

# СПИСКИ

ПОДГОТОВИЛА: ДАККА В.С.

# ПЛАН

- ▶ Создание меню при помощи структуры списков (`<ul>`, `<li>`).
- ▶ Создание вложенных списков.
- ▶ Форматирование списков при помощи CSS.
- ▶ Списки определений: элементы `<dl>`, `<dd>`, `<dt>`.

# Создание меню при помощи структуры списков (<ul>, <ol>, <li>),

## МАРКИРОВАННЫЙ

```
1 <ul>
2   <li>Microsoft</li>
3   <li>Google</li>
4   <li>Apple</li>
5   <li>IBM</li>
6 </ul>
```

- Microsoft
- Google
- Apple
- IBM

## НУМЕРОВАННЫЙ

```
1 <ol>
2   <li>Microsoft</li>
3   <li>Google</li>
4   <li>Apple</li>
5   <li>IBM</li>
6 </ol>
```

1. Microsoft
2. Google
3. Apple
4. IBM

Атрибут	Описание, принимаемое значение
<code>reversed</code>	Атрибут <code>reversed</code> задает отображение списка в обратном порядке (например, 9, 8, 7...).
<code>start</code>	Атрибут <code>start</code> задает начальное значение, от которого пойдет отсчет нумерации, например, конструкция <code>&lt;ol start="10"&gt;</code> первому пункту присвоит порядковый номер «10». Также можно одновременно задавать тип нумерации, например, <code>&lt;ol type="I" start="10"&gt;</code> .
<code>type</code>	Атрибут <code>type</code> задает вид маркера для использования в списке (в виде букв или цифр). Принимаемые значения: <ul style="list-style-type: none"><li><code>1</code> — значение по умолчанию, десятичная нумерация.</li><li><code>A</code> — нумерация списка в алфавитном порядке, заглавные буквы (A, B, C, D).</li><li><code>a</code> — нумерация списка в алфавитном порядке, строчные буквы (a, b, c, d).</li><li><code>I</code> — нумерация римскими заглавными цифрами (I, II, III, IV).</li><li><code>i</code> — нумерация римскими строчными цифрами (i, ii, iii, iv).</li></ul>

## СВОЙСТВА ДЛЯ НУМЕРОВАННОГО СПИСКА

# Список определений <dl>, <dt>, <dd>

```
1 <dl>
2   <dt>Режиссер:</dt>
3     <dd>Петр Точилин</dd>
4   <dt>В ролях:</dt>
5     <dd>Андрей Гайдулян</dd>
6     <dd>Алексей Гаврилов</dd>
7     <dd>Виталий Гогунский</dd>
8     <dd>Мария Кожевникова</dd>
9 </dl>
```

Режиссер:

Петр Точилин

В ролях:

Андрей Гайдулян

Алексей Гаврилов

Виталий Гогунский

Мария Кожевникова

# Вложенный СПИСОК

Свойства  
`<ul type='disc|circle|square'>`:

```
1 <ul>
2   <li>Пункт 1.</li>
3     <li>Пункт 2.
4       <ul>
5         <li>Подпункт 2.1.</li>
6         <li>Подпункт 2.2.
7           <ul>
8             <li>Подпункт 2.2.1.</li>
9             <li>Подпункт 2.2.2.</li>
10          </ul>
11        </li>
12        <li>Подпункт 2.3.</li>
13      </ul>
14    </li>
15    <li>Пункт 3.</li>
16  </ul>
```

- Пункт 1.
- Пункт 2.
  - Подпункт 2.1.
  - Подпункт 2.2.
    - Подпункт 2.2.1.
    - Подпункт 2.2.2.
  - Подпункт 2.3.
- Пункт 3.

# Многоуровневый нумерованный СПИСОК

```
<ol>
```

```
<li>пункт</li> <!--1.-->
```

```
<li>пункт
```

```
<ol>
```

```
<li>пункт</li> <!--2.1.-->
```

```
<li>пункт</li> <!--2.2.-->
```

```
<li>пункт
```

```
<ol>
```

```
<li>пункт</li> <!--2.3.1.-->
```

```
<li>пункт</li> <!--2.3.2.-->
```

```
<li>пункт</li> <!--2.3.3.-->
```

```
</ol>
```

```
</li> <!--2.3.-->
```

```
<li>пункт</li> <!--2.4.-->
```

```
</ol>
```

```
</li> <!--2.-->
```

```
<li>пункт</li> <!--3.-->
```

```
<li>пункт</li> <!--4.-->
```

```
</ol>
```

## Стили CSS для многоуровневого списка

- ▶ `ol { counter-reset: li; /*Идентифицируем счетчик и даем ему имя (знач. -0)*/`  
`list-style: none; /*убираем стандартную нумерацию*/}`
- ▶ `li:before {`  
`counter-increment: li; /* Псевдоэлемент before указывает, что`  
`содержимое будет располагаться до пунктов списка.`  
`content: counters(li, ".") ". "; /*content выводится номер пункта.`  
`counters() = значению всех счетчиков с таким именем. Точка в`  
`кавычках добавляет разделяющую точку между цифрами, а точка с`  
`пробелом добавляется перед содержимым каждого пункта`

# ПРИМЕР:

1. пункт

2. пункт

2.1. пункт

2.2. пункт

2.3. пункт

2.3.1. пункт

2.3.2. пункт

2.3.3. пункт

2.4. пункт

3. пункт

4. пункт

# СТИЛИ СПИСКА

## ▶ list-style-type

```
1 ul {  
2 list-style-type: square;  
3 }  
4 ol {  
5 list-style-type: lower-alpha;  
6 }
```

- |               |                |
|---------------|----------------|
| ■ WordPress   | a. WordPress   |
| ■ Joomla      | b. Joomla      |
| ■ ModX        | c. ModX        |
| ■ TextPattern | d. TextPattern |
| ■ Drupal      | e. Drupal      |

HOME 

ARCHIVES

WORK 

ABOUT

CONTACT

CSS

## list-style-image

Значения:

<code>url(url)</code>	Путь к изображению.
<code>none</code>	Изображение не будет отображаться.

```
1 ul {  
2 list-style-image: url("images/romb.png");  
3 }
```

- ☐ WordPress
- ☐ Joomla
- ☐ ModX
- ☐ TextPattern
- ☐ Drupal

# list-style-image:url()

## list-style-position

Значения:

outside

Значение по умолчанию. Маркер располагается вне блока с текстом.

inside

Маркер списка изображается в одном блоке с текстом.  
Последующие строки текста будут располагаться под значком маркера, т.е. маркер будет обтекаться текстом.

```
1 ul {  
2 list-style-position: inside;  
3 }  
4 ol {  
5 list-style-position: outside;  
6 }
```

- WordPress
- Joomla
- ModX
- TextPattern
- Drupal

- a. WordPress
- b. Joomla
- c. ModX
- d. TextPattern
- e. Drupal

# Практика

## Пример нумерованного списка

Для всего списка тип нумерации изменен на "a" - латинская нумерация в верхнем

- a. Элемент списка 1
- b. Элемент списка 2
- c. Элемент списка 3
- d. Элемент списка 4
- e. Элемент списка 5

## Пример многоуровневого упорядоченного списка

Нумерация каждого уровня упорядоченных списков задается с помощью стилей.

- **Зима**
  12. Декабрь
  1. Январь
  2. Февраль
- **Весна**
  3. Март
  4. Апрель
  5. Май
- **Лето**
  6. Июнь
  7. Июль
  8. Август
- **Осень**
  9. Сентябрь
  10. Октябрь
  11. Ноябрь

---

### В этом выпуске:

- Русская кухня. Уха бурлацкая
- Украинская кухня. Вареники
- Молдавская кухня. Паприкаш
- Кавказская кухня. Суп-харчо
- Прибалтийская кухня. Вертиняй

# Практика

1. пункт 1



a1



a2



a3

2. пункт 2

3. пункт 3

## Список определений

SGML

Standard Generalized Markup Language - стандартизованный обобщенный язык

HTML

HyperText Markup Language - язык разметки гипертекста. В последнее время  
- Internet.

CSS

Cascading Style Sheets - каскадные таблицы стилей. Используются для визу

XML

eXtensible Markup Language - расширяемый язык разметки. С его помощью

# ДОМАШНЯЯ РАБОТА

## Плюс из файла

### Пример списка с графическими маркерами

- First
- Second
- Third
- Forth
- Fifth

- i. First
- ii. Second
- iii. Third
- iv. Forth
- v. Fifth

### Пример многоуровневого списка

- A. Cascading Style Sheets
- B. Cascading Style Sheets
  - 1. Cascading Style Sheets
  - 2. Cascading Style Sheets
  - 3. Cascading Style Sheets
- C. Cascading Style Sheets
  - 1. Cascading Style Sheets
    - Cascading Style Sheets
    - Cascading Style Sheets
    - Cascading Style Sheets
  - 2. Cascading Style Sheets