

JavaScript. Оператори

Лекція 2

JavaScript. Оператори

Лекція 2

- Оператор вибору
- Оператор циклу **while** (з передумовою)
- Оператор циклу **for** (з параметром)
- Оператор циклу **do while** (с післяумовою)
- Оператор передачі управління **continue**
- Оператор передачі управління **break**
- Оператор-перемикач **switch**
- Тернарний оператор **?:**



Умовний оператор

- *Вибір з однією альтернативою*

Формат оператора:

```
if (вираз) {  
    Оператор;  
}
```

- *Вибір з двома альтернативами*

Формат оператора:

```
if (вираз) {  
    Оператор1;  
}  
else {  
    Оператор2;  
}
```

Оператор циклу while (з передумовою)

Формат оператора:

```
while (умова) {  
    Оператор;  
}
```

Оператор циклу виконується якщо умова істинна. Якщо умова хибна, управління передається наступному за циклом оператору.

Приклад. Обчислення факторіала.

//варіант 1

```
<script>
```

```
    var n, f = 1, i = 1;
```

```
    n = prompt(" Введіть значення числа факторіал якого  
треба обчислити ", 1);
```

```
    n = parseFloat(n);
```

```
    while (i<=n) f*=i++; //{f=f*i; i=i+1;}
```

```
    document.write("факторіал числа" + n + " дорівнює  
"+f);
```

```
</script>
```

Оператор циклу while (з передумовою)

// варіант 2 зміна умови виходу і префіксна операція

```
<script>
```

```
    var n, f = 1, i = 1;
```

```
    n = prompt(" Введіть значення числа факторіал якого  
треба обчислити ", 1);
```

```
    n = parseFloat(n);
```

```
    while (i<n) f*=++i; //{i=i+1; f=f*i;}
```

```
    document.write(" факторіал числа"+ n +"дорівнює " +  
f);
```

```
</script>
```

Якщо оператор простий (одна команда) фігурні дужки можна не ставити

Оператор циклу for (з параметром)

Формат оператора:

```
for (вираз1; вираз2; вираз3) {  
    Оператор;  
}
```

Спочатку обчислюється вираз 1. Зазвичай вираз 1 ініціалізує змінну, яка використовується в циклі.

Потім обчислюється вираз 2. Якщо вираз 2 істинний, то виконується оператор в тілі циклу, обробляється вираз 3, і управління переходить до початку циклу, тобто до виразу 2.

Все повторюється до тих пір поки вираз 2 не стане хибним. Цикл закінчується і управління передається наступному оператору.

Вираз 1 і вираз 3 можуть складатися з декількох виразів, між якими ставиться кома.

Оператор циклу for (з параметром)

Приклад. Обчислення факторіала.

```
//вариант 1
```

```
<script>
```

```
    var n, f = 1;
```

```
    n = prompt(" Введіть значення числа факторіал якого  
треба обчислити ", 1);
```

```
    n = parseFloat(n);
```

```
    for (i=1; i<=n; i++) f*=i;
```

```
    document.write(" факторіал числа " + n + " дорівнює "  
+ f );
```

```
</script>
```

Оператор циклу for (з параметром)

```
// варіант 2 У тілі циклу порожній оператор
```

```
<script>
```

```
    var n, f = 1;
```

```
    n = prompt(" Введіть значення числа факторіал якого  
треба обчислити ", 1);
```

```
    n = parseFloat(n);
```

```
    for (i=1; i<=n; f*=i, i++) ;
```

```
    document.write(" факторіал числа " + n + " дорівнює "  
+ f );
```

```
</script>
```


Оператор циклу for (з параметром)

```
// варіант 3 вираз 1 відсутній, вираз 3 з одного,  
порожній оператор
```

```
<script>
```

```
    var n, f = 1, i = 1;
```

```
    n = prompt(" Введіть значення числа факторіал якого  
треба обчислити ", 1);
```

```
    n = parseFloat(n);
```

```
    for ( ; i<=n; f*=i++) ;
```

```
    document.write(" факторіал числа " + n + " дорівнює "  
+ f );
```

```
</script>
```

Оператор циклу for (з параметром)

```
//варіант 4
```

```
<script>
```

```
    var n, f = 1, i;
```

```
    n = prompt(" Введіть значення числа факторіал якого  
треба обчислити ", 1);
```

```
    n = parseFloat(n);
```

```
    for (i=n; i>0; f *= i--);
```

```
    document.write(" факторіал числа " + n + " дорівнює "  
+ f );
```

```
</script>
```

Оператор циклу `do while` (с післяумовою)

Формат оператора:

```
do {  
    Оператор;  
} while (умова);
```

Спочатку виконується оператор в тілі циклу. Потім перевіряється умова, і якщо воно істинне то управління передається назад до початку оператора тіла циклу і процес повторюється. Коли умова помилкова цикл закінчується і управління передається наступному оператору.

Особливості:

- Умова виходу відрізняється від циклу **REPEAT UNTIL**;
- Оператор тіла циклу виконується мінімум 1 раз;

Оператор циклу do while (с післяумовою)

Приклад. Обчислення факторіала.

```
//вариант 1
```

```
<script>
```

```
    var n, f = 1, i = 1;
```

```
    n = prompt(" Введіть значення числа факторіал якого  
треба обчислити ", 1);
```

```
    n = parseFloat(n);
```

```
    do f*=i++; while (i<=n);
```

```
    document.write(" факторіал числа " + n + " дорівнює "  
+ f );
```

```
</script>
```

Оператор передачі управління `continue`

Формат оператора:

`continue;`

Оператор направляє потік управління всередині циклу. Змушує припинитися поточну ітерацію циклу і починає наступну. Використовується тільки всередині операторів `while`, `for`, `do while`.

Приклад. Накопичення суми введених додатних чисел

Оператор передачі управління continue

```
<script>
  var x, sum = 0;
  var flag = true;
  while (flag) {
    x = prompt(" Введіть значення числа x-> ");
    x = parseFloat(x);
    if (x>0) {
      sum+=x;
      continue;
    }
    flag=false;
  }
  document.write("сума чисел " + sum);
</script>
```

Введіть значення числа x-> 5

Введіть значення числа x-> 2

Введіть значення числа x-> 6

Введіть значення числа x-> -8

сума чисел 13

Оператор передачі управління `break`

Формат оператора:

`break ;`

Оператор викликає вихід з циклу `while`, `for`, `do while`, також з оператора перемикача `switch`. Управління передається наступному оператору.

Типове використання - завершення циклу достроково.

Приклад. Накопичення суми введених додатних чисел

Оператор передачі управління break

```
<script>
  var x, sum = 0;
  while (true) {
    x = prompt(" Введіть значення числа x-> ");
    x = parseFloat(x);
    if (x<=0) {
      break;
    }
    sum += x;
  }

  document.write("сума чисел ")
</script>
```

Введіть значення числа x-> 5

Введіть значення числа x->
10

Введіть значення числа x->
15

Введіть значення числа x->
20

Введіть значення числа x->
-5

сумма чисел 50

Оператор-перемикач switch

```
switch (выражение) {  
    case константа1:  
        оператор1  
...  
    case константаN:  
        операторN  
    default:  
        оператор  
}
```

Мітка **case** - константний цілочисельний вираз. Всі мітки повинні бути унікальними.

Мітка **default** - необов'язкова.

Значення виразу - символ або ціле число, яке порівнюється з константами у всіх **case**. І передають управління рівному йому. Якщо рівного немає то передається оператору **default**. Зазвичай в кінці оператора ставлять **break**, що передає управління команді наступної за перемикачем **switch**. Якщо **break** відсутній то управління передається наступному **case**.

Оператор-перемикач switch

```
<script>
  var x, sum = 0;
  x = prompt("Скільки буде 2*2 = ");
  x = parseInt(x);
  switch (x) {
    case 3:
      document.write("Малувато");
      break;
    case 4:
      document.write("В точку!");
      break;
    case 5:
      document.write("Перебір");
      break;
    default:
      document.write("Я таких значень не знаю");
  }
</script>
```

Оператор-перемикач switch

```
<script>
  var x, sum = 0;
  x = prompt("Скільки буде 2*2 = ");
  x = parseInt(x);
  switch (x) {
    case 4:
      document.write("Вірно!");
      break;
    case 3:
    case 5:
      document.write("Невірно!");
      break;
    default:
      document.write(" Я таких значень не знаю ");
  }
</script>
```

Тернарний оператор ?:

Формат оператора:

Логічний вираз ? вираз1 : вираз2;

Компактна форма оператора **if else**. Алгоритм роботи операції наступний:

- Обчислюється логічний вираз.
- Якщо логічний вираз істинний, то обчислюється значення виразу 1, інакше - значення виразу 2.
- Обчислення значення повертається.

Приклад. Обчислення модуля і обчислення факторіала (використовується рекурсивний виклик функції)

Тернарний оператор ?:

```
<script>
    var m_i,i;
    i = prompt("Введіть значення числа модуль якого треба
обчислити ");
    i = parseFloat(i);
    m_i = (i<0) ? (-i) : (i);
    document.write("|" + i + "|" = " + m_i);
</script>
```

Що станеться під час виконання даного коду?

1.

```
switch (1)
{
    case 1:
    case 2:
    case 3: alert("Hello");
    break;
}
```
2.

```
var a = 2, b = 3;
var c = a++ >= b ? 0 : 1;
alert(c);
```
3.

```
var a = 2, b = 2;
var c = a == b++ ? a-- : b++;
alert(c);
```
4.

```
int a = 1, b = 2;
if (a < b)
    alert("a < b");
    alert("a не меньше b");
```
5.

```
var p = 1 > 2 ? (3 > 3 ? 1 : 2) : (3 < 3 ? 3 : 4);
```
6.

```
do
{
```

Що станеться під час виконання даного коду?

```
1.    var i = 10;
      do
        while (i++ < 15)
          i = i + 20;
      while (i < 2);
      alert(i);

2.    var a = 10;
      if (a++ > 10) {
        console.log("true");
      }
      {
        console.log("false");
      }
      console.log("ABC");
```

```
3.    var num = 10;
      switch (num) {
```

Дякую!