

Федеральное агентство по рыболовству БГАРФ ФГБОУ ВО «КГТУ» Калининградский морской рыбопромышленный колледж

Утверждаю Заместитель начальника колледжа по учебно-методической работе А.И.Колесниченко

МДК 05.02.06 ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

Методические указания для выполнения практических занятий по специальности 09.02.07 Информационные системы и программирование (специалист по информационных системам)

МО-09 02 07-ПМ.05. МДК.05.02. Тема 5.2.6. ПЗ

РАЗРАБОТЧИК

Халина Е.Н.

ЗАВЕДУЮЩИЙ ОТДЕЛЕНИЕМ

Судьбина Н.А.

ГОД РАЗРАБОТКИ

2025

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

Содержание

ВВЕДЕНИЕ	3
ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ ЗАНЯТИЙ	4
Практическое занятие №1 Применение гипертекстовых ссылок	5
Практическое занятие № 2 Построение таблиц различного вида. Обтекание таблицы	
текстом	8
Практическое занятие №3 Формы ввода данных	16
Практическое занятие № 4 Использование изображений. Карты МАР	20
Практическое занятие №5 Пример использования таблиц стилей. Теги «div» и «span»	23
Практическое занятие № 6 Шрифты в CSS	29
Практическое занятие №7 Поля в CSS. Списки в CSS.	33
Практическое занятие № 8 Позиционирование в CSS	39
Практическое занятие № 9 Контекстные селекторы	48
Практическое занятие №10 Работа над html страницей по индивидуальному заданию	51

ВВЕДЕНИЕ

Методические указания для выполнения практических занятий разработаны в соответствии с рабочей программой ПМ.05 «Проектирование и разработка информационных систем» по специальности 09.02.07 «Информационные системы и программирование»

Данное методическое пособие раскрывает рекомендации по выполнению практических занятий для темы 5.2.6 Интернет программирование по теме «Введение в HTML и CSS» и предусматривает проведение 10 практических занятий-20 академических часа в соответствии с учебным планом.

Целью проведения практических занятий является формирование профессиональных компетенций согласно ФГОС, закрепление теоретических знаний и приобретение необходимых практических навыков и умений по отдельным темам курса.

Внеаудиторная самостоятельная работа выполняется в отдельных тетрадях в виде конспекта (реферата, презентации).

Перед проведением практического занятия обучающиеся должны повторить пройденный материал по данной теме, уяснить цель занятия, ознакомиться с последовательностью его проведения. Преподаватель должен проверить знания обучающихся и готовность к выполнению задания.

Для ведения записей (отчетов) по выполнению практических занятий обучающийся должен иметь отдельную тетрадь с полями. Записи должны вестись четко, аккуратно, разборчивым почерком. Каждая новая работа оформляется с новой страницы.

Для закрепления знаний по теме и подготовки к защите в конце каждой работы приводится перечень вопросов для самоконтроля. К защите обучающийся должен представить оформленный отчет по работе согласно задания с выводами по работе, должен знать теорию по данной теме, уметь пояснить, как проводилась работа и уметь анализировать полученные результаты. МО-09 02 07- МДК 05.02.06.ПЗ

ΚΜΡΚ БΓΑΡΦ ΦΓБΟУ ΒΟ «ΚΓΤУ»

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

C. 4/52

ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

№ п/п	Наименование практических занятий	Кол-во часов
1	Применение гипертекстовых ссылок.	2
2	Построение таблиц различного вида. Обтекание таблицы текстом	2
3	Формы ввода данных	2
4	Использование изображений. Карты МАР.	2
5	Пример использования таблиц стилей. Теги «div» и «span»	2
6	Шрифты в CSS.	2
7	Поля в CSS. Списки в CSS.	2
8	Позиционирование в CSS.	2
9	Контекстные селекторы	2
10	Работа над html страницей по индивидуальному заданию	2
	Итого	20

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

Практическое занятие №1 Применение гипертекстовых ссылок

Цель занятия: Применение форматирования текстовой информации.

Исходные материалы и данные:

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

Формирование или форматирование текстовой информации представляет собой изменение вида текста, применяя различные начертания и использования эффектов. В различных средах хранения информации, формирование текста используются по-своему. В нашем случае форматирование базируются на понятии тег или дескриптор, что подразумевает, какой-то объект. Объявление тегов бывают трех видов: открывающиеся – <название тега; атрибуты> закрывающиеся – </название тега; атрибуты> использование случае случае случае на видов.

Использование тегов для форматирования текста происходит так:

<название тега; атрибуты>Текст</название тега >

Мы рассмотрим самые основные теги для форматирования текстовой информации:

Код в HTML	Описание	Пример
Текст	Жирное начертание текста	Текст
<i>Текст</i>	Курсивное начертание текста	Текст
 Текст 	Верхний индекс (надстрочный)	e=mc ²
 Текст 	Нижний индекс (подстрочный)	H ₂ O
<pre>Teкcт</pre>	Текст пишется как есть, включая все пробелы	Текст
Teкcт	Курсивный текст	Текст
Teкcт	Жирное начертание текста	Текст
<u></u> Текст 	Нижнее подчеркивание	Текст

Любые теги форматирования текста можно использовать совместно друг с другом. Чтобы сделать текст одновременно жирным и курсивным шрифтом используется сочетание тегов ****и **<i>**(рис. 1.1.).

	ΚΜΡΚ БΓΑΡΦ ΦΓБΟУ ΒΟ «ΚΓΤУ»	
МО-09 02 07- МДК 05.02.06.ПЗ		
		C. 6/52
ANON CHORNO BUDGBAN MB	<>>>>>нис	BHUKOB
<і>Лух и жизненная сила<	/i> в нём постирия scorepmencesa	BRAKUD.
Но вот бела — никто не с	Meet IDNHETS ETO BUBOB.	
🙆 Текст		
	- 🚱 🖒 🔍 - Google	
Он словно	вырезан из камня, стоек и неподвижен в	
отличие от	его противников. Дух у жизненная сила в	
нём достиг.	па <i>совершенства.</i> Но вот беда — никто не	
смеет прин	ять его вызов.	

Рисунок 1.1.

Следует отметить, что теги **<b**> и ****, также как **<i>** и **** являются не совсем эквивалентными и заменяемыми. Первый тег ****— является тегом физической разметки и устанавливает жирный текст, а тег ****— тегом логической разметки и определяет важность помеченного текста. Такое разделение тегов на логическое и физическое форматирование изначально предназначалось, чтобы сделать HTML универсальным, в том числе не зависящим от устройства вывода информации. Теоретически, если воспользоваться, например, речевым браузером, то текст, оформленный с помощью тегов **** и ****, будет отмечен по-разному. Однако получилось так, что в популярных браузерах результат использования этих тегов равнозначен.

Задание:

1. Создадим НТМ скумент по шаблону. Рис.1.2.

<html> <head> <title></title> </head>

<body>

</body>

Рисунок 1.2.

2. Дадим название страницы в теге TITLE-Текстовая информация в HTML.

3. Создадим тег – заголовок 2 (<h2></h2>) с тем же названием, что и TITLE.

4. Создадим тег параграф – и в него занесём следующий текст: HTML

(от англ. HyperTextMarkupLanguage — «язык разметки гипертекста») — стандартный язык разметки документов во Всемирной паутине.

Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж 5. Отформатируем текст, таким образом:



Рисунок 1.3.

Исходный код:

6. Добавим еще один параграф и запишем туда такой текст:

Под действием гидроксиламина альдегиды превращаются в оксимы: CH₃CHO + NH₂OH = CH₃C(=NOH)H + H₂O

И отформатируем, используя тег .

7. Как же работает тег ? Давайте посмотрим добавив его и внесем в него следующий стих, самое главное записать его в код надо прям так как есть:

Улица провалилась, как нос сифилитика.

Река - сладострастье, растекшееся в слюни.

Отбросив белье до последнего листика,

сады похабно развалились в июне.

Я вышел на площадь, выжженный квартал надел на голову, как рыжий парик. Людям страшно - у меня изо рта

шевелит ногами непрожеванный крик.

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

Результат работы:



Содержание отчёта:

1. Наименование практического занятия.

- 2. Цель занятия.
- 3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок

выполнения задания»

- 4. Список использованных источников.
- 5. Выводы и предложения.

Вопросы для самопроверки:

- 1. Какие шаблоны существуют при создании гипертекстовых ссылок?
- 2.Листинг написания
- 3.Как сделать ссылку в пределах одного документа?

Практическое занятие № 2 Построение таблиц различного вида. Обтекание таблицы текстом

Цель занятия:

Научиться строить таблицы различного вида

Исходные материалы и данные.

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

Существует два вида таблиц, простые и сложные. Различия сложной таблицы от простой являются в ее структуре. Сложная таблица имеет объединения, когда простая не имеет. Давайте посмотрим на примере:

Простая таблица:



Сложная таблица:

Как же происходит объединение ячеек в HTML? Объединение происходит, таким образом, мы знаем из чего, состоит начальный и одиночный тег. Он состоит из названия и атрибута. Так вот существует тег **<TD>**, который создает нам колонки 2 атрибута ROWSPAN и COLSPAN, которые имею значения.

ROWSPAN – объединяет строки. При использовании надо занеси в него количество объединяемых строк, указанное количество на последующей строке надо удалить.

Пример:



Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж COLSPAN – объединяет колонки. При использовании надо занеси в него количество объединяемых колонок, лишние колонки в строке надо удалить.



Задание:

Для начала нам надо представить себе, как же будет выглядеть таблица. Сделаем набросок:

№	Наименование	Распределение по предметам		о Кол-во	Колич	оличество часов J				Распределение по курсам и предметам						
п/п предмета	предмета					Из них		2 курс		3 курс		4 курс		5 курс		
		Экзаменов	Курс. проект	контрольных работ	Bcero	Уроки	Лаб. раб.	Курс проект	1 сем. 16 нед.	2 сем. 17 нед.	3 сем. 16 нед.	4 сем. 19 нед.	5 сем.	6 сем.	7 сем.	8 сем.
Фак	ультативные		Изучаемы	ых предметов												
предметы, консультации		Bcero	Курсовы	х проектов												
		Экзамен)B												

Следующим нашим шагом будет определение, сколько будет использоваться строк и столбцов.

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟΥ ΒΟ «ΚΓΤΥ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 11/52

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	7
коно	ультации		Экзамен	в													10
пре	меты,	Bcero	Курсовы	проектов													9
Фак	льтативные		Изучаемн	х предметов													8
																	7
																	6
																	5
																	4
		Экзаменов	Курс. проект	работ	Всего	Уроки	Лаб. раб.	Курс проект	16 нед.	2 сем. 17 нед.	5 сем. 16 нед.	ч сем. 19 нед.	5 сем.	6 сем.	7 . сем.	8 сем.	3
11/11	предмета	предметам		Кол-во контрольных		Изних	(2 курс 1 сем	2 can	3 курс 3 сем	A case	4 куј	pc	5 ку	oc	2
JNº TT/TT	Наименование	Распределе	ение по		Коли	ество	часов		Распре	делени	по кур	сам и п	едме	етам			Į-
20					7.0				70								11

В нашем случае строк будет 10, а столбцов 17.

Теперь определим, где будут наши объединения. Синим маркерам отмечены

вертикальные объединения, а фиолетовым горизонтальные.

N₂	Наименование	Распределе	ение по		Колич	ество	насов		Распре	делени	по кур	сам и п	едме	там			1
п/п	предмета	предметам		Кол-во		Из них	c.		2 курс		3 курс		4 ку	рс	5 кур	oc	2
		Экзаменов	Курс. проект	контрольных работ	Всего	Уроки	Лаб. раб.	Курс проект	1 сем. 16 нед.	2 сем. 17 нед.	3 сем. 16 нед.	4 сем. 19 нед.	5 сем.	6 сем.	7 . сем.	8 сем.	3
																	4
																	5
																	6
																	7
Фак	льтативные		Изучаемь	іх предметов													8
пред	меты,	Bcero	Курсовы	а проектов													9
конс	ультации		Экзамен	В													10
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	1

Создадим новый HTML документ и создадим таблицу с 10-ю строками и 17-ми колонками. После создания начнём редактировать таблицу.

Результат работы:

	Построени	е таблиц разл	×	-	-												
¢	← → C ↑ [] file:///C:/Users/Cepreй/Desktop/3.html											2 =					
	🛜 Рекомендуемые уз 🕒 🕒 Коллекция веб-фр 🧀 Импортировано из 🚫 Скины / Skins » Всё 🔵 Деятельность скла 🧧 Вакансии 🛛 👘																
N	Наимен	ование	Распределе	ние по		Колич	ество ч	асов		Распре	деление	е по кур	сам и пр	редме	там		
π /1	п/п предмета предметам			Кол-во		Из них			2 курс		3 курс		4 ку	рс	5 кур	, c	
			Экзаменов	Курс. проект	контрольных работ	Всего	Уроки	Лаб. раб.	Курс проект	1 сем. 16 нед.	2 сем. 17 нед.	3 сем. 16 нед.	4 сем. 19 нед.	5 сем.	6 сем.	7 сем.	8 сем.
	_																
Φ	акультатив	ные		Изучаеми	ых предметов												
пр	предметы, Всего Курсовы		х проектов														
ко	нсультаци	и		Экзамен	OB												

Код:

Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж

MO-09	02 07-	мдк	05.02	.06.ПЗ
	•= •.			

C. 12/52

```
№ п/п
Наименование предмета
Распределение по предметам
Кол-во контрольных paбот
Количество часов
Распределение по курсам и предметам
Bcero
Из них
2 курс
3 курс
4 kypc
5 курс
Экзаменов
Kypc.проект
Ypoки
Лаб. paб.
Kypc проект
1 сем. 16 нед.
2 сем. 17 нед.
3 сем. 16 нед.
4 сем. 19 нед.
5 сем.
6 сем.
7 сем.
8 сем.
&nbsp
```

	ΚΜΡΚ БΓΑΡΦ ΦΓБΟУ ΒΟ «ΚΓΤУ»	
МО-09 02 07- МДК 05.02.06.113	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 13/52
%td>%nbsp		
<ta> </ta>		
<ta>&npsp</ta>		
<u><u><u><u><u><u><u><u><u><u><u><u><u><</u></u></u></u></u></u></u></u></u></u></u></u></u>		
<u><u><u><u><u><u><u><u><u><u><u><u><u><</u></u></u></u></u></u></u></u></u></u></u></u></u>		
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% <td></td> <td></td>		
%		
//d²		
•		

Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж

	ΚΜΡΚ Η ΑΡΦ ΦΙ ΕΟΥ ΒΟ «ΚΙ ΤΥ»		
wo-os oz or- wiдn os.uz.uo.113	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 14/52	
		J. 17/02	
<td co<="" rowspan="3" td=""><td>) SPAN=2>Факультативные предметы, консульта</td><td>auuu⊲/td></td></td>	<td>) SPAN=2>Факультативные предметы, консульта</td> <td>auuu⊲/td></td>) SPAN=2>Факультативные предметы, консульта	auuu⊲/td>
B		cero	ации чар
I/3		VU2EMLIX DEEMETOR	
	участых предметов члах		
%td>			
%td>			
//d>			
&td>			
Ky	рсовых проектов		
Эк	заменов		

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

Содержание отчёта:

- 1. Наименование практического занятия.
- 2. Цель занятия.

3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок выполнения задания»

- 4. Список использованных источников.
- 5. Выводы и предложения.

Вопросы для самопроверки:

- 1. Какие теги создают столбцы таблицы?
- 2. Какие теги создают строки таблицы?
- 3. Перечислите теги для объединения ячеек?

Практическое занятие №3 Формы ввода данных

Цель занятия:

Научиться создавать формы и обрабатывать данные, полученные из формы.

Исходные материалы и данные.

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

Формы предназначены в HTMLдля отправки данных на сервер. Объявление форм происходит тегом **<form>**.Саму форму вы не увидите, это не визуальный тег. Основные атрибуты формы **method** и **action**. Чем являются, эти атрибуты рассмотрим: **method** метод отправки данных на сервер, а **action** – файл который будет принимать данные на сервере.

Для того чтобы отправить данные их надо с начало куда-то ввести. Для этого в HTMLectь дополнительные теги, которые встраиваются в форму.

Основные теги: <input/>, <button/>,<select></select>,<option></option>.

Рассмотрим по порядку теги.

<input/>– имеет основные атрибуты: type, value. Данный тег имеет множество задач благодаря выбору его типа (type). Описания основных задач:

	text	Ввод текстовой информации
	button	Кнопка
	hidden	Невидимые данные в виде текста
	checkbox	Выбор данных из группы данных
type	radio	Выбор одних данных из группы данных
	file	Выбор файла (пути к файлу) с клиента
	password	Ввод пароля
	reset	Сброс введенных данных
	submit	Отправка данных

Разберем по ближе два типа*checkboxuradio*. Для объявления *checkbox, radioy* нас появляется еще один дополнительный атрибут <u>checked="checked"</u>, он отвечает за выбранные данные. Рассмотрим пример с *checkbox:*

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟУ ΒΟ «ΚΓΤУ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 17/52



Для объявления типа *radio*нам понадобится добавить еще один атрибут *name*и создать несколько <input/>тегов, с одинаковыми именами (name).Смотрите пример:

<pre><iorm action="" method="get" name="Irm"> <input name="r1" type="radio"/></iorm></pre>	»
<input <u="" name="r1" type="radio"/> Checked="checked"/> <input name="r1" type="radio"/>	
<input name="r1" type="radio"/>	

**
button/>**– имеет основные атрибуты: **type**, **value**. Он имеет три типа: button, reset и submit.

<select></select>– Контейнер для создания списка выбора данных.

<option></option>–имеет основной атрибут selected="selected", предназначенный для выделения данных в меню.OPTION относится к SELECT'у для подразделения данных.

Пример объявления меню в форме:



Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟΥ ΒΟ «ΚΓΤΥ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 18/52

Задания:

Для начала надо создать HTMLдокумент. Мы создадим такую форму:



Создадим форму, и чтобы наша форма выглядела ровно, создадим в форме таблицу из 2 столбцов и 8 строк с BORDER=0.

Вот что должно получиться:

```
<form>
&ubsp;
&ubsp;
&ubsp;
&ubsp;
&ubsp;
&ubsp;
&ubsp;
```

И далее таблица следует

Создадим нужные нам теги в нужном месте с нужным типом, а это <input/>, <select></select>, <option></option>.

Наш результат с кодом:



Содержание отчёта:

- 1. Наименование практического занятия.
- 2. Цель занятия.

3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок выполнения задания»

- 4. Список использованных источников.
- 5. Выводы и предложения.

Вопросы для самопроверки:

- 1. Что является обязательными параметрами функции fopen()?
- 2. При помощи, какой функции можно записывать данные в файл при помощи

PHP?

- 3. Что выступает в качестве обязательных параметров функции fwrite()?
- 4. Какую функцию используют для построчного считывания файла?
- 5. Сколько обязательных параметров имеет функция fgets()?

6. Какую функцию нужно использовать для того, чтобы считать файл как единое целое?

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

7. Возможно, ли содержимое сайта считать в массив? И если да то с помощью, какой функции?

8. Какая функция поможет определить размер файла?

Практическое занятие № 4 Использование изображений. Карты МАР

Цель занятия:

Закрепить навыки работы с изображениями.

Исходные материалы и данные.

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

Карты МАРимеют механизм для клиентских карт изображений. Элементы МАРимеет имя, по которому он может быть отослан к элементу IMG, и содержит элементы AREA, которые определяют горячие зоны на связанном изображении и связывают эти горячие зоны с URL.

Визуальный элемент МАР никак не проявляется в документе, однако, вместе со связанными структурами он создает изображение реагирующей на нажатие клавиш карты.

Основной синтаксис:

<MAP>

<AREA элемент >

</MAP>

Атрибут МАР: NAME.

Атрибуты AREA: HREF - адрес, ALT-текстовое описание, SHAPE (=RECT(прямоугольник), CIRCLE(круг), POLY(многоугольник)) - очертание области, COORDS-координаты.

Атрибут : USEMAP

Задание:

Для начала работы мы создадим несколько новых HTML документов. Одни из них будет главным. У вас должно получиться примерно так:

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ



Откроем наш главный файл (5.html), и создадим там меню, которое будет ссылаться на другие файлы.

Так же отметим в других файлах (5.1.html;5.2.html;5.3html) заголовок с названием файла и ссылкой на главную страницу(5.html).

Примерно должно получиться так.

WWW BEARING		
	← → C file:///C:/Users/Cepreй/Desktop/5.1.html ☆ =	UF
S WE THE CT	🚺 Рекомендуемые уз 🗋 📋 Коллекция веб-фр 🧰 Импортировано из »	
A BAR S.	назад на главную страницу	ство
	Страница 5.1	IR K
5.3		×
← → C A D fi	← → C ↑ [hile:///C:/Users/Cepreй/Desktop/5.2.htm]	
Рекомендуемые уз	🔽 Рекомендуемые уз 🏱 🏱 Коллекция веб-фр 🧀 Импортировано из	»
назад на главную стра	назад на главную страницу	
Страница 5.3	Страница 5.2	
7		

Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж Вернем к созданию меню. Откроем Paint, нарисуем такое изображение – меню

(обязательно должны присутствовать многоугольник, окружность и прямоугольник.):



Приступим к написанию кода. Создадим 3 тега IMG(укажем путь к изображению и названия usemap), MAP(дадим название) и 3 – AREA(укажем параметры адрес и shapec координатами):

Осталось только указать координаты. Снова откроем наше изображение и узнаем координаты многоугольника (звезды). Укажем их в строке <are href="5.1.html" shape=POLY coords="" /> coord. Так проделаем с

кругом и прямоугольником.

Содержание отчёта:

- 1. Наименование практического занятия.
- 2. Цель занятия.
- 3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок

выполнения задания»

- 4. Список использованных источников.
- 5. Выводы и предложения.

Вопросы для самопроверки:

- 1. В чем заключается принцип работы кода html special chars()?
- 2. Что является основным в работе с PHP?

3. Как можно преобразить age (возраст), о котором нам известно, что оно должно быть число?

4. Какие переменные автоматически установлены средствами РНР?

Практическое занятие №5 Пример использования таблиц стилей. Теги «div» и «span»

Цель занятия:

Закрепить навык работы с CSS

Исходные материалы и данные.

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Задание:

Прежде чем приступить к работе сделаем небольшой набросок нашей главной страницы. Вот сам набросок.

ΚΜΡΚ БΓΑΡΦ ΦΓБΟУ ΒΟ «ΚΓΤУ»

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

C. 24/52

	Меню управления
Название страницы	
	Главная станица

Теперь прикинем, из чего будет состоять наш сайт, разобьём на блоки.

	id=menu Меню управления
Название страницы	class=hText
class=aLasson2	class=ico_2
class=aLassonHTML class=aLassonHTML	class=ico_2
class=aLassonHTML2 class=aLassonHTML2	class=menuRight
id=lesson2 class=lesson block	с id=content Главная станица

Кода все расписали и продумали, приготовим изображения в паке img. Приступим к верстке сайта.CSS Сайта:

МО-09 02 07- МДК 05.02.06.ПЗ

```
<style type="text/css">
@charset "windows-1251";
/* CSS Document */
body{
   margin:0 auto;
a{
    text-decoration:none;}
#menu{
   width:100%;
   height:30px;
   background:#00BCF2;
   padding:0 0 0 0;
   margin:0 auto;
}
#menu ul{
   margin:0 0 0 0;
   list-style-type:none;
   width:1000px;
#menu ul li{
   float:left;
   color:#fff;
   font-size:16px;
   cursor:pointer;
   margin:5px 0 0 20px;
}
#content{
   margin:0 auto;
   width:1000px;
}
.lesson{
   width:660px;
   min-height:400px;
   float:left;
.menuRight{
   float:left;
   width:340px;
   height:240px;
}
```

C. 26/52

```
.hText{
   width:900px;
   height:36px;
   margin: 30px 0 10px 0;
   padding:0 0 0 100px;
   color:#00BCF2;
   font-size:34px;
.aHome{
   display:block;
   width:34px;
   height:30px;
   background: url(img/aHome%201.png) no-repeat;
.aHome:hover{
   background:url(img/aHome%203.png) no-repeat;
.aLessonHTML, .aLessonHTML2, .aMenu, .aLesson2{
   color:#FFFFFF;
   float:left;
   margin:10px 10px 10px 10px;
   padding:0 20px 0 20px;
   display:block;
   width:260px;
   height:130px;
.aLesson1 top{
    font-size:23px;
    font-weight:bold;
    margin:10px 0 0 0;
.aLesson2{
    margin-right:441px;
    padding-left:50px;
   padding-top:40px;
    font-size:24px;
    width:559px;
    height:43px;
    background: url(img/logo_book_1.png) no-repeat;
.aLesson2:hover{
    background-position: 0px -117px;
/*----Темно Синий блок-----Темно Синий
.aLessonHTML{
    background:#5C8DC2 url(img/blog.png) no-repeat 200px 70px;
    border:2px #88ABD2 solid;
1
.aLessonHTML:hover{
    border:2px #66CC33 solid;
    background:#5C8DC2 url(img/blog2.png) no-repeat 200px 70px;
```

МО-09 02 07- МДК 05.02.06.ПЗ

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

```
/*----Оранживый блок-----
.aLessonHTML2{
   background:#FF9966 url(img/blog.png) no-repeat 200px 70px;
   border:2px #FFCC99 solid;
3
.aLessonHTML2:hover{
   border:2px #66CC33 solid;
   background:#FF9966 url(img/blog2.png) no-repeat 200px 70px;
/*---
                                                             ----*/
.ico{
   display:block;
   cursor:pointer;
   margin:5px 0 0 0;
   height:22px;
   background:url(img/link-list.png) no-repeat 155px 0;}
.ico 2{
   display:block;
   cursor:pointer;
   margin-top:10px;
   padding:10px 10px 10px 10px;
   color:#fff;
   width:338px;
   height:22px;
   border:#bfeded solid 2px;
   background:#6CC;
    }
           .ico_2:hover{
               border:#6CC solid 2px;
           }
           .liHover{
              color:#fff;
              border-bottom:#fff 7px solid;
           }
           .li{
               color:#e8e8e8;
           }
           .li:hover{
                  color:#fff;
               3
           }
           </style>
```

HTML код сайта:

```
<div id="menu">
   style="margin-top:0;"><a href="#0" class="aHome"></a>
   Раздел 1
   Раздел 2
   </div>
<div id="content">
   <div class="hText">
       Название главной страницы
       <span style=" font-size:16px; color:#0072C6;">Главное меню</span>
   </div>
   <div id="lesson2" class="lesson block" style="display:block;">
       <a href="#1" class="aLesson2">
           Интересная ссылка
       </a>
       <a href="#2" class="aLessonHTML">
           <div class="aLesson1_top">Практическая работа 1</div>
           <b>Tema:</b> Название<br />
           <b>0писание:</b> описание
       </a>
       <a href="#3" class="aLessonHTML">
           <div class="aLesson1 top">Практическая работа 2</div>
           <b>Tema:</b> Название<br />
           <b>0писание:</b> описание
       </a>
       <a href="#4" class="aLessonHTML2">
           <div class="aLesson1_top">Практическая работа 3</div>
           <b>Tema:</b> Название<br />
           <b>0писание:</b> описание
       </a>
   </div>
   <div class="menuRight" style="display:block;">
      <a href="#5" class="ico_2" style="margin-top:32px;">Дополнительная ссылка 1</a>
      <a href="#6" class="ico_2">Дополнительная ссылка 2</a>
   </div>
</div>
```

Должен получиться такой результат:

Раздел 1 Раздел 2

Название главной страницы главное меню



Содержание отчёта:

- 1. Наименование практического занятия.
- 2. Цель занятия.

3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок выполнения задания»

- 4. Список использованных источников.
- 5. Выводы и предложения.

Вопросы для самопроверки:

- 1. Что задаётся тегом \$HOST='LOCALHOST';?
- 2. Как расшифровывается данный запрос \$QUERY = "SELECT * FROM

`MY_SQL_TABLE`";?

3. Что делает запрос \$RES = MYSQL_QUERY(\$QUERY);?

Практическое занятие № 6 Шрифты в CSS

Цель занятия: Закрепить навык преобразования шрифтов в CSS.

Исходные материалы и данные.

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

Чтобы понимать, о чем идет речь, выберите любой шрифт со страницы Google Fonts и откройте его, нажав на кнопку Quick-use ⊡.

Шаг 1: выберите начертание

Первым делом на странице выбранного шрифта отобразятся варианты его начертания, а также иконка спидометра, которая означает не что иное, как скорость загрузки шрифта. Чем больше стилей для шрифта вы выбираете, тем больше времени потребуется на его загрузку. Поэтому рекомендуется выбирать только те варианты начертания, которые планируется использовать.

Шаг 2: выберите алфавит

Далее на странице есть возможность выбрать набор символов: латиница, кириллица и т. д. В зависимости от шрифта, в нем могут быть доступны не все варианты алфавита. Аналогично предыдущему пункту, лучше поставить галочку только напротив того алфавита, который понадобится.

Шаг 3: добавьте код на сайт

Далее Google предлагает подключить шрифт одним из нескольких способов: или стандартным, или через директиву @import, или с помощью JavaScript. Мы рассмотрим первые два варианта.

Первый способ подразумевает добавление в HTML-код ссылки на сервер Google, откуда и скачивается шрифт. Вам необходимо скопировать уже готовый кусок кода и поместить его между тегами <head></head> в вашем HTML-документе.

Пример:

1 <head>

2 <link href='http://fonts.googleapis.com/css?family=Roboto&subset=latin,cyrillic'

- 3 rel='stylesheet' type='text/css'>
- 4
- 5 ...
- 6 </head>

Второй способ – подключение шрифта с помощью директивы @import. Готовый код находится во второй вкладке пункта 3 на странице выбранного Googleшрифта. Его нужно добавить в самое начало вашей таблицы стилей (в противном случае файл не импортируется).

Выглядит код примерно так:

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟΥ ΒΟ «ΚΓΤΥ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 31/52

1 @import url(http://fonts.googleapis.com/css?family=Roboto&subset=latin,cyrillic);

Особенность первого способа заключается в том, что вам понадобится добавлять ссылку на шрифт в заголовок каждой страницы, где планируется его использовать. Это легко осуществить на сайтах с небольшим количеством страниц, но проблематично для крупных ресурсов. Второй способ удобен тем, что код можно поместить в самое начало внешней таблицы стилей, и тогда все страницы, к которым подключена эта таблица, получат необходимый шрифт, который будет загружаться с помощью директивы @import.

Google Fonts		More scripts About Analytics New to Google Fonts
1. Choose the styles you want: © Roboto © Trin 100 © Trin 100 talle © Light 300 traile © Normal 400 traile © Normal 400 traile © Medium 500 traile © Bold 700 © Bold 700 © Bold 700 © Bold 900 traile	В чещах юга жил бы цитрус? Да но фальшивый экзампляр) В чещах юга жил бы цитрус? Да но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр! В чещах юга жил бы цитрус? Да, но фальшивый экзампляр!	Page Load 24 Inpact on page load time Trp: Using many font styles can slow down your webpage, so only select the fort styles to only select the fort styles that you actually need on your webpage.
2. Choose the character sets you want: □ Cyrillic Extended (cyrillic-ext)	reek-ext) □ Greek (greek) □ Vietnamese (vietnamese) □ Latin Extended (latin-ext) 🕏 Cyrillic (cyrillic)	Tip: If you choose only the languages that you need, you'l help prevent slowness on your webgage.
Standard @import Javascript		Instructions: To enhand your folloction
3. Add this code to your website: <pre>clink href='http://fonts.googleapis.com/css?family=Bobbtc</pre>	ésubset=latin,cytillic' rel='stylesheet' type='text/css'>	Instructions: 10 embed your Conection into your web page, copy the code as the first element in the <head> of your HTML doc ument. > See an example</head>

Скриншот: этапы настройки шрифта Google

Шаг 4: создайте стиль

После осуществления предыдущих шагов можно начать применять шрифт. Как записывается такое правило CSS, вы уже видели ранее:

- 1 p{
- 2 font-family: 'Roboto', sans-serif;
- 3 }

Если на первом шаге вы выбрали несколько вариантов начертания (к примеру, добавили вариант жирного шрифта Bold 700), тогда на третьем шаге код немного видоизменится:

- 1 k href='http://fonts.googleapis.com/css?family=Roboto:700,400&subset=latin,cyrillic'
- 2 rel='stylesheet' type='text/css'>

Или:

@import

url(http://fonts.googleapis.com/css?family=Roboto:700,400&subset=latin,cyrillic);

Чтобы затем придать шрифту жирное начертание, запишите CSS-стиль таким образом:

1 p{

1

- 2 font-family: 'Roboto', sans-serif;
- 3 font-weight: 700;

4 }

Примечание: в Google Fonts для обозначения насыщенности шрифта используются только условные единицы от 100 до 900. Так, нормальное начертание (по умолчанию) эквивалентно значению 400 (normal), а стандартное полужирное начертание эквивалентно 700 (bold).

Главными преимуществами сервиса Google Fonts является:

 – легкость использования (разобраться с сервисом сможет даже новичок, а весь необходимый код генерируется автоматически – остается только скопировать его);

- доступность шрифтов (нет необходимости за них платить);

 предусмотрены все используемые форматы шрифтов (это означает, что каждый браузер сможет загрузить именно тот формат шрифта, с которым он работает).

Среди недостатков сервиса – не очень большое разнообразие шрифтов, особенно кириллических. К слову, в интернете есть и другие похожие сервисы, например, ТуреКіt (платный).

Содержание отчёта:

1. Наименование практического занятия.

2. Цель занятия.

3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок выполнения задания»

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

4. Список использованных источников.

5. Выводы и предложения.

Вопросы для самопроверки:

1. Что делает запрос \$QUERY = "SELECT FIRSTNAME FROM `MY_SQL_TABLE`";?

 Что делает запрос \$QUERY = "SELECT SURNAME FROM `MY_SQL_TABLE` WHERE ID=5";?

 Что делает запрос \$QUERY = "ALTER TABLE `MY_SQL_TABLE` DROP `SURNAME`";?

Практическое занятие №7 Поля в CSS. Списки в CSS

Цель занятия: Закрепить навык организации полей и списков в CSS.

Исходные материалы и данные.

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

Двумя наиболее распространенными типами HTML-списков являются упорядоченный и неупорядоченный. Для их стилизации списков мы будем использовать свойство list-style CSS.

Ниже приведен пример неупорядоченного списка. В нем видно, что по умолчанию у списков используется выравнивание по левому краю:

- яблоко
 - Мяч
- Кот

list-style-type

Чтобы задать стиль маркера для пунктов списка, можно использовать свойство list-style-type.

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟΥ ΒΟ «ΚΓΤΥ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 34/52

Стандартное значение этого свойства – disc, но можно использовать и другие значения: circle, square, none. Кроме этого можно использовать такие значения, как upper-alpha, lower-alpha, upper-roman, lower-roman, decimal и т. д.

В приведенном ниже примере свойство list-style-type имеет значение square:

ul {

list-style-type : square;

}		
	 яблоко 	
	 Мяч 	
	• Кот	

Если вам вообще не нужны маркеры, то CSS list style type необходимо присвоить значение none:

ul {			
}	list-style-type : none;		
1	яблоко		
	Мяч		
	Кот		

Поля и отступы

Чтобы добавить в список отступы, можно использовать свойство margin.

В приведенном ниже примере использования **CSS ul li list style** мы прибавляем поля по **30px** над и под списком:

```
ul {
list-style-type : square;
margin-top : 30px;
margin-bottom : 30px;
```

}

МО-09 02 07- МЛК 05 02 06 ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟΥ ΒΟ «ΚΓΤΥ»	
МО 00 02 01 МДК 00.02.00.110	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 35/52
■ яб	локо	
= Ms	14	
= Ко	т	

В следующем примере мы полностью исключаем поля:

ul {

```
list-style-type : square;
```

```
margin : 0;
```

```
}
```

```
яблокоМячКот
```

В примере CSS ul list style мы убираем отступы:

```
ul {
```

```
list-style-type : square;
margin : 0;
padding : 0;
```

```
}
```

```
    яблоко
```

- Мяч
- Кот

Здесь мы задаем списку фиксированную ширину:

ul {

```
list-style-type : square;
width : 200px;
```

```
}
```

- Этот текст является элементом неупорядоченного списка.
- Это второй пункт того же списка.
- А это последний пункт.

List-style-position

Это свойство отвечает за то, как текст внутри **list style CSS** будет переходить на следующую строку. Стандартное значение **outside** означает, что при переносе строки текст также получит выравнивание по левому краю. Если выставлено значение **inside**, то строка будет перенесена под маркер списка.

В примере, приведенном ниже, представлен список, в котором свойство **list**style-position выставлено на outside (значение по умолчанию):

ul {

list-style-type : square; list-style-position : outside; width : 200px;

}

В следующем примере показано, что будет, если свойство **list-styleposition** будет иметь значение **inside**. Таким образом, текст на новой строке будет начинаться прямо под маркером:

ul {

list-style-type : square; list-style-position : inside; width : 200px;

```
}
```

Этот текст является элементом неупорядоченного списка.
Это второй пункт того же списка.
А это последний пункт. List-style-image

Свойство list-style-image позволяет использовать собственную пиктограмму в качестве маркера.

В примере **CSS ul list style** мы используем изображение звезды вместо маркера списка:

Сокращение list-style

Можно использовать сокращенное свойство **list-style** для назначения тех же стилей. В приведенном ниже примере мы задаем тип, расположение и изображение маркера списка:

ul {

}

list-style: square outside url(star.png);

Этот текст является элементом неупорядоченного списка.

Это второй пункт того же списка.

📕 А это последний пункт.

Список inline

По умолчанию, все пункты списка перечисляются с новой строки, то есть, вертикально. Если нужно выводить список горизонтально или строчно (inline), нужно присвоить элементам списка display inline.

В приведенном ниже примере представлен строчный список:

li {

display: inline;

}

яблоко Мяч Кот

Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟУ ΒΟ «ΚΓΤУ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 38/52

Стилизация упорядоченного списка

Теперь пришло время задать стиль для CSS ol list style. По умолчанию, liststyle-type для упорядоченного списка установлен при помощи десятичного числа:

- 1. яблоко
- 2. Мяч
- 3. Кот

Чтобы изменить тип, можно поменять значение на upper-alpha, lower-

alpha, upper-roman, lower-roman.

Ниже приведен пример, где list-style-type имеет значение upper-alpha:

ol {

list-style-type : upper-alpha;

}

```
А. яблоко
В. Мяч
С. Кот
```

В следующем примере list-style-type имеет значение lower-roman:

ol {

list-style-type : lower-roman;

}

А. яблоко В. Мяч С. Кот

Содержание отчёта:

1. Наименование практического занятия.

2. Цель занятия.

3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок выполнения задания»

- 4. Список использованных источников.
- 5. Выводы и предложения.

Вопросы для самопроверки:

- 1. Что делает запрос list-style-type?
- 2. Что делает запрос ol list style?

Практическое занятие № 8 Позиционирование в CSS

Цель занятия: Закрепить навык позиционирования в CSS.

Исходные материалы и данные.

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

При помощи CSS-позиционирования вы можете разместить, элемент точно в нужном месте страницы. Вместе с поплавками позиционирование даёт вам большие возможности для создания точного и навороченного дизайна.

Принципы CSS-позиционирования. Представим окно браузера как систему координат:



Принципы CSS-позиционирования - в том, что вы можете расположить бокс в системе координат где угодно.

Скажем, мы хотим позиционировать заголовок. При использовании боксовой модели заголовок выглядит так:

КМРК БГАРФ ФГБОУ ВО «КГТУ»

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

C. 40/52



Если мы хотим расположить его на 100рх от верхней границы документа и на 200рх слева, мы должны ввести следующий код CSS:

h1 {

position:absolute;

top: 100px;

left: 200px;

}

Вот результат:



Как видите, Позиционирование с помощью CSS - очень точная техника при размещении элементов. Это намного проще, чем использовать таблицы, прозрачные изображения или ещё что-нибудь подобное.

Абсолютное позиционирование. Элемент, позиционированный абсолютно, не получает никакого пространства к документе. Это означает, что после позиционирования он не оставляет после себя пустое пространство.

Для абсолютного позиционирования элемента свойство *position* должно иметь значение *absolute*. Вы можете использовать значения *left, right, top u bottom* для размещения бокса.

#box1 {

position:absolute;

top: 50px;

Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж left: 50px;

}

Относительное позиционирование. Чтобы позиционировать элемент относительно, установите в свойстве position значение relative. Разница между относительным абсолютным позиционированием состоит в том, как обсчитывается позиционирование.

Позиция элемента, размещаемого относительно, обсчитывается относительно его оригинальной позиции в документе. Это означает, что вы смещаете элемент вправо, влево, вверх или вниз. Таким образом, элемент всё ещё занимает в документе пространство после позиционирования.

#dog1 { position:relative; left: 350px; bottom: 150px;

}

Фиксированное позиционирование Fixed - По своему действию это значение близко к absolute, но в отличие от него привязывается к указанной свойствами *left, top, right u bottom* точке на экране и не меняет своего положения при прокрутке вебстраницы. Браузер Firefox вообще не отображает полосы прокрутки, если положение элемента задано фиксированным, и оно не помещается целиком в окно браузера. В браузере Opera хотя и показываются полосы прокрутки, но они никак не влияют на позицию элемента.

Static – Элементы отображаются как обычно. Использование свойств *left, top, right u bottom* не приводит к каким-либо результатам.

Z-index. Страница сайта двумерна. У неё есть ширина и высота. Z-index добавляет третье измерение, глубину.



ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

Чем выше этот индекс тем выше на странице расположен элемент. С помощью него мы можем добиться чтобы один элемент располагался поверх другого. По умолчанию, его значение - ноль. Отрицательные значения также допустимы.

На самом деле, z-index гораздо сложнее, чем я его здесь описываю, но это тема для отдельной статьи. Сейчас, главное запомнить саму идею третьего измерения и то, что только позиционируемые элементы могут использовать это свойство.

Задание 1:

Координаты слоя

Положение слоя в документе зависит от заданных значений стилевых свойств left, top, right и bottom, они устанавливают соответственно позицию слоя слева, сверху, справа и снизу. Точка отсчета определяется в зависимости от свойства position, которое принимает обычно значение relative (относительное положение) или absolute (абсолютное положение).

При абсолютном позиционировании слой располагается относительно родительского элемента или окна браузера, если родитель не задан, как показано на рис. 1.



Рисунок 1 - Положение слоя относительно окна браузера

В случае использования относительного позиционирования, координаты отсчитываются от исходного положения слоя (рисунок. 2). Здесь пунктиром выделено первоначальное положение слоя, а сплошной линией результат его перемещения.



Рисунок 2. - Положение слоя относительно исходного положения

Положительные значения свойства left сдвигают слой вправо, отрицательные — влево. С right дело обстоит наоборот, положительные значения этого свойства смещают слой влево, а отрицательные — вправо. Аналогичная ситуация и с парой top/bottom. Значения со знаком минус у свойства top поднимают слой вверх, а положительные значения опускают слой вниз. bottom же действует с точностью наоборот.

Заметим, что указание свойства position без упоминания координат (значений left, top, right или bottom) не меняет положение текущего слоя, но оказывает влияние на расположение близлежащих или вложенных слоев.

ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ

Пример 1. Размещение двух слоев

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
 <head>
 <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
 <title>Позиционирование</title>
 <style type="text/css">
  #leftcol {
   position: absolute; /* Абсолютное позиционирование */
   width: 200px; /* Ширина слоя */
   background: #800000; /* Цвет фона */
   color: white; /* Цвет текста */
   ٦
  #rightcol {
   position: relative; /* Относительное позиционирование */
   left: 200px; /* Смещаем слой вправо на ширину левого слоя */
   width: 550px; /* Ширина слоя */
   background: #e0e0e0; /* Цвет фона */
   padding: 10px; /* Поля вокруг текста */
  }
 </style>
 </head>
 <body>
 <div id="leftcol">Левая колонка</div>
 <div id="rightcol">Правая колонка </div>
</body>
</html>
```

Результат:

🕒 Позиционирование 🛛 🗙		×
← → C ♠ 🗋 file:///C:/	′Users/Cepгей/Desktop/html%20; 🕎	=
🧧 📔 Рекомендуемые уз 🗋 🗋 Н	Коллекция веб-фр	»
Левая колонка	Правая колонка	*
•		۶.

Несмотря на простоту решения, в данном случае следует учесть несколько моментов.

Использование position: absolute для левой колонки, хотя и не дает видимого результата, необходимо для того, чтобы правая колонка не «перескакивала» вниз, и слои располагались на одном уровне.

Положение правого слоя rightcol зависит от значения стилевого свойства left, оно в данном случае должно равняться ширине левого слоя, в этом случае слои будут плотно прилегать между собой. Впрочем, ничего не мешает изменять этот

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟΥ ΒΟ «ΚΓΤΥ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 45/52

параметр в ту или иную сторону. Тогда между слоями появится разделительная полоса или наоборот, слои станут накладываться друг на друга.

Свойство padding, добавляющее поля вокруг текста, если его использовать в стиле левого слоя leftcol, вносит искажения в макет страницы для браузера. Это связано с тем, что ширина элемента в браузере складывается из значений width, padding и margin. Поскольку добавление полей увеличивает ширину слоя, то правая колонка начинает накладываться поверх левой колонки. Чтобы устранить эту особенность, атрибут padding следует применить к стилю абзаца, который должен располагаться внутри слоя или изменить значение width. Также можно добавить еще один внутренний слой и указать поля для него (пример 2).

Пример 2. Поля внутри слоя.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
 <head>
 <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
 <title>Позиционирование</title>
  <style type="text/css">
   #leftcol {
   position: absolute; width: 200px; background: #800000; color: white;
  #leftcol div {
   padding: 10px; /* Поля вокруг текста в левой колонке */
   3
   #rightcol {
   position: relative; left: 200px; width: 550px; background: #e0e0e0;
   ¥.
   #rightcol div {
   padding: 10px; /* Поля вокруг текста в правой колонке */
   ł
  </style>
 </head>
 <body>
  <div id="leftcol">
   <div>Левая колонка</div>
 </div>
  <div id="rightcol">
   <div>Правая колонка</div>
 </div>
 </body>
</html>
```

Результат:

Позиционирование ×		X
← → C 🕆 🗋 file:///0	C:/Users/Cepгей/Desktop/html%20and%20css/lessc 公	=
🎧 Рекомендуемые уз 🗋 🗌	ј Коллекция веб-фр 🦳 Импортировано из	»
Левая колонка	Правая колонка	* =
•	III	F

Задание 2 Повторите фиксированное и абсолютное позиционирование Фиксированное позиционирование (fixed)

Задание элементу фиксированного позиционирования осуществляется посредством установки ему position: fixed.

Фиксированное позиционирование похоже на абсолютное, но в отличии от него оно всегда привязывается к краям окна браузера (viewport), и остаётся в таком положении даже при скроллинге страницы.

Фиксированное позиционирование применяется для закрепления на странице навигационных меню, кнопки «вверх», панелей с социальными кнопками и многого другого.



Абсолютное позиционирование (absolute)

Установка абсолютного позиционирования элементу осуществляется посредством задания ему position: absolute.

Этот тип позиционирования позволяет разместить элемент именно там, где вы хотите.

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟΥ ΒΟ «ΚΓΤΥ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 47/52

Позиционирование выполняется относительно ближайшего позиционированного предка.

Под позиционированным элементом понимается элемент с position, равным relative, absolute, fixed или sticky.

Если среди предков у элемента с position: absolute нет позиционированного элемента, то в этом случае он будет позиционироваться относительно HTML страницы, т.е. элемента body.

Когда элементу устанавливаем position: absolute без указания CSSсвойств (top, left, right и bottom), определяющих его положение, он будет находиться в том месте, в котором он был бы расположен, если бы находился в потоке (при этом при вычислении его положения учитываются только элементы, расположенные до него в коде и находящиеся в потоке).

При этом другие элементы его видеть не будут, и, следовательно, они будут расположены на странице, не обращая никакого внимание на него.



CSS-свойства для управления положением абсолютно позиционированного элемента работают по-другому чем с position: relative.

CSS-свойства top, bottom, left и right задают положение элемента относительно ближайшего позиционированного предка или body, если такого предка нет.

МО-09 02 07- МДК 05.02.06.ПЗ	ΚΜΡΚ БΓΑΡΦ ΦΓБΟΥ ΒΟ «ΚΓΤΥ»	
	ИНТЕРНЕТ ПРОГРАММИРОВАНИЕ	C. 48/52

Установить ширину (высоту) абсолютно позиционированному можно с помощью установки ему двух координат top и bottom (left и right).

Если элементу одновременно установить top, bottom и height, то предпочтение будет отдано top и height.

Абсолютное позиционирование применяется очень часто совместно с относительным позиционированием в дизайнерских целях, когда необходимо разместить различные элементы относительного друг друга, так же может применяться для создания выпадающих меню, разметки сайта и т.д.



Содержание отчёта:

- 1. Наименование практического занятия.
- 2. Цель занятия.

3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок выполнения задания»

- 4. Список использованных источников.
- 5. Выводы и предложения.

Вопросы для самопроверки:

- 1. Опишите принципы позиционирования
- 2. Опишите фиксированное позиционирование Fixed

Практическое занятие № 9 Контекстные селекторы

Цель занятия: Изучить применение контекстных селекторов.

Исходные материалы и данные.

- 1. Конспект лекций по теме
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

Селектор может состоять из нескольких частей, разделённых пробелом, например:

```
p strong { ... }
ul .hit { ... }
.footer .menu a { ... }
```

```
Такие селекторы называют контекстными или вложенными. Их используют
для того, чтобы применить стили к элементу, только если он вложен в нужный
элемент.
```

Например, селектор .menu а сработает для ссылки а только в том случае, если она расположена внутри элемента с классом .menu.

Читать их проще всего справа налево:

```
/* выбрать все теги strong внутри тегов р */

p strong { ... }

/* выбрать все элементы с классом .hit внутри тегов ul */

ul .hit { ... }

/* выбрать все ссылки внутри элементов с классом .menu, которые лежат внутри элементов

с классом .footer */
```

.footer .menu a { ... }

Таким образом, можно задавать элементам различные стили в зависимости от их контекста. Если ссылка расположена внутри меню, сделать её крупнее, а если внутри основного текста, то задать ей нужный цвет. Соседние селекторы

Контекстные селекторы используются для вложенных друг в друга элементов, а соседние — для расположенных рядом.

Например, теги в списке являются соседними по отношению друг к другу и вложенными в тег .

Соседние селекторы записываются с помощью знака +, например, селектор1 + селектор2. Стили применятся к элементу, подходящему под селектор2, только если сразу перед ним расположен элемент, подходящий под селектор1.

Пример. Есть два элемента списка:

```
class="hit">
class="miss">
```

Селектор .hit + .miss применит стили к элементу с классом miss, так как перед ним есть элемент с классом hit.

Селектор .hit + li, а также селектор li + .miss, или даже li + li тоже применит стили к элементу с классом miss, то есть ко второму элементу списка.

А вот селектор .miss+.hit не сработает, так как элемент с классом miss находится после элемента с классом hit в разметке.

Контекстные и соседние селекторы

Селекторы в CSS можно очень гибко комбинировать. В частности, можно комбинировать контекстные и соседние селекторы.

Например, селектор .player-1 .hit + .miss сработает для тега с классом miss, если сразу перед ним расположен тег с классом hit и оба тега расположены внутри тега с классом player-1.

Задания

1. Создайте произвольный текст с заголовками и несколькими абзацами

2. Создадим и закрасим содержимое тега **div** в красный цвет (используя атрибут **style**):

3. Создадим и закрасим содержимое тега span в зеленый цвет

4. Сгруппируйте теги заголовков и измените их цвет

Документ управляется программными средствами 1С: Колледж Проверь актуальность версии по оригиналу, хранящемуся в 1С: Колледж 5. Создайте атрибут **id** и закрасьте содержимое синим цветом

6. Бывают такие ситуации, когда разные теги имеют один и тот же класс. К примеру, у абзацев и заголовков h2 одновременно задан класс **test**. Вам может потребоваться выбрать **только абзацы** с классом **test**, не затрагивая заголовков h2 с этим же классом.

В этом случае вместо селектора **.test** я должен написать следующее: **p.test** - таким образом я выберу все абзацы с классом **test**, не затронув заголовки. Если же я напишу **h2.test** - то выберу все заголовки h2 с классом **test**, не затронув абзацев.

Создайте в своем документе такой блок

Содержание отчёта:

- 1. Наименование практического занятия.
- 2. Цель занятия.

3. Отчёт о выполнении на каждый этап раздела «Содержание и порядок выполнения задания»

- 4. Список использованных источников.
- 5. Выводы и предложения.

Вопросы для самопроверки:

- 1. Как записываются соседние селекторы?
- 2. Что такое селектор?

Практическое занятие №10 Работа над html страницей по индивидуальному заданию

Цель занятия: Применить полученные знания на практике

Исходные материалы и данные.

- 1. Конспект лекций
- 2. Персональный компьютер
- 3. Программа Notepad++

Содержание и порядок выполнения задания:

Применить полученные знания на практике. Разработать веб-страничку со следующими обязательными элементами:

Заголовки

Текст с оформлением различными шрифтами

Формы (можно для получения отзывов или опрос)

Таблица (возможна для позиционирования)

Ссылки

Списки (различных форматов)

Картинки

Примерная тематика странички:

- 1. Историческое событие
- 2. Магазин
- 3. Рассказ о любимом животном
- 4. Персонажи компьютерных игр
- 5. Бюро путешествий
- 6. Страница о любимом (или популярном) музыканте или группе

Содержание отчёта:

- 1. Наименование практического занятия.
- 2. Цель занятия.
- 3. Вариант

4. Отчёт о выполнении на каждый этап раздела «Содержание и порядок

выполнения задания»

- 5. Список использованных источников.
- 6. Выводы и предложения.

Вопросы для самопроверки:

- 1. Для чего нужен html?
- 2. Для чего нужен css?
- 3. Какие еще языки для гипертекста вы знаете?