

ТЕОРИЈА: ВЕБ ПРОГРАМИРАЊЕ

353. Који протокол TCP/IP стек протокола подржава захтеве веб прегледача:

1. FTP
2. HTTP
3. TCP
4. IP

ОДГОВОР: 2.

Ово питање није најбоље дефинисано. Одговор је **HTTP**, зато што је то протокол који се користи за веб странице, али 99% браузерa подржава и **FTP** (иако се ретко среће). **TCP/IP** стек протокола развијен да омогући умреженим рачунарима да деле ресурсе путем мреже. **HTTP** (енгл. *HyperText Transfer Protocol*) је мрежни протокол и представља главни и најчешћи метод преноса информација на вебу. **FTP** (енгл. *File transfer protocol* – протокол за пренос датотека) је најчешће коришћен протокол за пренос података између два рачунара на мрежи.

354. Коју је платформу за развој веб апликације развио и Microsoft:

1. HTML
2. JSP
3. PHP
4. ASP.NET

ОДГОВОР: 4.

ASP.NET (енгл. *Active Server Pages .NET*) је веб технологија компаније Microsoft која омогућава програмерима да креирају динамичке веб сајтове, веб апликације и веб сервисе.

355. NET Framework представља:

1. Програмски језик
2. Библиотеку класа
3. Скрипт језик који се извршава на клијент страни
4. Скрипт језик који се извршава на серверској страни

ОДГОВОР: 2.

Microsoft **.NET Framework** је софтверска платформа која може бити инсталирана на рачунарима које покреће Microsoft Windows оперативни систем. Он укључује велики број готових библиотека кодова за уобичајене проблеме у програмирању и виртуелну машину која управља извршавањем програма писаних специјално за .NET Framework.

356. ASP.NET MVC 3.0 долази са новом техником за дефинисање погледа (View Engine):

1. ASP.NET View Engine
2. Salome
3. Razor
4. Default

ОДГОВОР: 3.

ASP.NET MVC 3.0 долази са новом техником за дефинисање погледа тј. Razor – ом. Razor је ASP.NET програмска синтакса која се користи за креирање динамичких веб страница са C# или Visual Basic .NET програмским језиком. **Razor** је развијен у јуну 2010. године и саставни је део Microsoft Visual Studio 2010 од јануара 2011.

357. Дата је ASP.NET MVC апликација у којој је креирана нова мастер страница (master layout page) која се зове `_Layout.WindowsPhone.cshtml`.

Ако желимо да укључимо нову мастер страницу на новом погледу (View), који сегмент кода треба да искористимо:

1. `@Html.ActionLink("_Layout.WindowsPhone.cshtml");`
2. `Layout = "~/views/Shared/_layout.WindowsPhone.cshtml";`
3. `Layout = "Layout.WindowsPhone.cshtml";`
4. `@Html.Partial("_Layout.WindowsPhone.cshtml");`

ОДГОВОР 2:

Layout представља "статички део странице". На пример, ако све странице имају лого фирме или исти мени или било шта што се понавља на страницама, онда се то издвоји у **Layout**, па се после укључи у остале странице. Razor каже да се ти **Layout** фајлови држе у фолдеру `"~/views/Shared"` и да би требало да почну са `"_"` и да се могу учитати релативно. Шта ово значи? Одговор под 2. је тачан, јер

помиње тачну путању до **Layout** фајла. Одговор под 3. би био тачан да пише "_Layout.WindowsPhone.cshtml", али није јер нема "_" на почетку.

358. Од понуђених опција, који симбол се користи за коментаре у ASP.NET MVC Razor синтакси:

1. //
2. /* ... */
3. <!--...-->
4. @* ... *@

ОДГОВОР: 4.

По дефиницији у ASP.NET MVC Razor синтакси, за коментаре се користи. @* ... *. **Први одговор** односи се на линијске коментаре у C#, **други одговор** на блоковске коментаре у C#. **Трећи одговор** односи се на коментаре у HTML – у.

359. Од понуђене синтаксе, која је исправна формулација која дефинише верзију XML документа:

1. <?xml version="1.0" />
2. <xml version="1.0" />
3. <?xml version="1.0"?>
4. <xml version="1.0"?>

ОДГОВОР: 3.

По дефиницији **<?xml version="1.0"?>**. XML документ на почетку (првих 5 карактера тј. **<?xml**) мора да садржи наведене карактере. То је зато што XML анализатор на основу првих пет знакова препознаје начин кодирања знакова у документу. Атрибут **version** углавном има вредност **1.0**. Дакле почетак и крај се обележава са знаком ?

360. SOAP протокол (Simple Object Access protocol) који се користи за размену података између рачунара коришћење веб сервиса, у основи користи један скрипт језик. О којем језику се ради:

1. HTML
2. CSS
3. JavaScript
4. XML

ОДГОВОР: 4.

SOPA користи XML јер су саме поруке XML кодиране. Овде је термин **script** унео забуну – али script језик би могао да се уопштено односи на сваки језик који се не компајлира него може да се интерпретира у неком runtime окружењу.

361. Улога Domain Name Server-а је:

1. превођење имена домена у IP адресу
2. хостовање веб сајта
3. да буде главни чвор у локалној рачунарској мрежи
4. приказ динамичких веб страница

ОДГОВОР: 1.

По дефиницији, **DNS** (енгл. Domain name system) је, у основи, систем који претвара имена рачунара (*hostnames*) у IP адресе.

362. Дефинишите кроз понуђене одговоре улогу Proху сервера:

1. Приступ удаљеном рачунару
2. Побољшава перформансе конекције, филтрира захтеве и прослеђује их на прави сервер
3. Пружа хостинг различитим медијским садржајима (Аудио, Видео)
4. Хостује веб стране

ОДГОВОР: 2.

Прокси сервер је сервер који другим рачунарима омогућава посредан приступ садржајима на Интернету. У преводу енглески појам *проху server* значи сервер посредник. Проху побољшава проток задржавајући у својој cache меморији раније захтеве читача, нпр. ако је неки корисник затражио некакву страницу, проху ће копирати садржај у свој cache. Дакле, има првенствену улогу да кешира, тј. сачува садржај страница које је неко посетио и на тај начин убрза сурфовање, али и смањи проток, јер се исте странице не "довлаче" поново са Интернета...

363. Дефинишите кроз понуђене одговоре улогу сервера датотека – File server (FTP):

1. Омогућавају испоруку веб страна на захтев корисника
2. Размену електронске поште међу корисницима
3. Сигуран и поуздан трансфер података са једног рачунара на други

4. Размену порука у реалном времену

ОДГОВОР: 3.

FTP (енгл. *File transfer protocol*) – протокол за пренос датотека) је најчешће коришћен протокол за пренос података између два рачунара на мрежи.

364. Telnet сервери омогућавају:

1. Размену порука у реалном времену
2. Приступ удаљеном рачунару
3. Размену електронске поште међу корисницима
4. Омогућавају испоруку веб страна на захтев корисника

ОДГОВОР: 2.

По дефиницији, **Телнет** је мрежни протокол унутар групе интернет протокола. Намена овог протокола је успостављање двосмерног осмобитног комуникационог канала између два умрежена рачунара. Најчешће се користи да осигура кориснику једног рачунара сесију за коришћење командне линије на другом рачунару. Сам назив протокола потиче од енглеских речи „телефонска мрежа“ (енгл. *Telephone Network*).

365. Током извршавања апликације у JavaScript језику, на типове променљивих односи се следећа тврдња:

1. Није могуће мењати типове променљивих у току извршавања апликације
2. Типови променљивих се могу мењати током извршавања програма
3. Сваки тип променљивих се обавезно мења током извршавања апликације у одговарајућу један веб тип променљиве
4. JavaScript не подржава типове променљивих

ОДГОВОР 2.

Током извршавања апликације у JavaScript језику, могуће је мењати типове променљивих у току извршавања апликације (1. одговор нетачан, 2 тачан). Сваки тип променљиве не мора нужно да се мења током извршавања апликације. JavaScript подржава типове променљивих.

366. Који типови сервиса могу бити хостовани у конзолној или десктоп апликацији:

1. ASMX
2. RESTful
3. WCF
4. XML

ОДГОВОР: 3.

ASMX не може јер захтева да се извршава на IIS. XML и RESTful не зависе од места хостовања. WCF може да се извршава и на конзолној и на десктоп апликацији.

367. Мајкрософт .NET Framework садржи базне класе које пружају широк спектар могућности. Заокружити све елементе које су укључене у њу:

1. класе корисничког интерфејса
2. класе за приступ подацима и базама
3. класе корисника
4. веб сервер и примере базе података
5. класе за манипулацију XML докумената
6. Скрипт језик који се извршава на клијент страни

ОДГОВОР: 1,2,4 и 5.

368. Од понуђених појмова шта дефинише језик WSDL (Web Services Description) Language:

1. Комуникациони интерфејс за веб сервис
2. Начин имплементације метода веб сервиса
3. Списак метода веб сервиса
4. Комуникациони протокол за веб сервис

ОДГОВОР: 1 и 4.

WSDL описује заправо типове података који се користе, операције (то је уствари комуникациони интерфејс) који излаже сервис напоследку наводи протоколе који се могу користити (да ли је HTTP, TCP, ...)

369. Дата је MVC стандардна рута (default route)

<http://localhost/Customers/Details/5>

која има 3 сегмента.

На основу дате руте вредности ових сегмената допуните реченицу:

Име контролера (Controller Name) је: **Customer** назив методе (Action Method Name) је: **Details** а ID параметра методе је дат са: **5**.

370. Веб обрасци могу да садрже неколико различитих типова компоненти.

Категоришите наведене компоненте веб образаца, и повежите их са датим називима категорија, уносећи тачан Број на линију:

1. HTML контроле	3	TextBox, Label, Button, ListBox, DropDownList, DataGrid
2. Контроле за податке	4	FileSystemWatcher, EventLog, MessageQueue
3. Серверске контроле	2	SqlConnection, SqlCommand, OleDbConnection
4. Системске компоненте	1	Text Area, Table, Image, Submit Button, Reset Button

371. У JavaScript језику, свака веб страница има доле наведене објекте. Са леве стране су дати називи објеката, а са десне стране својства.

Повезати својства објеката са називима објеката, уписујући тачан број на линију:

1. window	3	Садржи својства претходно посећених URL
2. location	4	Садржи својства садржаја текућег документа, као што су назив, боја подлоге, и форме
3. history	2	Садржи својства текуће URL
4. Document	1	Објекат вишег нивоа, садржи својства применљива на цео прозор

372. У JavaScript језику многи објекти имају уграђене функције (методе) које симулирају догађаје. Ти догађаји настају услед акција корисника.

Повежите догађаје са десне стране, са одговарајућим акцијама корисника, уносећи тачан број на линију:

1. Focus	4	Изађе из фокуса елемента форме
2. Submit	3	Учита страницу у прегледач
3. Load	1	Уђе у фокус неког елемента форме
4. Blur	2	Изврши слање форме

373. Садашње верзије IIS сервера изграђене су на модуларној архитектури.

Са леве стране дати су модули, а са десне стране њихове функције.

Повежите називе модула са функцијама које обављају:

1. HTTP модул	3	Модул за обављање послова везаних за евиденцију и дијагностику у захтеву
2. Безбедносни модул	4	Модул за обављање послова у вези са садржајем у захтеву, као што је обрада захтева за статичке фајлове, враћање подразумеване странице када клијент не наведе ресурс у захтеву и наводи садржај директоријума
3. Модули за евиденцију и дијагностику	1	Модул за обављање послова који су специфични за обраду и процесирање захтева, као што је одговарање на информације, враћање HTTP грешака и преусмеравање захтева
4. Модули садржаја	2	Модул за обављање послова везаних за безбедност захтева и процесирања, као што су обављање ауторизације URL-а и филтрирање захтева