

Основы Программирования

Занятие №3
Условные выражения
HTML



**IT Education
Academy**

WWW.ITEA.UA

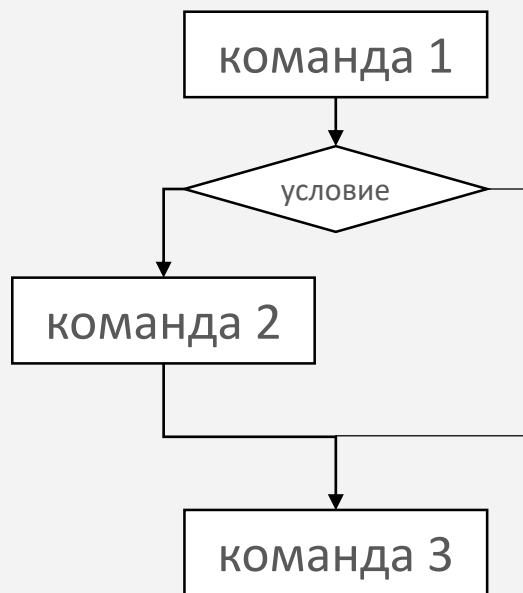
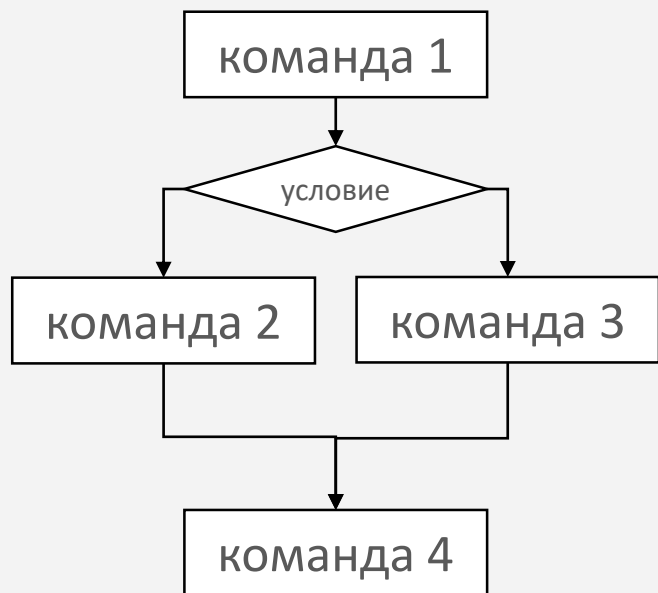
План занятия

- Синтаксис условных выражений
- Команда vs Блок команд
- HTML

Повторение материала

Условное выражение

Позволяет компьютеру принимать решения относительно выполнения следующих шагов.



Синтаксис

В JavaScript для описания условного выражения используется следующий синтаксис:

```
if (условие)
    команда1
else
    команда2
```

Если нужно, часть “else” можно не указывать. Тогда условное выражение имеет вид

```
if (условие)
    команда1
```

Обратите внимание на наличие скобок после слова if – они обязательны, этого требует спецификация языка.

Условное выражение

Пример. Создать папку на рабочем столе lesson3. Открыть notepad++ и написать:

```
var age = 16;
if (age < 18)
    console.log('Вам нет 18 лет. ');
else
    console.log('Вы уже достаточно взрослый');
```

И сохранить его как ex1.js в папку lesson3.

Открыть консольное окно и перейти в папку lesson3 с помощью команд:

```
cd Desktop
cd lesson3
```

Удостовериться, что в ней присутствует созданный файл:

```
dir (для Windows, а для Mac – ls)
```

и запустить его с помощью Node.js:

```
node ex1.js
```

* В дальнейшем подразумевается, что необходимо выполнять эти шаги для всех примеров, если не сказано иначе.

Команда vs Блок команд

ex2.js

```
var age = 16;
if (age < 18)
    console.log('Вам нет 18 лет.');//команда1
    console.log('Приходите через ' + (18 - age)); //команда2
```

```
console.log('Конец программы.') //команда 3
```

Поменять age с 16 на 21. И снова запустить.

Все честно, согласно спецификации языка:

```
if (условие)
    команда1
```

В данном примере интерпретатор рассматривает «команда2» и «команда3» как безусловные операции, которые нужно выполнять всегда. Для того, чтобы программа была семантически правильная (выполняла то что нам нужно), нам необходимо использовать синтаксис для объявления блока команд.

Блок команд

Для объявления блока команд используются фигурные скобки {...}, например:

ex2-correct.js

```
var age = 21;
if (age < 18) {
    console.log('Вам нет 18 лет.');//команда1
    console.log('Приходите через ' + (18 - age));//команда2
}
console.log('Конец программы.')
```

При чем, скобки можно указывать как на текущей строчке, так и на следующей, нет никакой разницы:

```
if (...) {
...
} //1
if (...)
{
...
} //2
```

Первый вариант принято использовать в JavaScript, Java, второй вариант в C, C++.

Условное выражение

Пример работы с условными выражениями.

В браузере.

ex3.html

```
//...<script>
var category = prompt('Введите категорию фильмов');
var movie;
if (category == 'ужасы') {
    var age = prompt('Сколько Вам лет?');
    if (age < 18)
        alert('Извините, Вам должно быть 18 лет для просмотра.');
```

else

```
    movie = prompt('Какой фильм бы Вы хотели посмотреть?');
}
else
    movie = prompt('Какой фильм бы Вы хотели посмотреть?');
if (movie) alert('Включаю ' + movie);
//</script>...
```

Преобразования

if (условие)
 команда

Поскольку JavaScript обладает нестрогим контролем типов, данные других типов превращаются в логический тип (Boolean).

Например 0 или undefined превращаются в false.
Подробнее [тут](#).

Несколько условий

Иногда существует необходимость проверки нескольких условий и выполнения соответствующих действий.

```
if (условие1) {  
    ...  
}  
else if (условие2) {  
    ...  
}  
else if (условие3) {  
    ...  
}  
else {  
    ...  
}
```

Условие

```
if (условие)
    команда
```

В условии может использоваться любое выражение, результатом которого является тип Boolean, т.е. **true** или **false**.

В простейшем случае -

```
if (true)
    console.log('Этот текст будет выводиться всегда');
```

```
if (false)
    console.log('...');
else
    console.log('Этот текст будет выводиться всегда');
```

Это могут быть операции проверки на равенство/неравенство (==, ===, !=, <, >, и тд), оператор instanceof (new Date() instanceof Object), оператор in (1 in [2, 3, 1, 5])

Практика

Задача. Перевести оценку студента из болонской системы в 5ти бальную систему:

100-95 - 5

94-75 - 4

74-60 - 3

59-0 - 2

Логические операторы

Помимо использования простых логических операций, существует возможность компоновать их с помощью логических операторов: && || !

Например:

ex4-1.js

```
var day = 12, month = 9, year = 2012;
if (month == 9) {
    if (year % 4 == 0) {
        if (day == 12) {
            console.log('С Днем разработчика!');
        }
    }
    else {
        if (day == 13) {
            console.log('С Днем разработчика');
        }
    }
}
```

ex4-2.js

```
var day = 12, month = 9, year = 2012;
if (month == 9) {
    if ( (year % 4 == 0 && day == 12) ||
        (day == 13) ) {
        console.log('С Днем разработчика!')
    }
}
```

Логические операторы

&& (Логическое И)	false	true
false	false	false
true	false	true

 (Логическое ИЛИ)	false	true
false	false	true
true	true	true

! (Логическое НЕТ)	false	true
	true	false

Условное выражение

Задача 1.

Написать программу, которая спрашивает два числа у пользователя и выводит на экран большее из них. (без использования Math).

Задача 2.

Написать программу которая спрашивает имя пользователя и пароль. И проверяет, если имя пользователя - "admin", то правильный пароль "root", если любое другое имя пользователя - пароль должен совпадать с именем. Программа должна проверять «подлинность» пользователя (аутентифицировать пользователя), если пользователь "admin" - выводить на экран "Вам доступны все права", иначе вывести на экран "Привет <имя пользователя>! Вы можете просматривать содержимое сайта."

Краткая форма

Код вида

```
var a;  
if (condition)  
    a = X  
else  
    a = Y
```

является популярной конструкцией, поэтому была придумана короткая форма:

```
var a = condition ? X : Y;
```

Называется тернарный оператор.

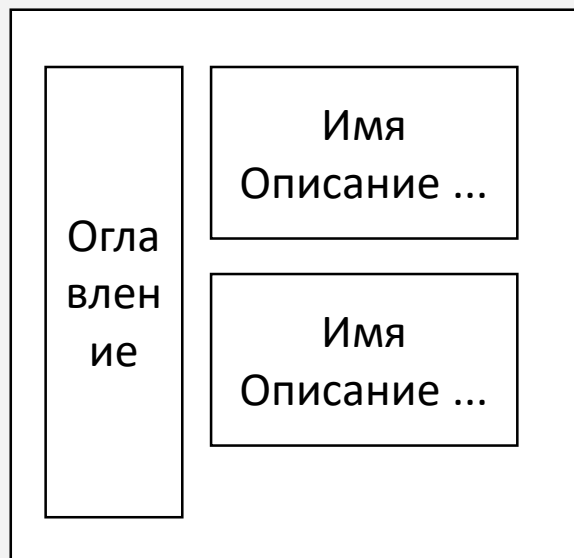
HTML

HTML - стандартизированный язык разметки WEB-страниц. Он описывает структуру страницы (ее содержимое).

С помощью HTML указывается взаимосвязь одних элементов страницы с другими.

Например, страница новостей состоит из разных новостных заголовков, а также оглавления.

Новостной заголовок состоит из имени заголовка и краткого описания.



HTML Элементы

HTML страничка состоит из элементов. Большинство элементов состоит из открывающегося тега, содержимого и закрывающегося тега.

Например,

```
<p>This is paragraph</p>
```

Содержимым элементов могут выступать другие элементы, например:

```
<div>
```

```
  <h1>This is heading</h1>
```

```
  <p>This is paragraph</p>
```

```
</div>
```

Каждый элемент имеет свое значение, например, элемент div используется как «контейнер для других элементов, а элемент p для описания параграфа в тексте. [Описание других элементов](#).

Также элементы обладают атрибутами, с помощью которых можно указывать дополнительную информацию об элементе, например:

```
<p align="left">This is left aligned</p>
```

```
<p align="center">This is centered</p>
```

```
<p align="right">This is right aligned</p>
```

или

```
<p id="main">Main text</p>
```

HTML Страница

Структура HTML страницы:

```
<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

Все теги, которые находятся в элементе head - «мета»-информация, т.е. она не отображается на странице, а всего лишь предоставляет дополнительную информацию, например на каком языке (русский, английский) написан текст страницы, или какие дополнительные файлы нужно открыть браузеру чтобы отобразить страницу и т.д.

Все теги, которые находятся в элементе body - содержимое страницы.

Группа элементов h

Позволяют описывать заголовки на страничке. Существует 6 вариантов от h1 до h6, где h1 - заголовок самого высокого уровня.

`<h1>Раздел 1</h1>`

`<h2>Пункт 1</h2>`

`<h3>Подпункт 1</h3>`

...

Некоторые другие элементы

Элемент `p` - описывает параграф на странице.

Элемент `ol` в сочетании с элементом `li` - список:

```
<ol>
  <li>Один</li>
  <li>Два</li>
</ol>
```

Элемент `div` - группирование нескольких элементов. Позволяет в дальнейшем удобнее работать с группой.

Элемент `script` - встраивание JavaScript кода.

Элемент `br` - один из немногих «пустых» элементов. Он состоит только из одного открывающегося тега и не содержит тела и закрывающегося тега: `
`. Означает перенос на другую строку текста.

Аттрибут id

Аттрибут id имеет важное значение, когда речь идет о взаимодействии JavaScript с HTML. С его помощью удобно манипулировать структурой странички в коде:

ex6.html

```
//...<body>
<div id='this'></div>
<script>
  var input = prompt('Enter Text to embed it.')
  document.getElementById('this').innerHTML = input
</script>
//</body>...
```

HTML + JavaScript

Создайте страничку, которая содержит два элемента `<div id='0'>Choose me</div>` и `<div id='1'>No, choose me</div>`. Напишите JavaScript код, который просит пользователя ввести число и если число четное - заменяет текст первого элемента на 'Yes, you've chosen me!', а второго на 'No, I lost' и наоборот, если число нечетное.

Подсказки:

число четное - если без остатка делится на 2.

чтобы менять содержимое элемента, его нужно сначала выбрать с помощью `document.getElementById(...)`

чтобы поменять содержимое, необходимо свойству элемента `innerHTML` присвоить значение, например если есть элемент `first`, то `first.innerHTML = "new content"`.

Summary

1. Условные выражения нужны компьютеру для принятия решений.
2. Синтаксис - `if (condition) command1 else command2`
3. Когда нужно выполнить несколько команд по условию - используется блок
4. Условия можно группировать с помощью логических операторов `&& || !`
5. Тернарный оператор - короткая форма записи.
6. HTML - язык разметки. Состоит из элементов.
7. Элемент = открывающийся тег + содержимое + закрывающийся тег.
8. В открывающемся теги можно записывать атрибуты.

Домашнее задание

1. Напишите программу нахождения наибольшего из трех чисел a, b, c числа.
2. Придумайте алгоритм о принятии решения о найме человека на работу программистом. Используйте не меньше 3х-4х критериев, рассмотрите разные возможные варианты.
3. Перепишите пример ex4-2.js с использованием тернарного оператора. Подсказка: по умолчанию - День программиста это 13 день сентября, но если это високосный год - эта дата смещается на -1, т.е. на 12 сентября.
 1. Используйте [правильное](#) формирование високосного года - это год который делится на 4 и не делится на 100, либо делится на 400.

Спасибо за внимание!