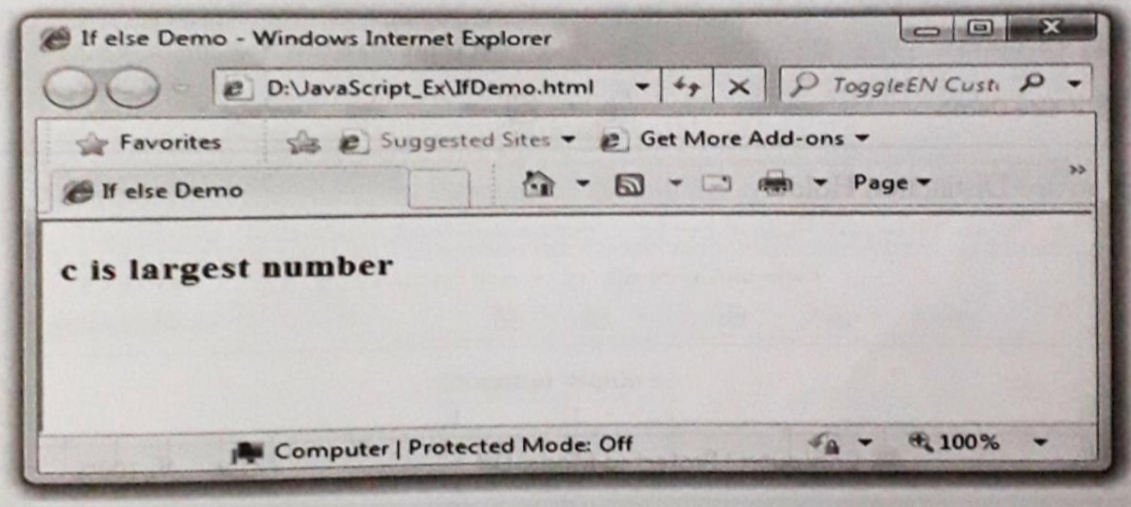
JavaScript[IfDemo.html]

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>If else Demo</title>
</head>
<body>
 <script type="text/javascript">
  var a,b,c;
  a=10;b=20;c=30;
  if(a>b)
  {
  if(a>c)
```

```
  document.write("<h3>a is largest number</h3>");
 else
  document.write("<h3>c is largest number</h3>");
 }
 else
 {
 if(b>c)
  document.write("<h3>b is largest number</h3>");
 else
  document.write("<h3>c is largest number</h3>");
 }
</script>
</body>
</html>
```

The above document is to find the largest number from the given three numbers. We have used nested if...else statements. The script is pretty simple, in which first of all we have compared *a* with *b*, if *a* is greater than *b* then we have compared *a* with *c*. Similarly in the *else* part *b* is compared with *c*. Thus all the three numbers get compared with each other and the largest number is found out among the three. Since already in our program we have set the a, b and c values as 10, 20 and 30 respectively the output should be c is largest number and here comes the output on web browser.

Output



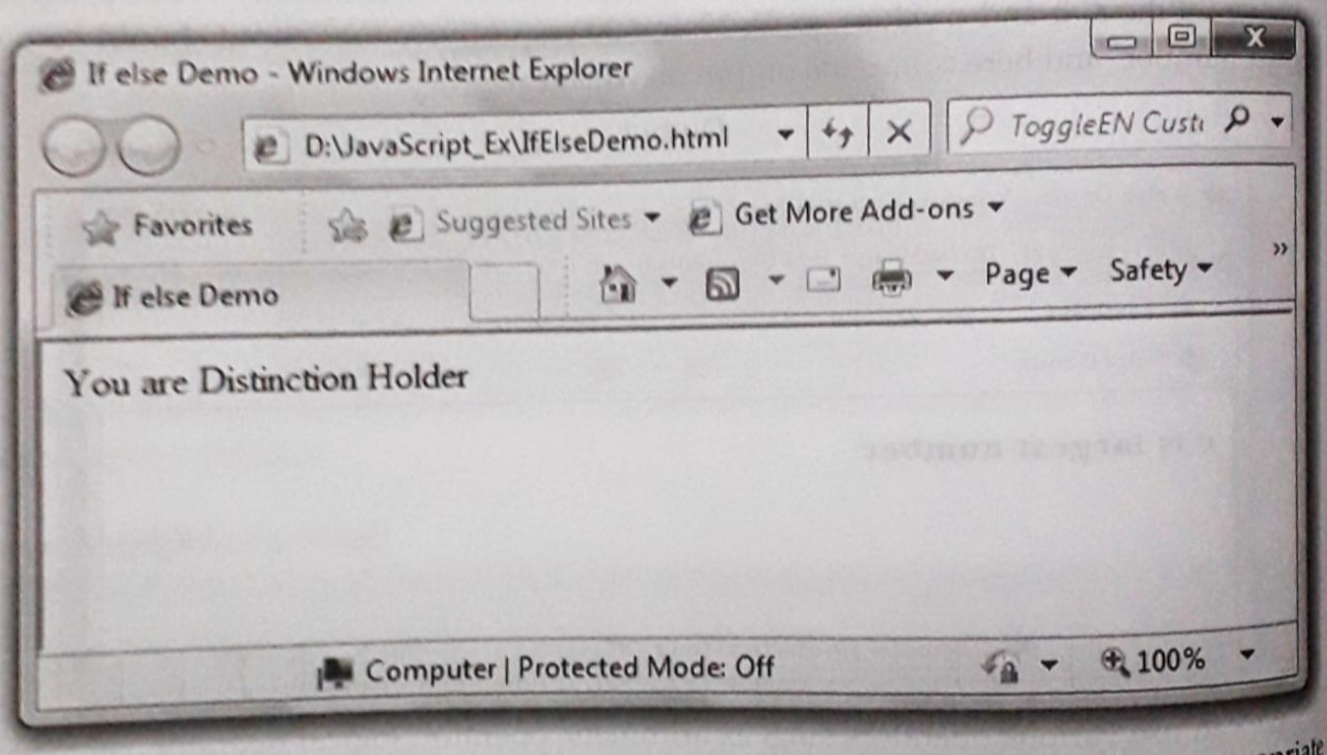
We can use various operators such as arithmetic operator, logical operator, relational operators and so on in the if statement. Following JavaScript makes use of such operators.

JavaScript[IfElseDemo.html]

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
 <title>If else Demo</title>
```

```
</head>
<body>
<script type="text/javascript">
  var marks;
  marks=80;
  if(marks<40)
    document.write("You are failed");
  else if(marks>=40 && marks<50)
    document.write("You are passed");
  else if(marks>=50 && marks<60)
    document.write("You have got Second class");
  else if(marks>=60 && marks<66)
    document.write("You have got First class");
  else
    document.write("You are Distinction Holder");
</script>
</body>
</html>
```

Output



In above script if we change the values of the marks variable then appropriate message will get displayed on the web browser.

4.8.2 while Statement

while statements help us in implementing the iterative logic of the program. The syntax of **while** is as follows -

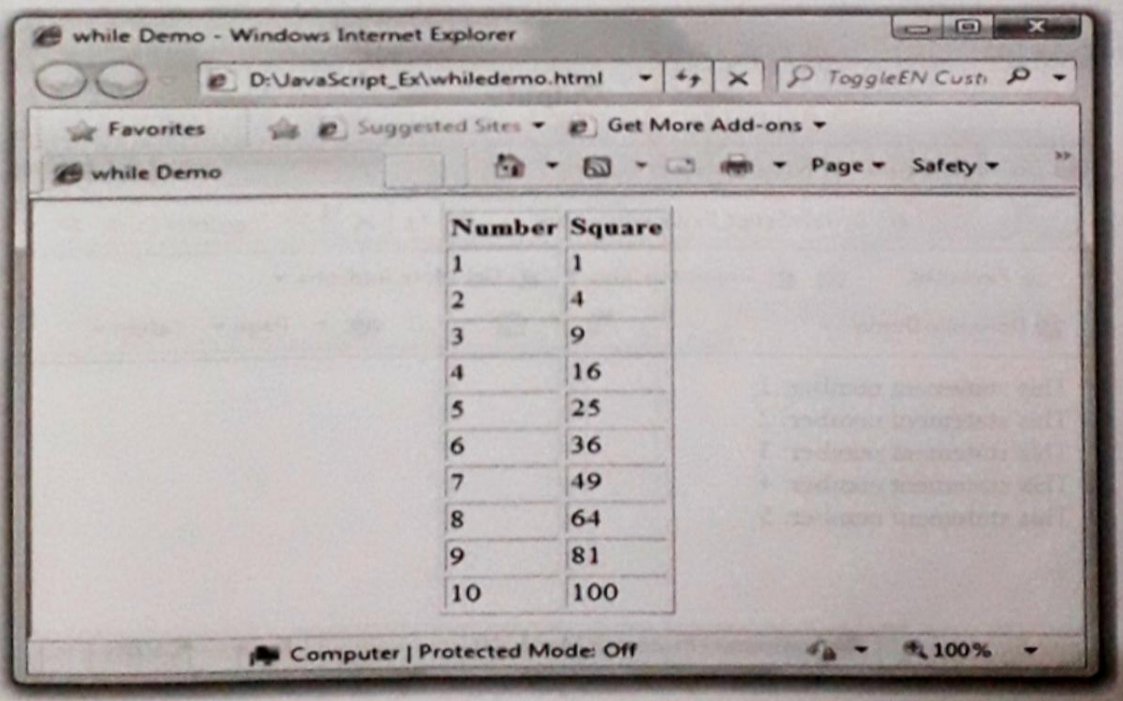
```
Some initial condition;
```

```
while(terminating condition)
{
 some statements;
 stepping condition;
}
```

The **while** is implemented by following JavaScript.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>while Demo</title>
</head>
<body>
 <table border=1 align="center">
  <th>Number</th><th>Square</th>
  <script type="text/javascript">
   i=1;
   while (i<=10)
   {
    document.write("<tr><td>"+i+"</td><td>"+(i*i)+"</td></tr>");
    i++;
   }
  </script>
</table>
</body>
</html>
```

Output



4.8.3 do...while

The do-while loop is similar to the while loop, the only difference is that the do-while executes at least once. The syntax of do...while is

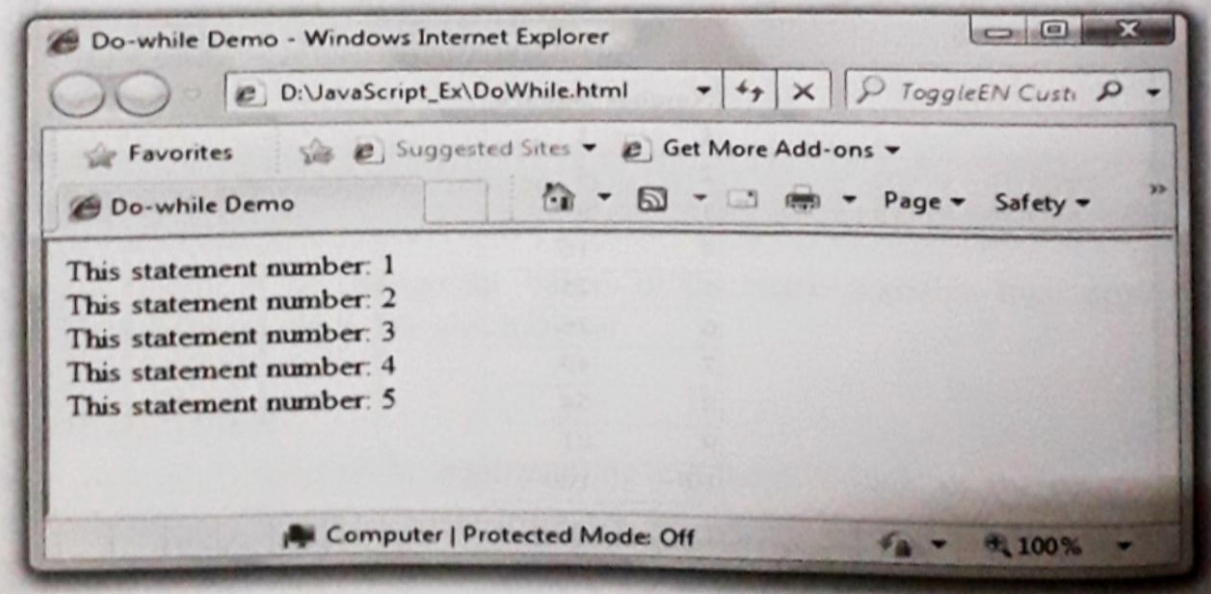
```
do
{
 ...
}while(condition);
```

The following JavaScript illustrates use of do...while

JavaScript[DoWhile.html]

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Do-while Demo</title>
</head>
<body>
 <script type="text/javascript">
  counter=1;
  do
  {
    document.write("This statement number: "+counter);
    document.write("<br>");
    counter++;
  }while(counter<=5);
 </script>
</body>
</html>
```

Output



4.8.4 for Loop

This is the most commonly used programming construct. The syntax of for loop is

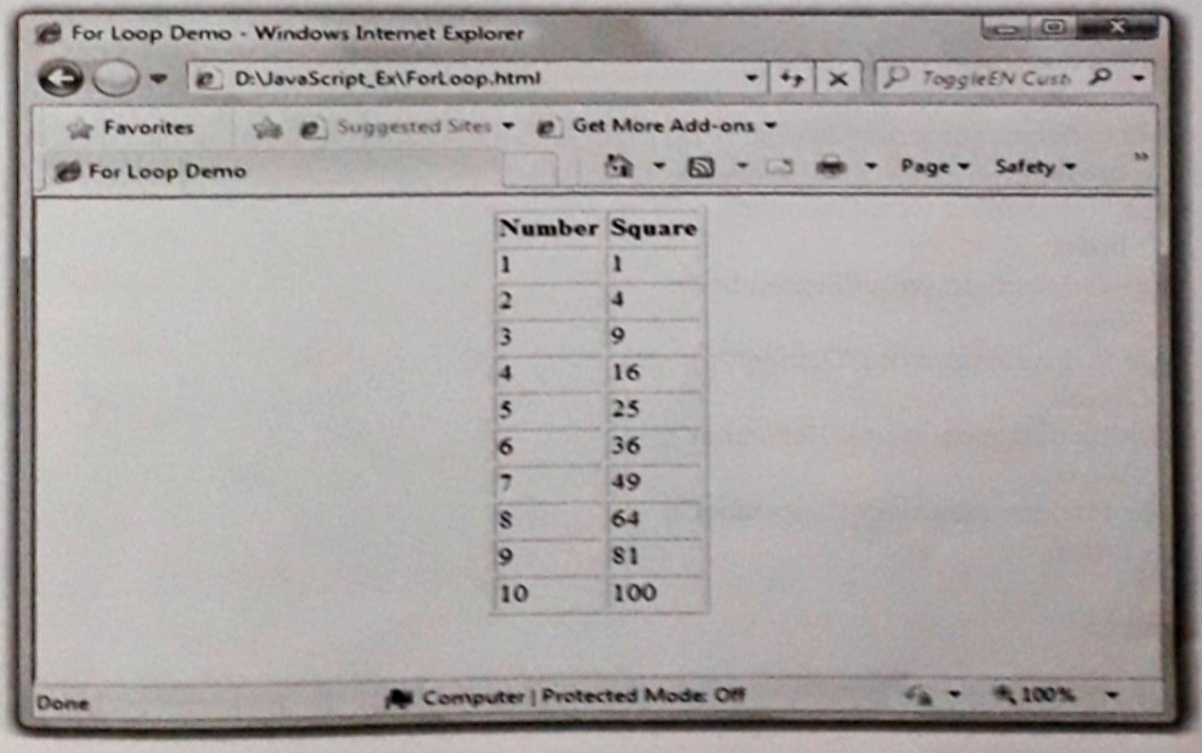
for(initial condition; terminating condition; stepping condition)

Here is a JavaScript which makes use of **for** loop

JavaScript[ForLop.html]

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>For Loop Demo</title>
</head>
<body>
 <table border=1 align="center">
  <th>Number</th><th>Square</th>
  <script type="text/javascript">
   for (i=1; i<=10; i++)
   {
    document.write("<tr><td>"+i+"</td><td>"+(i*i)+"</td></tr>");
   }
  </script>
</table>
</body>
</html>
```

Output



Number	Square
1	1
2	4
3	9
4	16
5	25
6	36
7	49
8	64
9	81
10	100

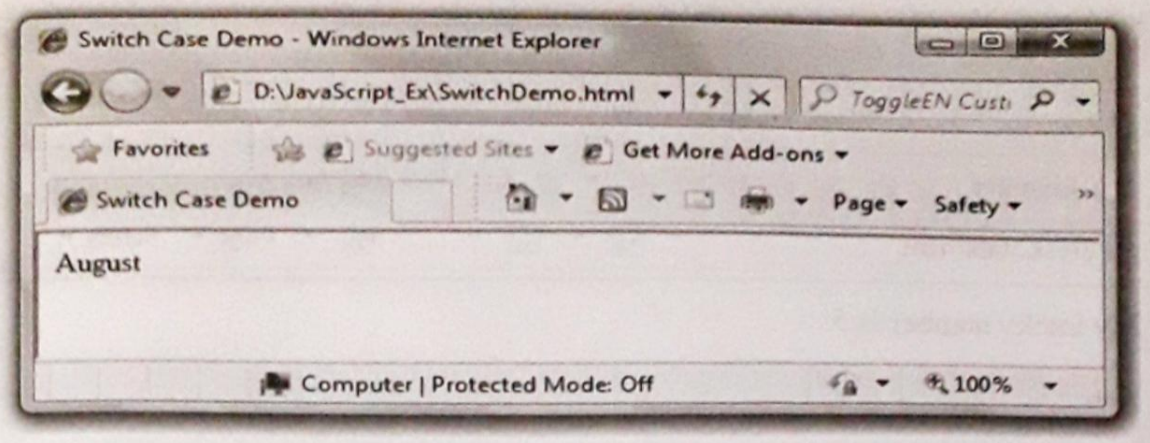
4.8.5 The switch Statement

switch case statement is basically to execute the desired choice. The syntax of this control structure is similar to that in C or C++.

JavaScript[SwitchDemo.html]

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Switch Case Demo</title>
</head>
<body>
 <script type="text/javascript">
  d=new Date();
  ch=d.getMonth();
  switch(ch)
  {
   case 0: document.write("January");
        break;
   case 1: document.write("February");
        break;
   case 2: document.write("March");
        break;
   case 3: document.write("April");
        break;
   case 4: document.write("May");
        break;
   case 5: document.write("June");
        break;
   case 6: document.write("July");
        break;
   case 7: document.write("August");
        break;
   case 8: document.write("September");
        break;
   case 9: document.write("October");
        break;
   case 10: document.write("November");
        break;
   case 11: document.write("December");
        break;
  }
 </script>
</body>
</html>
```

Output



In this script we have used the object **Date** which returns the value of the current date. Then using **getMonth()** method the month value can be obtained. The month value starts from 0 to 11 representing January to December. This value is basically obtained from system date function. The month is taken in variable *ch* according to the value in *ch* the corresponding case will get executed.

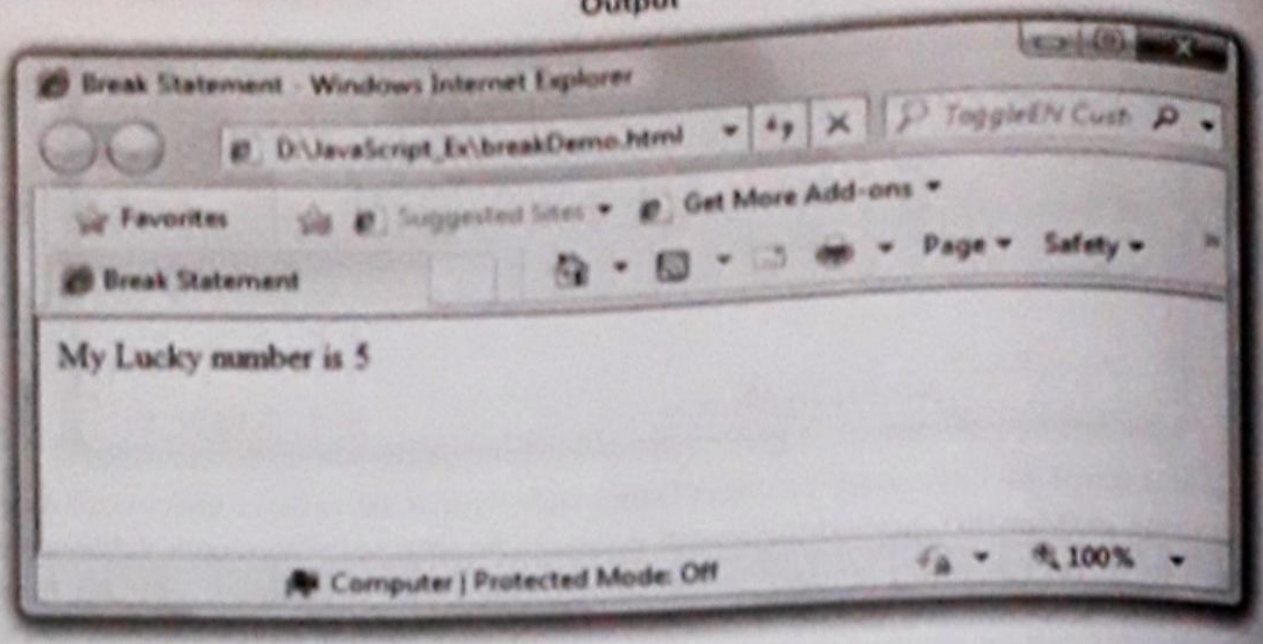
4.8.6 break Statement

Similar to C or C++, the break statement is used to break the loop. It is given be keyword **break**. The following script shows the use of break.

JavaScript[breakDemo.html]

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Break Statement</title>
</head>
<body>
   <script type="text/javascript">
    for(i=10;i>=0;i--)
    {
     if(i==5)
      break;
    }
    document.write("My Lucky number is "+i);
</script>
</body>
</html>
```

Output



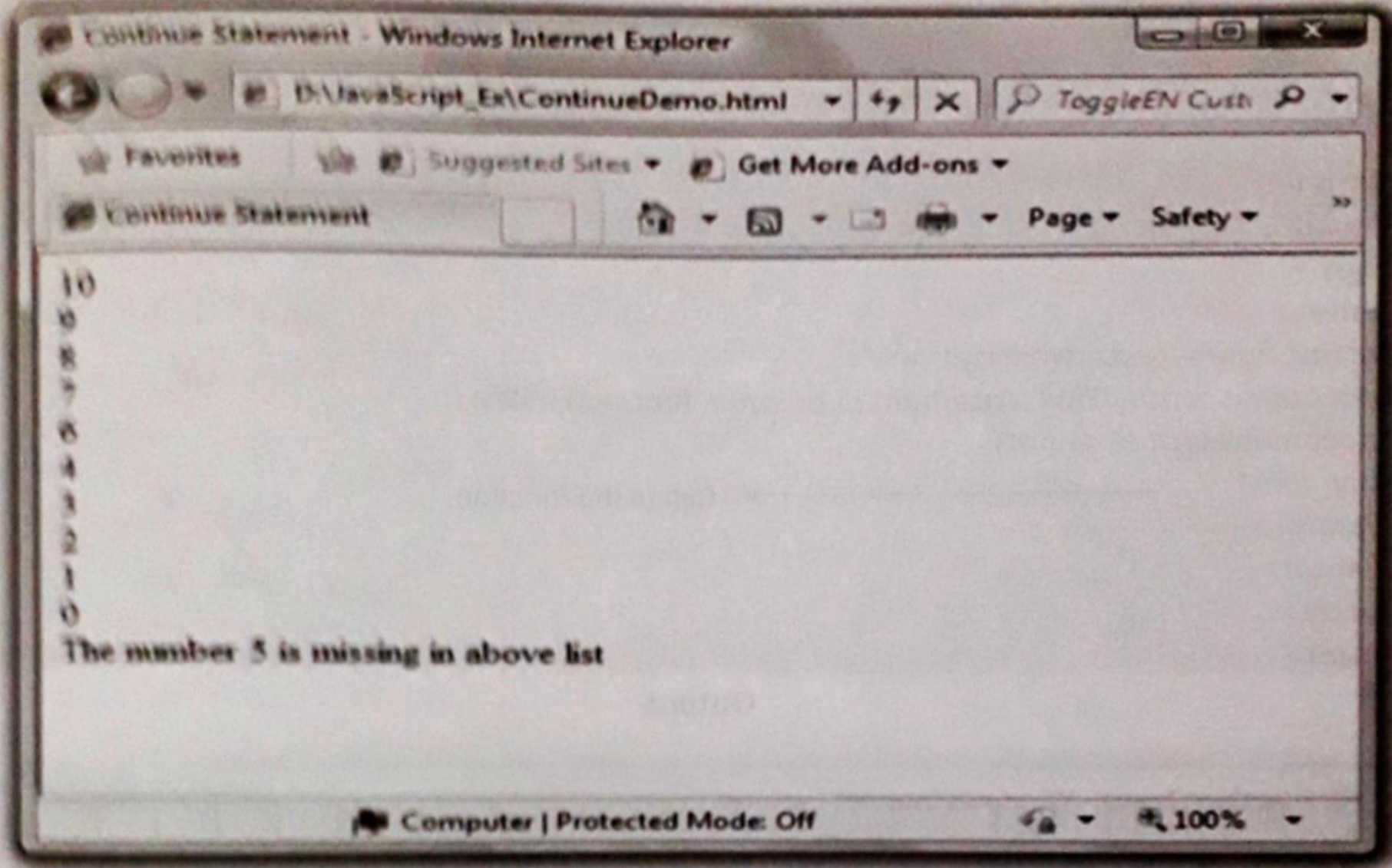
4.8.7 The continue Statement

The continue statement is used in a loop in order to continue(skip). The keyword **continue** is used to make use of continue statement in a loop.

Javascript Program[ContinueDemo.html]

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Continue Statement</title>
</head>
<body>
<script type="text/javascript">
 for(i=10;i>=0;i--)
 {
  if(i==5)
  {
   x=i;
   continue;
  }
  document.write(i);
  document.write("<br>");
 }
 document.write("The number " + x + " is missing in above list" );
</script>
</body>
</html>
```

Output


Continue Statement - Windows Internet Explorer
D:\JavaScript_Ex\ContinueDemo.html
ToggleEN Custo
Favorites
Suggested Sites
Get More Add-ons
Continue Statement
Page
Safety
10
9
8
7
6
4
3
2
1
0
The number 5 is missing in above list
Computer | Protected Mode: Off
100%