Изпитна тема №15

Интернет програмиране

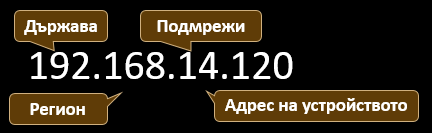
**План-тезис:** Мрежови протоколи. Комплект протоколи TCP/IP. Сокети. HTTP протокол. GET и POST методи. Формуляри. Семантични елементи и най-често използвани тагове в HTML. Стилизиране на уеб страници чрез CSS. HTTP сървъри и сервиране на статични HTML файлове.

* Познава мрежови протоколи, сокети и комплект протоколи TCP/IP

**Мрежов протокол** е набор от правила и стандарти, които позволяват комуникация между мрежовите устройства. Мрежовите протоколи включват механизми за идентифициране на устройствата и осъществяване на връзката помежду им.

Всяко съобщение, файл, или поток от информация се разбива на малки парчета, наречени **пакети**. Когато пакетите се изпращат в Интернет, те обикновено пътуват в мрежата по едно и също трасе. Има ситуации, в които се налага трасето да бъде променено, за да може пакетите да достигнат дестинацията си. Всеки пакет съдържа главна част, наречена header, и секция с данните, които пренася.

Един от най-важните протоколи, използвани в интернет комуникацията, е интернет протоколът (IP). Всички устройства в интернет имат адреси, които се наричат **IP адреси**. IP адресът е уникален за всеки компютър или устройство в края на мрежата.



Тази версия на IP адресиране се нарича IPv4 - последователност от четири трицифрени числа, разделени от точка, като всяко число от нула до 255. През 1995 г. е създадена нова версия на интернет протокола, наречен IPv6. IPv6 използва 128 бита, организиране в осем 16-битови части.

 -> IPv6 адрес

**DNS** е името на домейн начин за адресация на устройства, работеща с текст, вместо IP адрес. Всеки домейн има конфигурирана връзка с IP адреси, като списъка със съответствията формира база от данни, чрез която се осъществява адресация на системи в Интернет. Когато име на домейн бъде въведено в браузъра, се отправя заявка към система, наречена DNS сървър  
(сървър на имена на домейни). Този сървър съдържа списък с имена на домейни и техните съвпадащи IP адреси.

Когато пакетите се предават от едно място на друго, те могат да поемат различни пътища. Когато пристигат на дестинацията си, някои от тях са с грешки от преноса, други са изгубени по трасето и се трябва да бъдат изпратени отново. Затова след пристигане на всички пакети, съобщението трябва да бъде преразгледано и конструирано от пристигналите пакети в правилна последователност - **TCP** прави точно това.

Използва процес, при който разглежда всички пакети в   
съобщение и ги проверява, използвайки информацията в header-а на всеки пакет, който знае колко са, колко големи трябва да бъдат, каква е последователността на изпращане на пакетите. Използвайки тази информация, получателят може да конструира съобщението от пристигналите пакети. Ако се установи, че пакет не съответства на очакваната характеристика, той бива изоставен. TCP трябва да провери дали всички пакети са в правилния ред е без всякакви проблеми. След това удостоверява дали данните и пакетите се обединяват заедно, за да пресъздадат оригиналното съобщение, което е било изпратено от устройството на изпращача.

* Разбира HTTP протокол и прави разлика между методите GET и POST.

HTTP - Hyper Text Transfer Protocol

HTTP е мрежов протокол, от приложния слой на OSI модела, за пренос на информация в компютърни мрежи. HTTP дефинира методи за посочване на желаното действие, което трябва да се извърши върху идентифицирания ресурс.

GET методът служи за извличане на ресурси.

POST служи за съхраняване на ресурси.

HTTP съобщение за заявка се състои от **HTTP ред за заявка**, който съдържа информация за метод на заявка (пр. GET, POST, PUT, DELETE), URL, версия на протокола, HTTP и **HTTP хедър**, който съдържа допълнителни параметри, както и **тяло на HTTP заявката**.

* Познава най-често използваните тагове и семантичните елементи в HTML. Познава техники за стилизиране на уеб страници чрез CSS

**HTML** – HyperText Markup Language

HTML основният маркиращ език за описание и дизайн на уеб страници.

Tаговете предоставят метаинформация за съдържанието на страницата и определят нейната структура. Един HTML документ се състои от много тагове, които се влагат. Таговете са най-малкият елемент в HTML. Атрибутите са свойсвтвата на таговете - размер, цвят и т.н. Елементите са комбинация от отварящ, затварящ таг и атрибути.

**Основни тагове:**

**<!DOCTYPE html>** - Дефинира документът да бъде HTML5

**<html> -** основният елемент на една HTML страница

**</html>**

**<head> -** съдържа мета данни за документа

**<title>HTML Example</title>** - Определя заглавието на документа

**<body>** - Съдържа видимото съдържание на страницата

**<h1>Hello HTML!</h1>** - Дефинира голямо заглавие

**<p>HTML describes formatted text using tags.</p>** - Дефинира параграф

**<header>, <footer>, <nav>, <aside>, <section>** - Семантични тагове за оформление

Html има шест различни HTML заглавия:

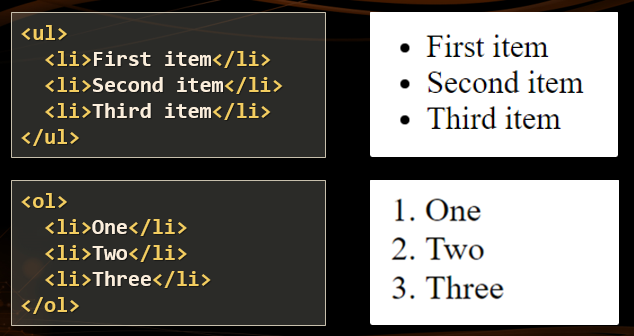
**<h1>** определя най-важното заглавие.

**<h6>** определя най-малко важното заглавие.

**<br>** - Дефинира нов ред

**<ul>** - Създава булети

**<ol>** - Създава номериран списък



**<a> -** Създава хипервръзки

**href="https://google.bg" –** Посочва се адресът на връзката

**<a href="**[**https://google.bg**](https://google.bg)**">Google</a>**

**<a href="welcome.html">Review "welcome.html"</a> -** Връзка към същия уебсайт

**Връщане към началото на страницата:**

**<h1 id="top">Heading</h1>**

**… <!– голямо съдържание -->**

**Go to <a href="#top" target="\_self">top</a>**

**<img> -** Добавя снимки

**<table> -** Дефинира таблица

**<tr>** - Дефинира ред в таблицата

**<th>FirstName</th>** - Дефинира заглавна клетка

**<td>Ivan</td>** - Дефинира клетка от таблица

**<table border="1"> -** Определя дали границите на таблицата трябва да бъде видима

**<th colspan="2">Telephone</th> -** Дефинира колко колони ще обхваща клетката

**<th rowspan="2">Telephone:</th> -** <th rowspan="2">Telephone:</th>

Създаване на форма:

**<form>**

**First name: <input type="text" name="firstname"><br>**

**Last name: <input type="text" name="lastname"><br>**

**<input type="submit" value="Submit">**

**</form>**

**<input type="text" -** Определя поле за въвеждане на текст

**value="First Name" /> -** Текстът по подразбиране се показва в полето за въвеждане

**<input type="text"**

**placeholder="Last Name" />** - Текст, който се визуализира, но се премахва при въвеждане от потребителя

**<input type="password"**

**placeholder="Password" />** - Определя поле за въвеждане на парола(текстът се маскира с ● или \*)

**<input type="radio" name="gender"/>Male<br/> -** Дефинира радио бутон. Всички радио бутони на група ТРЯБВА да споделят едно и също име

**<input type="checkbox"/> I have a bike<br/>** - Дефинира отметка

**<input type="submit" value="Submit"/> -** Дефинира бутон за изпращане и текстът, показан вътре в бутона

**<textarea rows="10" cols="30"> -** Многоредови полета за въвеждане на текст. Атрибутите row и col дефинират колко реда и колони ще обхване текстовата област

**CSS** определя стила на HTML елементите, шрифтове, цветове, полета, размери, позициониране и се декларира в следния формат: **свойсвто:стойност.**

**color**: определя цвета на буквите

**font-family**: трябва да съдържа няколко шрифта. Ако браузърът не поддържа първия, той ще опита следващия

**font-size**: задава размера

**<div>** - Дефинира секция в HTML документа

**<span> -** Използва се за оцветяване на част от текст

**border:** определя типа, дебелината, цвета на границата

**border-radius**: закръгля граничните краища

**background**: задава фона

Външни отстояния се използват за генериране на пространство около елементи.

**margin** свойството задава размера на празното пространство извън границата

**padding** задава размера на празното пространство вътре в границата

**.class** – избира група елементи с посочения клас

**#id** – избира уникален елемент

**tag** – избира всички посочени тагове

**\*** - избира всичко

* Разбира начин на работа на HTTP сървър

В HTTP протокола има [клиент](https://bg.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B8%D0%B5%D0%BD%D1%82_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0))и (обикновено това са Web-[браузърите](https://bg.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D1%8A%D1%80) (или web навигаторите) – т.е. самите [приложения](https://bg.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD_%D1%81%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80), a не физическите [хостове](https://bg.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82) в мрежата) и [сървър](https://bg.wikipedia.org/wiki/%D0%A1%D1%8A%D1%80%D0%B2%D1%8A%D1%80)и (това са [уеб сървърите](https://bg.wikipedia.org/wiki/%D0%A3%D0%B5%D0%B1_%D1%81%D1%8A%D1%80%D0%B2%D1%8A%D1%80) – т.е. самите приложения, а не хостовете). [Браузърът](https://bg.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D1%8A%D1%80) изпраща GET заявка към сървъра, която съдържа необходимите данни за сървъра да я изпълни. [Уеб сървърът](https://bg.wikipedia.org/wiki/%D0%A3%D0%B5%D0%B1_%D1%81%D1%8A%D1%80%D0%B2%D1%8A%D1%80) изпраща ресурса, който е поискан.

HTTP е безсесиен протокол – това означава, че резултата на всяка следваща заявка не зависи от резултата на предишната и така всички клиенти получават равноправно еднакви ресурси. Тази функционалност би създала проблем например в един електронен магазин, където потребителите би трябвало да бъдат идентифицирани с различните си потребителски имена и покупки.

Съществуват различни способи за приложението на сесии в HTTP. Най-надеждният от тях е употребата на бисквитки (cookies). При този способ сървърът залага бисквитките на клиентите със Set-Cookie в хедъра.

Във версиите 0.9 и 1.0 на HTTP, сървърът затваря връзката с клиента след всяка заявка. С версия 1.1. е въведен нов механизъм за поддържане на връзката наречен „keep alive“, при който връзката може да бъде използвана многократно. Този тип постоянна връзка премахва забавянето, получено при установяването на [TCP](https://bg.wikipedia.org/wiki/TCP) връзката след първата http заявка. Това свойство на протокола обикновено трябва изрично да бъде настроено на сървърния [софтуер](https://bg.wikipedia.org/wiki/%D0%A1%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80) (напр. директивата [Keep-Alive](http://httpd.apache.org/docs/2.2/mod/core.html) в [Apache](https://bg.wikipedia.org/wiki/Apache) уеб сървъра).

3FFE:F200:0234:AB00:0123:4567:8901:AB

Всяко съобщение, файл или поток от информация се разбива на малки парчета, наречени паке

Набор от правила и стандарти, които позволяват комуникация между мрежовите устройства