**Report**

**Hypertext Markup Language (HTML) is the standard markup language for documents designed to be viewed in a web browser. It can be supported by technologies such as cascading stylesheets and scripting languages ​​such as JavaScript.**

1. **Code Editor:** A code editor or source code editor is software specially designed to assist developers in coding. These are text editors with additional features for managing and editing your code. It can be standalone or part of the IDE.

Some Code Editors Are:

1. Sublime Text
2. ATOM
3. VSC
4. Notepad
5. **CSS**: CSS is the language used to format HTML documents. CSS describes how HTML elements are displayed.
6. Visual Studio Code: Visual Studio Code, commonly referred to as VS Code, is Microsoft's source code editor for Windows, Linux, and macOS.

Features Include:

1. Intelligence
2. Run & Debug
3. Extensions
4. **Bootstrap**: Bootstrap is a free open source CSS framework for responsive front-end web development for mobile devices. It includes HTML, CSS, and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.
5. **Web 2.0 VS Web 3.0:**

| **Web 2.0** | **Web 3.0** |
| --- | --- |
| **Centralized.** Application delivery, cloud services and platform are governed and operated by centralized authorities. | **Decentralized.** Edge computing, peer-to-peer and distributed consensus increasingly become the norm in Web 3.0. |
| **Fiat currency.** Payments and transactions occur with government-issued currency, such as $USD. | **Cryptocurrency.** Transactions can be funded with encrypted digital currencies, such as Bitcoin and Ethereum. |
| **Cookies.** The use of cookies helps to track users and provide personalization. | **NFTs.** Users can get unique tokens that are assigned value or provide some form of perk. |

1. **Blockchain:** A blockchain can be defined as a chain of blocks that contain information. This technology aims to time stamp digital documents so that they cannot be backdated or tampered with. The purpose of the blockchain is to solve the problem of duplicate records without the need for a central server.

For Example:

A Bitcoin Block contains information about the Sender, Receiver, number of bitcoins to be transferred.

1. (1) HTML Attributes: HTML attributes provide additional information about HTML elements.

Example:

1. Href
2. Src
3. Alt
4. Style
5. Lang
6. Title

(2) **HTML Controls:** It is Boolean attribute.

Example:

1. Dropdown Menu
2. Input
3. Range
4. **HTML VS Html5:**

|  |  |
| --- | --- |
| **Html** | **Html5** |
| A hypertext markup language (HTML) is the primary language for developing web pages. | HTML5 is a new version of HTML with new functionalities with markup language with Internet technologies. |
| Language in **HTML** does not have support for video and audio. | **HTML5** supports both video and audio. |
| The HTML browser uses cache memory as temporary storage. | HTML5 has the storage options like: **application cache, SQL database,** and **web storage**. |
| HTML is compatible with almost all browsers because it has been present for a long time, and the browser made modifications to support all the features. | In HTML5, we have many new tags, elements, and some tags that have been **removed/modified**, so only some browsers are fully compatible with **HTML5**. |

1. **CDN:** A CDN (Content Delivery Network) is a group of servers distributed in many locations. These servers store duplicate copies of the data so that the servers can make data requests based on the server closest to each end user. The CDN guarantees high-speed service that is less susceptible to high traffic.

The CDN is typically used to serve stylesheets and JavaScript files (static assets) from libraries such as Bootstrap and jQuery. Using a CDN for these library files is recommended for several reasons:

1. Serving the library's static assets via the CDN offloads the request on the organization's own servers.
2. Most CDNs have servers all over the world, so a CDN server may be geographically closer to you than yours. Geographical distance affects in proportion to delay.
3. **Chrome Developer Tools**: Chrome Developer Tools is a comprehensive developer toolkit built into the Chrome browser. With these tools, you can edit web pages in real time, diagnose problems faster, and build better websites faster. You don't have to be a developer to use the developer tools.
4. **PWA** (Progressive Web App): Progressive Web Applications (PWAs), commonly referred to as Progressive Web Apps, are a type of application software that is delivered over the Internet and built using common web technologies such as HTML, CSS, JavaScript, and Web Assembly.

Advantage:

1. Smaller & faster
2. Offline capabilities

Disadvantage:

1. Limited hardware and software support
2. No download store presense
3. UI (User Interface): The user interface (UI) is the point of human-computer interaction and communication within the device. This includes the appearance of the screen, keyboard, mouse, and desktop.

#.

Note:

1Byte= 8 bit

1024byte=1kb

1024kb=1mb

1024mb=1gb

1. **DNS:** Domain Name System or also known as Name resolver.

#. Google DNS: 8.8.8.8 or 4.2.2.2

1. **HTML:** stands for Hypertext Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between web pages. A markup language is used to define the text document within the tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (e.g. HTML) are human-readable. The language uses tags to define what manipulation has to be done on the text.
   1. **Elements and Tags:** HTML uses predefined tags and elements which tell the browser how to properly display the content. Remember to include closing tags. If omitted, the browser applies the effect of the opening tag until the end of the page.
   2. **HTML page structure:** The basic structure of an HTML page is laid out below. It contains the essential building-block elements (i.e., doctype declaration, HTML, head, title, and body elements) upon which all web pages are created.
   3. **<!DOCTYPE html>:** This is the document type declaration (not technically a tag). It declares a document as being an HTML document. The doctype declaration is not case-sensitive.
   4. **<html>:** This is called the HTML root element. All other elements are contained within it.
   5. **<head>:** The head tag contains the “behind the scenes” elements for a webpage. Elements within the head aren’t visible on the front-end of a webpage. HTML elements used inside the <head> element include:

<style> <title> <base> <noscript> <script> <meta> <link>

* 1. **<body>:** The body tag is used to enclose all the visible content of a webpage. In other words, the body content is what the browser will show on the front-end. An HTML document can be created using any text editor. Save the text file using .html or .htm. Once saved as an HTML document, the file can be opened as a webpage in the browser.

Example: This example illustrates the basic structure of HTML code.

<!DOCTYPE html>

<html>

<head>

<title>Demo Web Page</title>

</head>

<body>

<h1>GeeksforGeeks</h1>

<p>A computer science portal for geeks</p>

</body>

</html>

1.7 **Features of HTML:** It is easy to learn and easy to use. It is platform independent. Images, videos, and audio can be added to a web page. Hypertext can be added to the text. It is a markup language.

* 1. **Why learns HTML?**

It is a simple markup language. Its implementation is easy. It is used to create a website. Helps in developing fundamentals about web programming. Boost professional career.

* 1. **Advantages vs Disadvantages:**

. HTML is used to build websites.

. It is supported by all browsers.

. It can be integrated with other languages like CSS, JavaScript, etc.

. HTML can only create static web pages. For dynamic web pages, other languages must be used.

. A large amount of code must be written to create a simple web page.

. The security feature is not good.

1. **CSS (Cascading Style Sheets)** : is a stylesheet language used to design a webpage to make it attractive. The reason for using this is to simplify the process of making web pages presentable. It allows you to apply styles on web pages. More importantly, it enables you to do this independent of the HTML that makes up each web page.
   1. **There are three types of CSS which are given below:**
2. Inline: Inline CSS contains the CSS property in the body section attached with the element known as inline CSS.
3. Internal or Embedded: The CSS ruleset should be within the HTML file in the head section i.e.the CSS is embedded within the HTML file.
4. external: External CSS contains a separate CSS file that contains only style property with the help of tag attributes.
   1. **Basic Format:** It is the basic structure of HTML webpage, and we use CSS style inside webpage. In a web page, we use internal CSS (i.e., adding CSS code inside <head> tag of HTML code).

<!DOCTYPE html>

<html>

<head>

<! -- Head section of web page -->

<title>

</title>

<! -- Stylesheet of web page -->

<style>

</style> <

/head>

<body>

<! -- Body section of web page -->

</body>

</html>

* 1. **What does CSS mean?**

Tags for formatting a web page were never intended in HTML. HTML was established to define a web page’s content. The addition of tags like <font> and color attributes to HTML created a big problem for web developers. The creation of large websites, where fonts and color information were added to each page, became a time-consuming and costly procedure. CSS was established to address this issue. CSS eliminated the HTML page’s style formatting.

* 1. **How CSS is different from HTML?**

HTML is used to define a structure of a web page whereas CSS is used to style the web pages by using different styling features. HTML consists of tags inside which text is enclosed and CSS consists of selectors and declaration blocks. CSS can be internal or external depending upon the requirement. We cannot use HTML inside a CSS sheet, but we can use CSS inside an HTML document. CSS has comparatively higher backup and support than HTML.

1.5 **Why is CSS used in HTML?**

**. Solves a big problem:** Font, color, background style, element alignments, border, and size tags had to be duplicated on each web page before CSS. This was a lengthy procedure.

**. Saves a lot of time:** Because CSS style definitions are stored in external CSS files, updating only one file can modify the entire website. Provide more attributes: CSS gives more specific features for defining the look and feel of a website than simple HTML.

**. Pages load faster:** CSS doesn’t require you to write HTML tag attributes all the time. A tag’s rule can be written once and applied to all instances of the tag. As a result, CSS uses less code, resulting in speedier download times.

**. Easier Website maintenance:** CSS makes website maintenance much easier. If we need to make a global change to the file, we can simply alter the style, which will update all the elements on the web page.

**. Multiple device compatibility:** We can use CSS with older language versions because it is compatible with them. CSS makes it possible to optimize material for several devices.

## 1.6 Why we learn CSS?

Styling is an essential property for any website. It increases the standards and overall look of the website that makes it easier for the user to interact with it. A website can be made without CSS, as styling is MUST since no user would want to interact with a dull and shabby website. So for knowing Web Development, learning CSS is mandatory.

* **Base for web development:** HTML and CSS is the basic skill that every web developer should know. It is the basic skill that is required for building a website.
* **Makes your website look attractive:** A website that’s dull and plain will not attract the user most probably, so adding some style would surely make your website presentable to the user.
* **Makes the design come live:** A web developer is responsible for making the design given to him as a live product. It is used for styling to develop the design of the website.
* **Increases user experience of the website:** A website with a simple yet beautiful UI would help the users to go through the website easily. It is used to make the user interface better.
* **More career opportunities:** Since CSS is a basic requirement while learning Web Development, therefor there are abundant career opportunities for it. As a freelancer, you can land up to many projects.

1. **Bootstrap:** Bootstrap is a free and open-source tool collection for creating responsive websites and web applications. It is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites. It solves many problems which we had once, one of which is the cross-browser compatibility issue. Nowadays, the websites are perfect for all the browsers (IE, Firefox, and Chrome) and for all sizes of screens (Desktop, Tablets, Phablets, and Phones)
   1. **Why Bootstrap?**

Faster and Easier Web Development. It creates Platform-independent web pages. It creates Responsive Webpages. It is designed to be responsive to mobile devices too. It is Free! Available on [www.getbootstrap.com](http://www.getbootstrap.com)

* 1. **How to use Bootstrap 4 on a webpage:** There are two ways to include Bootstrap on the website.

**.** Include Bootstrap from the CDN link.

**.** Download Bootstrap from getbootstrap.com and use it.

* 1. **Bootstrap 4 from CDN:** This method of installing Bootstrap is easy. It is highly recommended to follow this method.

Go to www.getbootstrap.com and click Getting Started. Scroll down and copy the Bootstrap CDN for CSS, JS, Popper.js, and jQuery links.

* 1. **Bootstrap CSS Library:**

<link rel=” stylesheet” href=”https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css” integrity=”sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T” crossorigin=”anonymous”>

1. **What is Django?**

Django is a Python framework that makes it easier to create web sites using Python.

Django takes care of the difficult stuff so that you can concentrate on building your web applications.

Django emphasizes reusability of components, also referred to as DRY (Don't Repeat Yourself), and comes with ready-to-use features like login system, database connection and CRUD operations (Create Read Update Delete).

* 1. **How does Django Work?**

Django follows the MVT design pattern (Model View Template). **. Model** - The data you want to present, usually data from a database.

**. View** - A request handler that returns the relevant template and content - based on the request from the user.

**. Template** - A text file (like an HTML file) containing the layout of the web page, with logic on how to display the data.

## 1.2 Model:

The model provides data from the database.

In Django, the data is delivered as an Object Relational Mapping (ORM), which is a technique designed to make it easier to work with databases.

The most common way to extract data from a database is SQL. One problem with SQL is that you have to have a pretty good understanding of the database structure to be able to work with it.

Django, with ORM, makes it easier to communicate with the database, without having to write complex SQL statements.

The models are usually located in a file called models.py.

## 1.3 View:

A view is a function or method that takes http requests as arguments, imports the relevant model(s), and finds out what data to send to the template, and returns the final result.

The views are usually located in a file called views.py.

## 1.4 Template:

A template is a file where you describe how the result should be represented.

Templates are often .html files, with HTML code describing the layout of a web page, but it can also be in other file formats to present other results, but we will concentrate on .html files.

Django uses standard HTML to describe the layout, but uses Django tags to add logic:

<h1> My Homepage </h1>

## 1.5 URLs

Django also provide a way to navigate around the different pages in a website.

When a user requests a URL, Django decides which viewit will send it to.

This is done in a file called urls.py.

me is {{firstname}}</p>

The templates of an application is located in a folder named templates.

## 1.6 So, What is Going On?

When you have installed Django and created you first Django web application, and the browser requests the URL, this is basically what happens:

1. Django receives the URL, checks the urls.py file, and calls the view that matches the URL.
2. The view, located in views.py, checks for relevant models.
3. The models are imported from the models.py file.
4. The view then sends the data to a specified template in the template folder.
5. The template contains HTML and Django tags, and with the data it returns finished HTML content back to the browser.

Django can do a lot more than this, but this is basically what you will learn in this tutorial, and are the basic steps in a simple web application made with Django.

1. **JavaScript:**

A JavaScript report links a specific data grid with code that runs in the user's browser. The code can access the underlying data, transform it as desired, and render a custom visualization or representation of that data (for example, a chart, grid, summary statistics, etc.) to the HTML page. Once the new JavaScript report has been added, it is accessible from the **(Reports)** menu on the grid.

* [Create a JavaScript Report](https://www.labkey.org/Documentation/wiki-page.view?name=jsViews#create)
* [GetData API](https://www.labkey.org/Documentation/wiki-page.view?name=jsViews#getDataAPI)
* [Modify the Query Configuration](https://www.labkey.org/Documentation/wiki-page.view?name=jsViews#modify)
* [JavaScript Scope](https://www.labkey.org/Documentation/wiki-page.view?name=jsViews#scope)
* [Example](https://www.labkey.org/Documentation/wiki-page.view?name=jsViews#example)

Note that you must have the [Platform Developer or Trusted Analyst role](https://www.labkey.org/Documentation/wiki-page.view?name=devRoles) to add JavaScript reports.

### 1.1 GetData API

There are two ways to retrieve the actual data you wish to see, which you control using the **JavaScript Options** section of the source editor, circled in red at the bottom of the following screenshot.

Graphical user interface, text, application, email

Description automatically generated

* If **Use GetData API** *is* selected (the default setting), you can pass the data through one or more transforms before retrieving it. When selected, you pass the query config to LABKEY.Query.GetData.getRawData().
  + For details see [LABKEY.Query.GetData API](https://labkey.github.io/labkey-api-js/modules/query.getdata.html).
* If **Use GetData API** *is not* selected, you can still configure columns and filters before passing the query config directly to LABKEY.Query.selectRows()
  + For details see [LABKEY.Query.selectRows()](https://labkey.github.io/labkey-api-js/modules/query.html#selectrows).

### 1.2 Modify the Query Configuration

Before the data is retrieved, the query config can be modified as needed. For example, you can specify filters, columns, sorts, maximum number of rows to return, etc. The example below specifies that only the first 25 rows of results should be returned:

queryConfig.maxRows = 25;

Your code should also add parameters to the query configuration to specify functions to call when selectRows succeeds or fails. For example:

. . .  
queryConfig.success = onSuccess;  
queryConfig.error = onError;  
. . .  
  
function onSuccess(results)  
{  
. . .Render results as HTML to div. . .  
}  
  
function onError(errorInfo)  
{  
 jsDiv.innerHTML = errorInfo.exception;  
}