

ЛАБОРАТОРНАЯ РАБОТА №2

Тема: *Организация диалога пользователя при использовании графического пользовательского интерфейса типа "Hand User Interface".*

Цель: *Изучение и приобретение навыков разработки пользовательского интерфейса GUI с использованием гиперссылок в формате HTML*

ТЕОРИЯ

GUI (англ. Graphical user interface) – графический пользовательский интерфейс (ГИП) – разновидность пользовательского интерфейса, в котором элементы интерфейса (меню, кнопки, значки, списки и т. п.), представленные пользователю на дисплее, исполнены в виде графических изображений.

Впервые концепция ГИП была предложена учеными из исследовательской лаборатории Xerox PARC в 1970-х. В 1973 году в лаборатории Xerox PARC собрали молодых учёных и дали свободу исследований. В результате, кроме всего прочего, на свет появляется концепция графического интерфейса WIMP (Windows, Icons, Menus, Point-n-Click). В рамках этой концепции создаётся компьютер Alto.

В 1979 году Three Rivers Computer Corporation выпускает рабочую станцию PERQ, похожую по принципам построения на Alto. В 1981 году Xerox выпускает продолжение Alto – Star.

Коммерческое воплощение концепция ГИП получила в продуктах корпорации Apple Computer. В операционной системе AmigaOS ГИП с многозадачностью был использован в 1985 году. В настоящее время ГИП является стандартной составляющей большинства доступных на рынке операционных систем и приложений.

ГИПЕРССЫЛКИ В HTML

Главным признаком HTML-документа является наличие в нём гиперссылок (или просто ссылок) на другие документы, сайты, файлы, картинки и т. д. Именно возможность вставлять в страницы ссылки на объекты вне неё и сделала Интернет столь популярным и удобным для использования. Поэтому при создании своего сайта всегда помни про «магию» ссылок.

Анкор ссылки (от англ. «anchor» - «якорь») – то, что находится в коде гиперссылки между тегами < A> и < /A>. Например, там может быть:

- Текст
 - Код: < a href="http://www.seoded.ru/">Анкор ссылки< /a>
 - Вид в браузере: [Анкор ссылки](http://www.seoded.ru/)
- Адрес ссылки
 - Код: < a href="http://www.seoded.ru/">http://www.seoded.ru/< /a>
 - Вид в браузере: http://www.seoded.ru/

Ссылки бывают внешние и внутренние, графические, а также бывают абсолютные и относительные. Внешние ссылки ведут за «пределы» html-страницы, внутренние на различные части исходной страницы. Графический – это объект, по которому нужно щелкнуть для перехода, содержат какую-либо картинку. Все эти разновидности ссылок создаются в HTML с помощью тега < A>< /A> (сокращение от англ. Anchor – якорь). Рассмотрим его поподробнее.

Ссылки на файл, ссылки на сайт, ссылки на страницу

У многих юных разработчиков возникают вопросы о том, чем отличается ссылка на файл от ссылки на сайт или его отдельную страницу. Ответ такой: ничем. Все перечисленные ссылки являются внешними по отношению к **исходной** странице и создаются одним и тем же способом.

Итак, **код ссылки на сайт** выглядит следующим образом:

```
< a href="http://www.seoded.ru/">Ссылка на сайт< /a>
```

Код **ссылки на страницу** представлен ниже:

```
< a href="http://www.seoded.ru/beginner.html">Ссылка на страницу< /a>
```

Код **ссылки на файл** имеет следующий синтаксис:

```
< a href="http://www.seoded.ru/downloads/soft/colortfinal1.5.rar">Ссылка на  
файл< /a>
```

Таким образом, все типы ссылок создаются абсолютно одинаково. Разница только в адресе объекта, на который нужно сослаться.

Внешние ссылки

Для создания внешней ссылки на сайт, страницу или файл служит атрибут тега **< A>** – **href**. В качестве значения этот атрибут принимает **URL-адрес** сайта, страницы или файла. Между тегами **< A>** и **< /A>** располагается видимая часть ссылки (анкор ссылки), то есть то, что пользователь видит на экране браузера. Анкор ссылки может быть как обычным текстом (текстовая ссылка), так и графическим изображением (ссылка-картинка). Ссылка-картинка создаётся вставкой тега **< IMG>** между тегами **< A>** и **< /A>**. Синтаксис создания ссылки выглядит так:

```
< a href="URL-адрес или имя файла">текст ссылки (анкор)< /a>
```

Например, чтобы создать текстовую ссылку на главную страницу сайта, нужно написать следующий HTML-код:

```
< a href="http://www.seoded.ru/">Главная страница сайта Seoded.ru< /a>
```

Внутренние ссылки

Для комфортного перемещения по страницам с большим количеством контента используются внутренние ссылки. Внутренние ссылки создаются по тому же принципу, что и внешние. Только в значении атрибута **href** указывается «якорь» ссылки. «Якорь» создаётся атрибутом **name**:

```
< a name="имя якоря">текст< /a>
```

Имя «якоря» задаётся произвольно. Тут стоит сказать, что не все браузеры понимают русские имена «якорей», поэтому рекомендую

использовать латиницу. Текст между тегам `< A>< /A>` для создания «якоря» не обязателен и, чаще всего, не указывается.

«Якорь» располагается в тех местах страницы, в которые должен попасть пользователь после щелчка по ссылке.

Как уже было сказано выше, в атрибуте **href** внутренней ссылки вместо адреса указывается имя нужного «якоря» с обязательным символом решётки (#) перед ним. Разберём на примере.

Создадим «якорь» с именем **zagolovok** и разместим его в коде страницы рядом с заголовком лабораторной. Код «якоря» следующий:

```
< a name="zagolovok">< /a>
```

HTML-код внутренней ссылки будет выглядеть так:

```
< a href="#zagolovok">К заголовку< /a>
```

После щелчка по этой ссылке, вы попадаете к заголовку вашей страницы.

Графические ссылки

Для того, чтобы создать ссылку-картинку, нужно вместо текста использовать изображение. Пример такой ссылки выглядит следующим образом:

```
< a href="http://www.seoded.ru/">< img src="logo.jpg">< /a>
```

По умолчанию, браузер окружает картинку в графической ссылке рамкой. Если это нежелательно, то атрибуту **border** тега **IMG** нужно присвоить значение 0:

```
< a href="http://www.seoded.ru/">< img src="logo.jpg" border="0">< /a>
```

Текстовые и графические ссылки можно комбинировать. Синтаксис выглядит следующим образом:

```
< a href="http://www.seoded.ru/">< img src="logo.jpg"> Главная  
страница< /a>
```

Абсолютные и относительные ссылки

Также ссылки бывают **абсолютными** и **относительными**. Абсолютная ссылка – это обычная ссылка вида:

```
< a href="http://www.seoded.ru/">Главная страница< /a>
```

А вот с относительными ссылками немного посложнее. В таких ссылках адрес указывается либо **относительно** корневой папки сайта (той, в которой лежит главная страница), либо относительно исходной страницы. Такие ссылки нужны, например, в том случае, если сайт лежит на домашнем компьютере. Или это не сайт, а страница-указатель на другие документы.

Разберём сначала создание ссылки **относительно** **корневой папки** сайта.

Допустим нам нужно сослаться на страницу **klienty.html**, которая лежит **в одной папке** с главной страницей сайта. Тогда код относительной ссылки примет вид:

```
< a href="/klienty.html">Клиенты< /a>
```

То есть, при формировании ссылки относительно корневой папки сайта опускается начальная часть адреса **http://www.seoded.ru** и остается всё остальное. Использование слеша «/» в начале **обязательно!**

Теперь рассмотрим создание гиперссылок **относительно исходной страницы**. Допустим, пользователь имеет страницу **price.html** (исходная страница) и с неё нужно сослаться на страницу **klienty.html**. Возникают следующие типовые варианты:

1. Страницы **price.html** и **klienty.html** находятся **в одной папке**. Тогда код ссылки будет таким:

```
< a href="klienty.html">Клиенты< /a>
```

2. Страница **klienty.html** и папка **zakazy** находятся **в корневой папке сайта**, страница **price.html** лежит **в папке zakazy** (т. е. страница **klienty.html** относительно исходной страницы **price.html** лежит **на один уровень выше**). Код станет таким:

```
< a href="../klienty.html">Клиенты< /a>
```

Две точки показывают, что нужно выйти из текущей папки на уровень выше.

3. Страница **klienty.html** и папка **zakazy** находятся **в корневой папке сайта**, папка **mebel** лежит **в папке zakazy**, страница **price.html** лежит **в папке mebel** (т. е. страница **klienty.html** относительно исходной страницы **price.html** лежит **на два уровня выше**). Код ссылки примет вид:

```
< a href="../../klienty.html">Клиенты< /a>
```

Т. е. каждый уровень обозначается двумя точками и слешем «/».

4. Страница **price.html** (исходная страница) и папка **zakazy** находятся **в корневой папке сайта**, страница **klienty.html** лежит **в папке zakazy** (т. е. теперь страница **klienty.html** относительно исходной страницы **price.html** лежит **на один уровень ниже**). Теперь код ссылки будет таким:

```
< a href="zakazy/klienty.html">Клиенты< /a>
```

В этом случае точки и слеша не ставятся.

5. Страница price.html (исходная страница) и папка zakazy находятся в корневой папке сайта, папка mebel лежит в папке zakazy, страница klienty.html лежит в папке mebel (т. е. теперь страница klienty.html относительно исходной страницы price.html лежит на два уровня ниже). Код ссылки такой:

```
< a href="zakazy/mebel/klienty.html">Клиенты< /a>
```

6. В корневой папке сайта лежат две папки: zakazy и oplata. Страница klienty.html лежит в папке zakazy, исходная страница price.html лежит в папке oplata (т. е. обе страницы лежат в разных папках на уровень ниже от корневой папки сайта). Код ссылки станет следующим:

```
< a href="../zakazy/klienty.html">Клиенты< /a>
```

По аналогии создаются относительные ссылки и в том случае, если страницы лежат на разных уровнях относительно корневой папки. Будет меняться лишь количество указателей уровня (две точки и слеш) и адрес страницы, на которую ставится ссылка.

Атрибуты ссылок

Атрибут	Описание, принимаемое значение
download	Дополняет атрибут href и сообщает браузеру, что ресурс должен быть загружен в момент, когда пользователь щелкает по ссылке, вместо того, чтобы, например, предварительно открыть его (как PDF-файл). Задавая имя для атрибута, таким образом задается имя загружаемому объекту. Разрешается использовать атрибут без указания его значения: <pre> Загрузить отчет за Февраль 2014</pre>
href	Значением атрибута является URL-адрес документа, на который указывается ссылка.
hreflang	Определяет язык связанного веб-документа. Используется только вместе с атрибутом href. Принимаемые значения — аббревиатура, состоящая из пары букв, обозначающих код языка, например: <pre>Anysite</pre>
media	Определяет, для каких типов устройств файл будет оптимизирован. Значением может быть любой медиа-запрос.
ping	Отслеживает URL-адреса ресурсов, по которым переходил пользователь.

<div data-bbox="220 593 279 627" data-label="Text">rel</div>	<p>Дополняет атрибут <code>href</code> информацией об отношении между текущим и связанным документом. Принимаемые значения:</p> <ul style="list-style-type: none"> <code>alternate</code> – ссылка на альтернативную версию документа (например, печатную форму страницы, перевод или зеркало). <code>author</code> – ссылка на автора документа. <code>bookmark</code> – постоянный URL-адрес, используемый для закладок. <code>help</code> – ссылка на справку. <code>license</code> – ссылка на информацию об авторских правах на данный веб-документ. <code>next/prev</code> – указывает связь между отдельными URL. Благодаря этой разметке Google может определить, что содержание данных страниц связано в логической последовательности. <code>nofollow</code> – запрещает поисковой системе переходить по ссылкам на данной странице или по конкретной ссылке. <code>noreferrer</code> – указывает, что переходе по ссылке браузер не должен посылать заголовок HTTP-запроса (Referrer), в который записывается информация о том, с какой страницы пришел посетитель сайта. <code>prefetch</code> – указывает, что целевой документ должен быть кэширован, т.е. браузер в фоновом режиме загружает содержимое страницы к себе в кэш. <code>search</code> – указывает, что целевой документ содержит инструмент для поиска. <code>tag</code> – указывает ключевое слово для текущего документа.
<div data-bbox="220 1182 327 1216" data-label="Text">target</div>	<p>Указывает на то, в каком окне должен открываться документ, к которому ведет ссылка. Принимает следующие значения:</p> <ul style="list-style-type: none"> <code>_self</code> – страница загружается в текущее окно; <code>_blank</code> – страница открывается в новом окне браузера; <code>_parent</code> – страница загружается во фрейм-родитель; <code>_top</code> – страница загружается в полное окно браузера.
<div data-bbox="220 1355 295 1388" data-label="Text">type</div>	<p>Указывает <u>MIME-тип</u> файлов ссылки, т.е. расширение файла, носит больше справочную информацию.</p>

ЗАДАНИЕ

1. Спроектируйте и реализуйте 4 вида логически организованных групп гипертекстовых документов (HTML-файлов), систем ссылок внутри документов, представляющими пользователю удобную и интуитивно понятную навигацию по гипертексту.

1.1. Ссылки на файл, на сайт, на страницу

1.2. Внешние ссылки

1.3. Графические ссылки

1.4. Внутренние ссылки

2. Согласно индивидуальному варианту, с помощью CSS-стилей, реализуйте анимацию гипертекстовых ссылок (для реализации данного задания смотрите приложение А).

3. Оформите отчет. В отчете обязательно должен быть сценарий в виде листинга с комментариями и скриншоты работы перехода по ссылкам.

4. Продемонстрируйте выполненную работу преподавателю.

ИНДИВИДУАЛЬНЫЕ ВАРИАНТЫ

1.	Зачеркнутая ссылка или текст при наведении другим цветом
2.	Изменение цвета ссылки при наведении на нее
3.	Подчеркивание у ссылки при наведении
4.	Анимация с подчёркивающей и исчезающей линией у ссылки
5.	Анимация с подчёркивающей линией у ссылки
6.	Изменение фона и цвета ссылки

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие компоненты пользовательского интерфейса используются в интерфейсе типа "Hand User Interface"?

2. Какова технология создания гиперссылок?

3. Перечислите атрибуты ссылок.

4. Анкор ссылки – это?

ПРИЛОЖЕНИЕ А

1. Зачеркнутая ссылка или текст при наведении другим цветом *HTML-код:*

```
<!-- Ссылка -->
<div class="crossed"><a href="#" class="crossed-link" title="">Lorem Ipsum</a></div>
<!-- Текст -->
<div class="crossed"><p class="crossed-text">Lorem Ipsum</p></div>
```

CSS-код:

```
/* Ссылка */
a.crossed-link {
  color: #f66d52; /* Цвет ссылки */
  font-weight: 400;
  text-transform: uppercase;
  text-decoration: none;
  font-size: 1em;
  position: relative;
  display: inline-block;
}
a.crossed-link:before {
  content: '';
  position: absolute;
  width: 100%;
  height: 1px; /* Высота зачеркивающей линии */
  background: #666; /* Цвет зачеркивающей линии */
  top: 45%;
  animation: out 0s cubic-bezier(1, 0, 0.58, 0.97) 1 both;
}
div.crossed:hover a.crossed-link:before {
  animation: in 0.2s cubic-bezier(1, 0, 0.58, 0.97) 1 both;
}

/* Текст */
p.crossed-text{
  color: #ffa000; /* Цвет текста */
  font-weight: 400;
  text-transform: uppercase;
  font-size: 1em;
  position: relative;
  display: inline-block;
}
```

```
p.crossed-text:before {
  content: '';
  position: absolute;
  width: 100%;
  height: 1px; /* Высота зачеркивающей линии */
  background: #666; /* Цвет зачеркивающей линии */
  top: 45%;
  animation: out 0s cubic-bezier(1, 0, 0.58, 0.97) 1 both;
}

div.crossed:hover p.crossed-text:before {
  animation: in 0.2s cubic-bezier(1, 0, 0.58, 0.97) 1 both;
}

/* Анимация */
@keyframes in {
  0% {
    width: 0;
    left: 0;
    right: auto;
  }
  100% {
    left: 0;
    right: auto;
    width: 100%;
  }
}

@keyframes out {
  0% {
    width: 100%;
    left: auto;
    right: 0;
  }
  100% {
    width: 0;
    left: auto;
    right: 0;
  }
}

@keyframes show {
  0% {
    opacity: 0;
  }
}
```

```
transform: translateY(-10px);
}
100% {
opacity: 1;
transform: translateY(0);
}
}
```

2. Изменение цвета ссылки при наведении на нее

HTML-код:

```
<a href="#" class="change-color-link" title="">Lorem Ipsum</a>
```

CSS-код:

```
.change-color-link{
color: #f66d52; /* Цвет ссылки */
font-size: 2em;
font-weight: bold;
padding: 20px 0px;
line-height: 1;
text-decoration: none;
-webkit-transition: all 1s ease;
transition: all 1s ease; /* Время изменения цвета */
}
.change-color-link:hover{
color: #ffa000; /* Меняем цвет ссылки */
}
```

3. Подчеркивание у ссылки при наведении

HTML-код:

```
<a href="#" title="Lorem Ipsum">Lorem Ipsum</a>
```

CSS-код:

```
a {
color: #00bfff; /* Цвет обычной ссылки */
text-decoration: none; /* Убираем подчеркивание */
}
a:visited {
color: #8a2be2; /* Цвет посещённой ссылки */
}
a:hover {
color: #f66d52; /* Цвет ссылки при наведении на нее курсора мыши */
text-decoration: underline; /* Добавляем подчеркивание */
}
```

4. Анимация с подчёркивающей и исчезающей линией у ссылки

HTML-код:

```
<a href="#" class="underline-one">Lorem Ipsum</a>
```

CSS-код:

```
.underline-one {
    color: #00bfff; /* Цвет обычной ссылки */
    position: relative;
    cursor: pointer;
    text-decoration: none; /* Убираем подчеркивание */
}

.underline-one:after {
    content: "";
    display: block;
    position: absolute;
    right: 0;
    bottom: -3px;
    width: 0;
    height: 2px; /* Высота линии */
    background-color: black; /* Цвет подчеркивания при исчезании линии*/
    transition: width 0.5s; /* Время эффекта */
}

.underline-one:hover:after {
    content: "";
    width: 100%;
    display: block;
    position: absolute;
    left: 0;
    bottom: -3px;
    height: 2px; /* Высота линии */
    background-color: red; /* Цвет подчеркивания при появлении линии*/
    transition: width 0.5s; /* Время эффекта */
}
```

5. Анимация с подчёркивающей линией у ссылки

HTML-код:

```
<div class="link-ease-in-out">
    <a href="#" title="Lorem Ipsum">Lorem Ipsum</a>
</div>
```

CSS-код:

```
.link-ease-in-out a {
    display: inline-block;
    color: #00bfff; /* Цвет обычной ссылки */
    line-height: 1;
    text-decoration: none; /* Убираем подчеркивание */
    cursor: pointer;
}
.link-ease-in-out a:after {
    background-color: #f66d52; /* Цвет линии при наведении на нее курсора мыши */
    display: block;
    content: "";
    height: 2px; /* Высота линии */
    width: 0%;
    -webkit-transition: width .3s ease-in-out;
    -moz--transition: width .3s ease-in-out;
    transition: width .3s ease-in-out;
}
.link-ease-in-out a:hover:after,
.link-ease-in-out a:focus:after {
    width: 100%;
}
```

6. Изменение фона и цвета ссылки

HTML-код:

```
<p>Lorem Ipsum</p>
```

CSS-код:

```
p {
    color: #5a5a5a; /* Цвет текста */
    background: #ffc; /* Цвет фона */
    padding: 10px; /* Поля вокруг текста */
    transition: 1s linear; /* Время изменения */
}
p:hover {
    color: #fff; /* Цвет текста */
    background: #f66d52; /* Цвет фона */
}
```