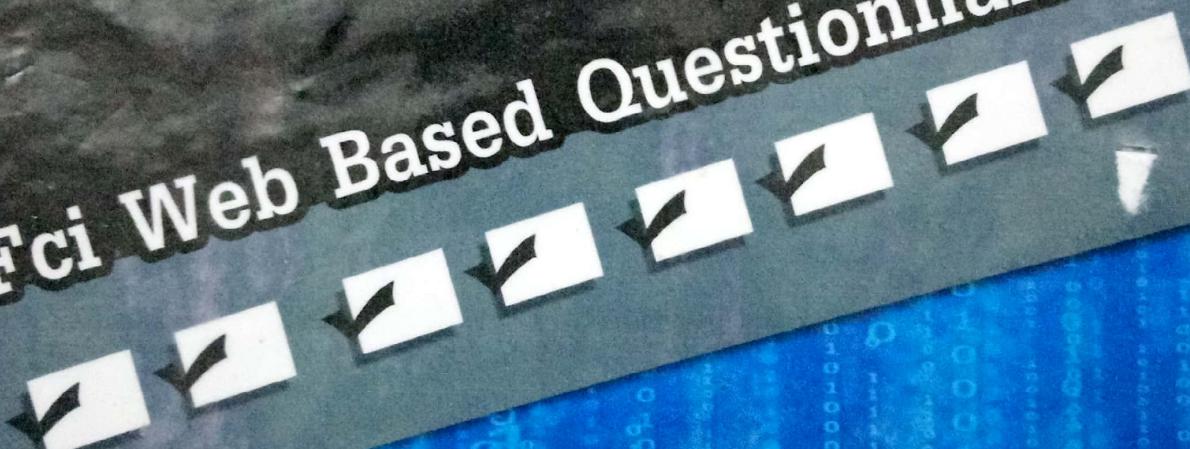




# Fci Web Based Questionnaire



Under supervision of

Dr: Osama El-komy

Dr:Ehab Rushdy

## Team members

1. Asmaa Ibrahim Mahmoud
2. Alaa Mahmoud El-Ghandour
3. Amal AboHashem Rashad
4. Rofayda Ahmed El Sayed
5. Sally Salah Abd El-Kader

## Acknowledge

We just want to say "we give thanks to almighty ALLAH for all things we had achieved in this project".

We must and we move our recent steps in the faculty of computer and informatics of the pause and go back to the years we have spent in university our professors who have given us a lot so great efforts in building tomorrow's generation to send the nation from New ...

Before we offer our deepest thanks and gratitude and appreciation and love to those who carried the holiest message **in life** ...

To all our professors Distinguished...

Great special thanks to our parents "[Dr.Ehab Rushdy](#) & [Dr.Osama El-Komy](#)" who support us with all love and everything we need to achieve our dream.

We can't deny the efforts of our supervisor engineer, we want to thank you very much [Eng/Mahmoud Mahdy](#).

We very happy to send a message of thanks and gratitude to our parents

## ABSTRACT

We have the honor to present this project to our faculty.

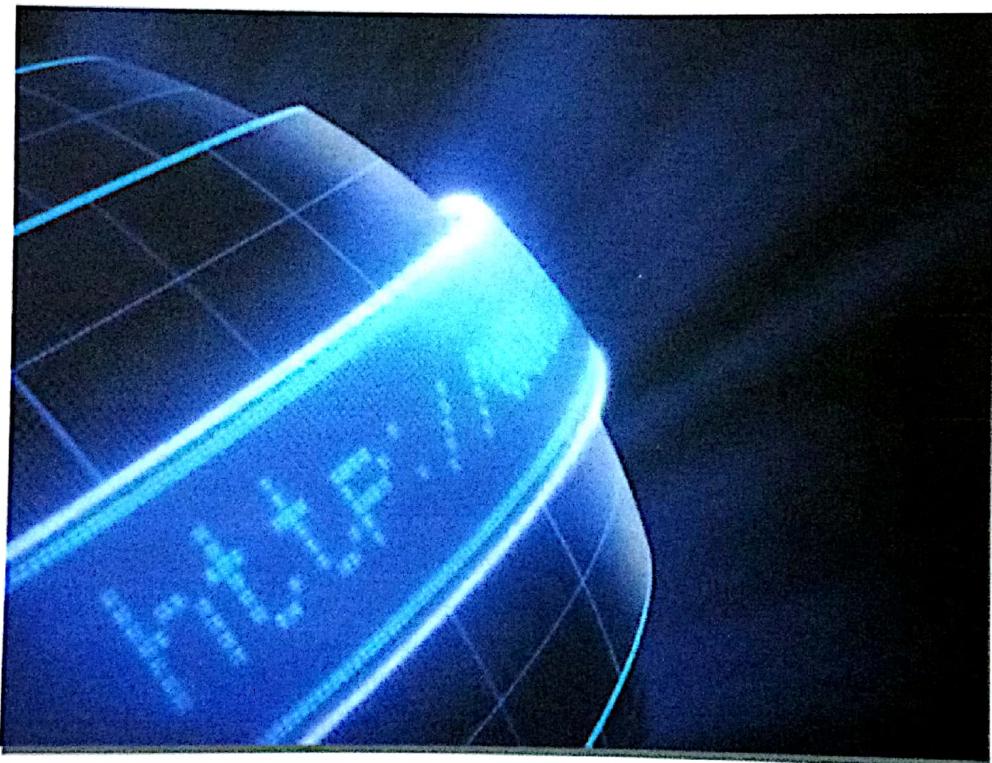
In these days, the website for the faculty is very essentially so we decide to help our professors and students to know new news about faculty and exams.

It helps them to express their opinions of professors and subjects through questionnaires which exist in our web.

It allows professors to upload any materials and present to students links that helps them in subjects and all language programming.

# CHAPTER [1]

( what is the website?)



- ✓ 1.1: What is the website?
- ✓ 1.2: History
- ✓ 1.3 :Uses Of Websites

## 1.1: What is the website?

is a set of related web pages served from a single web domain. A website is hosted on at least one web server, accessible via a network such as the Internet or a private local area network through an Internet address known as a Uniform Resource Locator. All publicly accessible websites collectively constitute the World Wide Web.

A webpage is a document, typically written in plain text interspersed with formatting instructions of Hypertext Markup Language (HTML, XHTML). A webpage may incorporate elements from other websites with suitable markup anchors.

Webpages are accessed and transported with the Hypertext Transfer Protocol (HTTP), which may optionally employ encryption (HTTP Secure, HTTPS) to provide security and privacy for the user of the webpage content. The user's application, often a web browser, renders the page content according to its HTML markup instructions onto a display terminal.

The pages of a website can usually be accessed from a simple Uniform Resource Locator (URL) called the web address. The URLs of the pages organize them into a hierarchy, although hyperlinking between them conveys the reader's perceived site structure and guides the reader's navigation of the site which generally includes a home page with most of the links to the site's web content, and a supplementary about, contact and link page.

Some websites require a subscription to access some or all of their content. Examples of subscription websites include many business sites, parts of news websites, academic journal websites, gaming websites, file-sharing websites, message boards, web-based email, social networking websites, websites providing real-time stock market data, and websites providing various other services (e.g., websites offering storing and/or sharing of images, files and so forth).

## 1.2: History

The World Wide Web ("WWW" or simply the "Web") is a global information medium which users can read and write via computers connected to the Internet. The term is often mistakenly used as a synonym for the Internet itself, but the Web is a service that operates over the Internet, just as e-mail also does. The history of the

Internet dates back significantly further than that of the World Wide Web.

The hypertext portion of the Web in particular has an intricate intellectual history; notable influences and precursors include Vannevar Bush's Memex, IBM's Generalized Markup Language, and Ted Nelson's Project Xanadu.

The concept of a home-based global information system goes at least as far back as "A Logic Named Joe", a 1946 short story by Murray Leinster, in which computer terminals, called "logics," were in every home. Although the computer system in the story is centralized, the story captures some of the feeling of the ubiquitous information explosion driven by the Web.

## **1979–1991: Development of the World Wide Web**

In 1980, Tim Berners-Lee, an independent contractor at the European Organization for Nuclear Research (CERN), Switzerland, built ENQUIRE, as a personal database of people and software models, but also as a way to play with hypertext; each new page of information in ENQUIRE had to be linked to an existing page.

## **1992–1995: Growth of the WWW**

In keeping with its birth at CERN, early adopters of the World Wide Web were primarily university-based scientific departments or physics laboratories such as Fermilab and SLAC. By January 1993 there were fifty Web servers across the world; by October 1993 there were over five hundred.

## **Early browsers**

**The turning point for the World Wide Web was the introduction of the Mosaic web browser in 1993, a graphical browser developed by a team at the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign (UIUC), led by Marc Andreessen.**

## **Web organization**

In May 1994, the first International WWW Conference, organized by Robert Cailliau, was held at CERN; the conference has been held every year since. In April 1993, CERN had agreed that anyone could use the Web protocol and code royalty-free; this was in part a reaction to the perturbation caused by the University of Minnesota's announcement that it would begin charging license fees for its implementation of the Gopher protocol.

## **1996–1998: Commercialization of the WWW**

By 1996 it became obvious to most publicly traded companies that a public Web presence was no longer optional. Though at first people saw mainly the possibilities of free publishing and instant worldwide information, increasing familiarity with two-way communication over the "Web" led to the possibility of direct Web-based commerce (e-commerce) and instantaneous group communications worldwide.

### **1999–2001: "Dot-com" boom and bust**

Low interest rates in 1998–99 facilitated an increase in start-up companies. Although a number of these new entrepreneurs had realistic plans and administrative ability, most of them lacked these characteristics but were able to sell their ideas to investors because of the novelty of the dot-com concept.

### **2002–present: The Web becomes ubiquitous**

In the aftermath of the dot-com bubble, telecommunications companies had a great deal of overcapacity as many Internet business clients went bust. That, plus ongoing investment in local cell infrastructure kept connectivity charges low, and helping to make high-speed Internet connectivity more affordable.

### 1.3 : Uses Of Websites

The World Wide Web is has a multitude of uses. Some of which have not even been discovered yet. Below is a partial list of the kinds of web sites that are available. There are so many sub categories under each main category that I will not try to list them all. I will however try to give an idea of the variety here by listing a few.

- **Business:** ( Agriculture ,Funeral Homes,Design,Wholesale,Travel)
- **Culture:** (Advice,People,Fashion,Museums,Religion)
- **Education:** (K-12,Higher Education, Journals / Research, Classroom Webpages, Individual Student Webpages)
- **Personal:** (Hobbies,Interests,Family,Fan Pages, Pets)
- **Employment:** (Careers, Job Searches, Work Place Issues,Unions,Organizations)
- **Health:** (Nursing,Fitness,Disease,Pharmacy,First Aid)
- **Entertainment:** (Movies,Music,Games,Humor,Radio)
- **Government:** (Elections,Student Government,Law)

- **Search engines:** (Yahoo!Advice, Google, Altavista, Ask Jeeves, Lycos)

## CHAPTER [2]

### (Project scope)



- ✓ 2.1: Database (MySQL)
- ✓ 2.2: C#
- ✓ 2.3: ASP.NET
- ✓ 2.4: JQUERY

✓ 2.5: HTML5

✓ 2.6: CSS3

✓ 2.7: Javascript

### 2.1.1: Database introduction:

A **database** is an organized collection of data, today typically in digital form. The data are typically organized to model relevant aspects of reality in a way that supports processes requiring this information

The term database is correctly applied to the data and their supporting data structures, and not to the database management system (DBMS).

The database data collection with DBMS is called a database system.

The term database system implies that the data is managed to some level of quality (measured in terms of accuracy, availability, usability, and resilience) and this in turn often implies the use of a general-purpose database management system (DBMS). A general-purpose DBMS is typically a complex software system that meets many usage requirements, and the databases that it maintains are often large and complex. The utilization of databases is now so widespread that virtually every technology and product relies on databases and DBMSs

for its development and commercialization, or even may have such software embedded in it. Also, organizations and companies, from small to large, depend heavily on databases for their operations.

Well known DBMSs include Oracle, IBM DB2, Microsoft SQL Server, Microsoft Access, PostgreSQL, MySQL, and SQLite. A database is not generally portable across different DBMS, but different DBMSs can inter-operate to some degree by using standards like SQL and ODBC together to support a single application. A DBMS also needs to provide effective run-time execution to properly support (e.g., in terms of performance, availability, and security) as many end-users as needed.

The term database may be narrowed to specify particular aspects of organized collection of data and may refer to the logical database, to the physical database as data content in computer data storage or to many other database sub-definitions

### 2.1.2: General-purpose DBMS

A DBMS has evolved into a complex software system and its development typically requires thousands of person-years of development effort. Some general-purpose DBMSs, like Oracle, Microsoft SQL Server, and IBM DB2, have been undergoing upgrades for thirty years or more. General-purpose DBMSs aim to satisfy as many applications as possible, which typically makes them even more

complex than special-purpose databases. However, the fact that they can be used "off the shelf", as well as their amortized cost over many applications and instances, makes them an attractive alternative (Vs. one-time development) whenever they meet an application's requirements.

Though attractive in many cases, a general-purpose DBMS is not always the optimal solution: When certain applications are pervasive with many operating instances, each with many users, a general-purpose DBMS may introduce unnecessary overhead and too large "footprint" (too large amount of unnecessary, unutilized software code). Such applications usually justify dedicated development.

### 2.1.3: Database languages

Database languages are dedicated programming languages, tailored and utilized to

- define a database (i.e., its specific data types and the relationships among them),
- manipulate its content (e.g., insert new data occurrences, and update or delete existing ones), and
- query it (i.e., request information: compute and retrieve any information based on its data).

Database languages are data-model-specific, i.e., each language assumes and is based on a certain structure of the data (which typically

differs among different data models). They typically have commands to instruct execution of the desired operations in the database. Each such command is equivalent to a complex expression (program) in a regular programming language, and thus programming in dedicated (database) languages simplifies the task of handling databases considerably. An expression in a database language is automatically transformed (by a compiler or interpreter, as regular programming languages) to a proper computer program that runs while accessing the database and providing the needed results.

#### **2.1.4: Database security**

Database security deals with all various aspects of protecting the database content, its owners, and its users. It ranges from protection from intentional unauthorized database uses to unintentional database accesses by unauthorized entities (e.g., a person or a computer program).

#### **2.1.5: Database design**

Database design is done before building it to meet needs of end-users within a given application/information-system that the database is intended to support. The database design defines the needed data and data structures that such a database comprises.

A design is typically carried out according to the common three architectural levels of a database . First, the conceptual level is designed, which defines the over-all picture/view of the database, and reflects all the real-world elements (entities) the database intends to model, as well as the relationships among them. On top of it the external level, various views of the database, are designed according to (possibly completely different) needs of specific end-user types.

A common way to carry out conceptual level design is to use the Entity-relationship model (ERM) (both the basic one, and with possible enhancement that it has gone over), since it provides a straightforward, intuitive perception of an application's elements and semantics. An alternative approach, which preceded the ERM, is using the Relational model and dependencies (mathematical relationships) among data to normalize the database . Though a large body of research exists for this method it is more complex, less intuitive, and not more effective than the ERM method. Thus normalization is less utilized in practice than the ERM method.

### 2.1.6: Entities and relationships

The most common database design methods are based on the Entity relationship model (ERM, or ER model). This model views the world in a simplistic but very powerful way: It consists of "Entities" and the

"Relationships" among them. Accordingly a database consists of entity and relationship types, each with defined attributes (field types) that model concrete entities and relationships.

Modeling a database in this way typically yields an effective one with desired properties (as in some normal forms; see normalization below). Such models can be translated to any other data model required by any specific DBMS for building an effective database.

### 2.1.7:MySQL database

**MySQL** is the world's most used open source relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases .The SQL phrase stands for Structured Query Language.

#### Platforms and interfaces

MySQL is written in C and C++. Its SQL parser is written in yacc, and a home-brewed lexical analyzer named sql\_lex.cc.

MySQL works on many different system platforms.

Many programming languages with language-specific APIs include libraries for accessing MySQL databases. These include MySQL Connector/Net for integration with Microsoft's Visual Studio (languages such as C# and VB are most commonly used) and the JDBC driver for Java.

## **Management and graphical frontends**

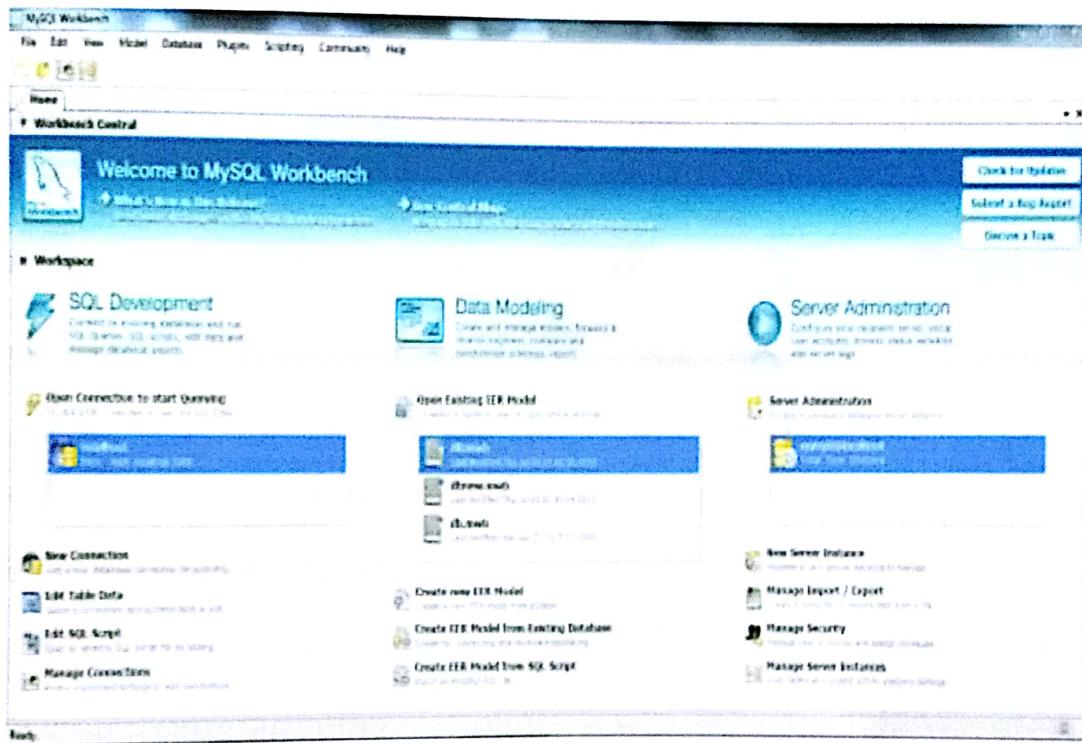
MySQL is primarily an RDBMS and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or download MySQL front-ends from various parties that have developed desktop software and web applications to manage MySQL databases, build database structures, and work with data records.

### **Official**

The official MySQL Workbench is a free integrated environment developed by MySQL AB, that enables users to graphically administer MySQL databases and visually design database structures. MySQL Workbench replaces the previous package of software, MySQL GUI Tools. Similar to other third-party packages, but still considered the authoritative MySQL frontend, MySQL Workbench lets users manage the following:

- Database design & modeling
- SQL development – replacing MySQL Query Browser
- Database administration – replacing MySQL Administrator

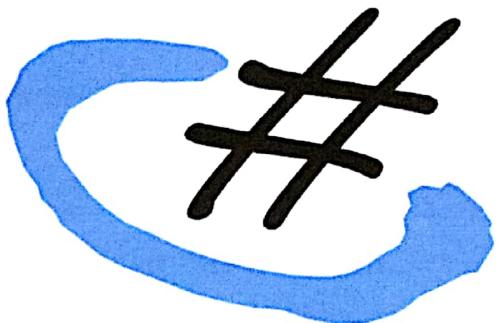
MySQL Workbench is available in two editions, the regular free and open source Community Edition which may be downloaded from the MySQL website, and the proprietary Standard Edition which extends and improves the feature set of the Community Edition.



## Command line

MySQL ships with a suite of command-line tools for tasks such as querying the database, backing up data, inspecting status, performing common tasks such as creating a database, and many more. A variety of third-party command-line tools is also available, including Maatkit, which is written in Perl.

## 2.2: C-sharp (c#)



C#(pronounced see sharp) is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, procedural, generic, object-oriented (class-based), and component-oriented programming disciplines. It was developed by Microsoft within its .NET initiative and later approved as a standard by Ecma (ECMA-334) and ISO (ISO/IEC 23270:2006). C# is one of the programming languages designed for the Common Language Infrastructure.

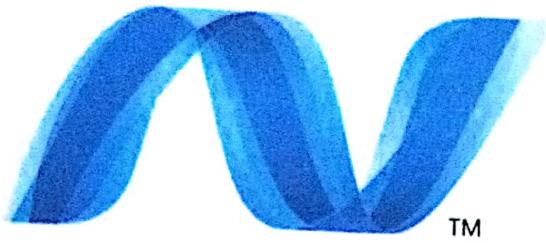
C# is intended to be a simple, modern, general-purpose, object-oriented programming language. Its development team is led by

Anders Hejlsberg. The most recent version is C# 5.0, which was released on August 15, 2012.

- C# supports strongly typed implicit variable declarations with the keyword var, and implicitly typed arrays with the keyword new[] followed by a collection initializer.
- Meta programming via C# attributes is part of the language. Many of these attributes duplicate the functionality of GCC's and VisualC++'s platform-dependent preprocessor directives.
- Like C++, and unlike Java, C# programmers must use the keyword virtual to allow methods to be overridden by subclasses.
- Extension methods in C# allow programmers to use static methods as if they were methods from a class's method table, allowing programmers to add methods to an object that they feel should exist on that object and its derivatives.
- The type dynamic allows for run-time method binding, allowing for JavaScript like method calls and run-time object composition.
- C# has strongly typed and verbose function pointer support via the keyword delegate.

- Like the QT framework's pseudo-C++ signal and slot, C# has semantics specifically surrounding publish-subscribe style events, though C# uses delegates to do so.
- C# offers Java-like synchronized method calls, via the attribute [MethodImpl(MethodImplOptions.Synchronized)], and has support for mutually-exclusive locks via the keyword lock.
- The C# language does not allow for global variables or functions. All methods and members must be declared within classes. Static members of public classes can substitute for global variables and functions.
- Local variables cannot shadow variables of the enclosing block, unlike C and C++.
- A C# namespace provides the same level of code isolation as a Java package or a C++ namespace, with very similar rules and features to a package

## 2.3: ASP.Net



# ASP.NET

ASP.Net is a web development platform, which provides a programming model, a comprehensive software infrastructure and various services required to build up robust web application for PC, as well as mobile devices.

ASP.Net works on top of the HTTP protocol and uses the HTTP commands and policies to set a browser-to-server two-way communication and cooperation.

ASP.Net is a part of Microsoft .Net platform. ASP.Net applications are complied codes, written using the extensible and reusable components or objects present in .Net framework. These codes can use the entire hierarchy of classes in .Net framework.

The ASP.Net application codes could be written in either of the following languages:

- C#
- Visual Basic .Net
- Jscript
- J#

ASP.Net is used to produce interactive, data-driven web applications over the internet. It consists of a large number of controls like text boxes, buttons and labels for assembling, configuring and manipulating code to create HTML pages.

### 2.3.1: ASP.Net Component Model

The ASP.Net component model provides various building blocks of ASP.Net pages. Basically it is an object model, which describes:

- Server side counterparts of almost all HTML elements or tags, like <form> and <input>.
- Server controls, which help in developing complex user-interface for example the Calendar control or the Gridview control.

ASP.Net is a technology, which works on the .Net framework that contains all web-related functionalities. The .Net framework is made

of an object-oriented hierarchy. An ASP.Net web application is made of pages. When a user requests an ASP.Net page, the IIS delegates the processing of the page to the ASP.Net runtime system.

The ASP.Net runtime transforms the .aspx page into an instance of a class, which inherits from the base class Page of the .Net framework. Therefore, each ASP.Net page is an object and all its components i.e., the server-side controls are also objects.

## 2.4: JQuery



jQuery is a lightweight, "write less, do more", JavaScript library.

The purpose of jQuery is to make it much easier to use JavaScript on your website.

jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

The jQuery library contains the following features:

- HTML/DOM manipulation
- CSS manipulation
- HTML event methods
- Effects and animations
- AJAX
- Utilities

#### 2.4.1: Why jQuery?

There are a lots of other JavaScript frameworks out there, but jQuery seems to be the most popular, and also the most extendable.

Many of the biggest companies on the Web use jQuery, such as:

- Google
- Microsoft
- IBM
- Netflix

## 2.4: HTML5?



**HTML5** will be the new standard for HTML.

The previous version of HTML, HTML 4.01, came in 1999. The web has changed a lot since then.

HTML5 is still a work in progress. However, the major browsers support many of the new HTML5 elements and APIs.

### 2.5.1: How Did HTML5 Get Started?

HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

WHATWG was working with web forms and applications, and W3C was working with XHTML 2.0. In 2006, they decided to cooperate and create a new version of HTML.

## Some rules for HTML5 were established:

- New features should be based on HTML, CSS, DOM, and JavaScript
- Reduce the need for external plugins (like Flash)
- Better error handling
- More markup to replace scripting
- HTML5 should be device independent
- The development process should be visible to the public

## 2.5: CSS3?



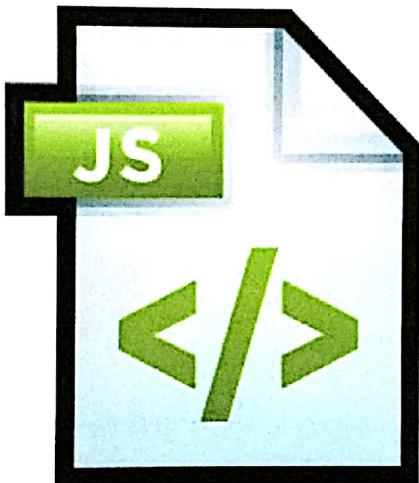
CSS stands for Cascading Style Sheets and is the means by which a Web site's presentation can be defined, styled and modified.

You can use CSS on any Web pages you create on your website in order to bring them to life with slick presentation features.

## Why we use CSS with HTML?

We use HTML with CSS because CSS handles the style of your web page, and lets HTML do the content. HTML has its limitations when it comes to layout, the main benefit in CSS, is that it manages to separate the style from the content on your web pages.

### 2.6: JavaScript



This tutorial has taught you how to add JavaScript to your HTML pages, to make your web site more dynamic and interactive.

You have learned how to create responses to events, validate forms and how to make different scripts run in response to different scenarios.

You have also learned how to create and use objects, and how to use JavaScript's built-in objects.

- JavaScript is very easy to implement. All you need to do is put your code in the HTML document and tell the browser that it is JavaScript.
- JavaScript works on web users' computers — even when they are offline!
- JavaScript allows you to create highly responsive interfaces that improve the user experience and provide dynamic functionality, without having to wait for the server to react and show another page.
- JavaScript can load content into the document if and when the user needs it, without reloading the entire page — this is commonly referred to as Ajax.