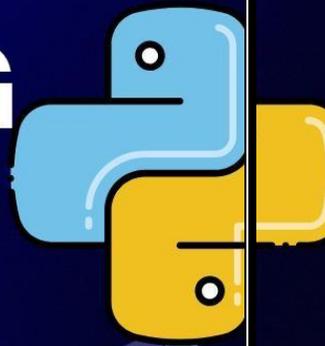
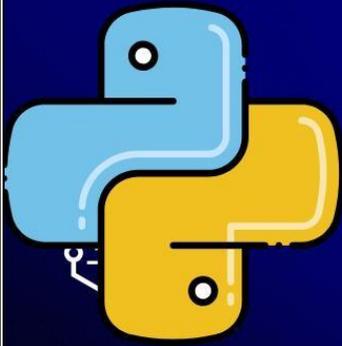


PYTHON PROGRAMMING LANGUAGE



```
// script src= address [#b dq ]  
status.command
```

```
logged=input.[true]  
if ("true") add.string<status> (L,dt)w  
logged=input.false function
```

```
response?  
[lock.command] #www.assort  
#key_in
```



VISIT OUR WEBSITE

www.topitcourses.com

Python is one of the most important and widely used programming languages in the world today. Its popularity comes mainly from its simplicity and readability. Python uses easy-to-understand, English-like syntax, which makes it an ideal choice for beginners as well as professionals. Unlike many other programming languages, Python allows developers to focus more on problem-solving rather than complex syntax, making learning and development faster and more enjoyable.

One of the major reasons for Python's importance is its wide range of applications across different industries. Python is used in web development, software development, data analysis, artificial intelligence, machine learning, automation, scientific computing, and cybersecurity. Popular companies like Google, Netflix, Instagram, and Amazon rely heavily on Python for building scalable and efficient systems. This versatility makes Python a valuable skill for anyone looking to build a strong career in technology.

Python plays a crucial role in data science and artificial intelligence, which are among the fastest-growing fields today. Powerful libraries such as NumPy, Pandas, Matplotlib, Scikit-learn, TensorFlow, and PyTorch enable developers and data scientists to analyze large amounts of data, create visualizations, and build intelligent models. Because of this, Python has become the preferred language for data-driven decision-making and AI-based solutions.

Another important advantage of Python is its ability to automate repetitive tasks. Python is widely used for scripting and automation in IT operations, testing, report generation, file handling, and system administration. By automating routine work, Python helps organizations save time, reduce errors, and improve productivity. This makes Python especially useful in DevOps and enterprise environments.

Python is also known for its strong community support and extensive library ecosystem. With thousands of open-source libraries and frameworks available, developers can quickly build applications without starting from scratch. Additionally, Python is a cross-platform language, meaning the same code can run on Windows, Linux, and macOS with minimal changes. Overall, Python's simplicity, flexibility, wide usage, and career opportunities make it one of the most important programming languages to learn in the modern digital world.

Introduction

Python is an easy programming language and popular programming language too. Python is open-source and can get those libraries from python website python.org. In python, function and datatypes were implemented in C, C++. It can be used for many applications like data cleaning, databases and high-performance computing etc. It holds Data libraries like SciPy, NumPyetc.

Python is being used in Data Science technology development like Neural Networks, Artificial Intelligence, Statistics etc.

Python – Installation

You can understand the Installation of python on windows and Ubuntu in this tutorial. Before that let me give provide you the download link for downloading the Python.

For Linux and Unix Systems, below is the link to download Python:

<https://docs.python.org/3/using/unix.html>

For Windows system, below is the link to download Python:

<https://docs.python.org/3/using/windows.html>

Installation on Ubuntu or Linux Systems:

Python will come pre-installed on most of the Linux distributions. Just you need to give the python3 to start programming in the terminal.

If you don't have the Python, then get the source from

<https://www.python.org/downloads/source/>

follow the below commands.

```
./configure
```

```
make
```

```
make install
```

Installation on windows systems:

Python installers are available to install python on 32-bit and 64-bit versions. Just download the installer and install the software. Once the installation opens the command prompt and give the python command to test.



Pic source: www.python.org

Python – Syntax

In this tutorial, you will learn the python syntax and an example about how to write a basic but popular Hello World print program.

Note: In this tutorial, we are using python version 3.5. Hence all the examples reflect the same results as like we execute in Python 3.5

Example:

```
>>> x="Hello World"
```

```
>>> print(x)
```

```
Hello World
```

```
>>>
```

In above Python terminal, x is a variable where we have assigned a value as hello world in double quotation as it is the string value. Then, we used print function to print the variable x which is in parenthesis. Do remember that we have given parenthesis for variable in print function.

Example:

```
>>> x=2
>>> print(x)
2
>>>
```

Python – Variables and Data types

Variable:

A variable is the location in the memory to store the values. A variable may hold different types of values like numbers, strings etc. In Python, no need to declare a datatype for a variable. It will understand by the value that is assigned to the variable.

The name of variable or a function that we define can be called as Identifier. An Identifier must obey the below rules.

1. Identifiers can have letters, digits, underscores.
2. there is no definite in length.
3. All identifiers must start with letter or underscore. You cannot use digits.
4. Identifier should not be a keyword. (which is a reserved word for python)

Examples of value assignment to a variable:

```
>>>x = 10          # x is Integer
>>>y = 10.1        # y is Float
>>>z = "Hello"     # z is String
>>>x,y = y,x      # assign x value to y and y value to x
>>>a,b,c = 10,20,30 # assign a,b,c values sperated by comma at a time
>>>print(x,y,a,b,c) # printing all the values
```

Data Types:

Python has different types of data types as below.

1. Numbers
2. String
3. List
4. Tuple
5. Dictionary
6. Boolean

Python – Numbers

In Python, Numbers can be defined in three ways.

1. Integer
2. Float
3. Complex

Let us understand how to work with Integer numbers

```
>>>x = 10 # assigning the integer value to variable x
```

```
>>>x
```

```
10
```

we can identify which type of value the variable x is holding

```
>>>x=10
```

```
>>>type(x)
```

```
<class 'int'>
```

Let us understand how to work with Float numbers

```
>>>y=10.1 # assigning the float value to variable y
```

```
>>>y
```

```
10.1
```

Let us identify the type of value the variable y is holding

```
>>>y=10.1
```

```
>>>type(y)
```

Let us understand how to work with complex type

>>>x=4+2y # 4 is the real part and 2y is the imaginary part

We can perform the various calculations with these numbers. Let us see few examples below.

```
>>>5+5
```

```
10
```

```
>
```

```
>
```

```
>
```

```
5
```

```
*
```

```
5
```

```
2
```

```
5
```

```
>>>5-4
```

```
1
```

Python – Strings

In this tutorial, we will work on the Python Strings where we can learn about the manipulation of Strings, using String Operators and string methods and Functions. First, let us understand that how do we declare the strings in python programming language. We can declare and print the strings by placing them in single Quotes ('.'), Double Quotes (".."), and using the print function too. Python Strings are Immutable (An Object with a fixed value).



How to install django:

1. Make sure Python is already installed in

our system `python --version`

2. Install django by using pip

```
pip install django
```

```
pip install django == 1.11.9
```

```
D:\>pip install django
```

```
Collecting django
```

```
Downloading
```

```
https://files.pythonhosted.org/packages/51/1a/6153103322/Django-2.1-py3-none-any.whl (7.3MB) 100% || 7.3MB 47kB/s
```

```
Collecting pytz (from django)
```

```
Downloading https://files.pythonhosted.org/packages/30/4e/53b898779a/pytz-2018.5-py2.py3-none-any.whl (510kB)
```

```
100% || 512kB 596kB/s
```

```
Installing collected packages: pytz, django
```

```
Successfully installed django-2.1 pytz-2018.5
```

```
You are using pip version 9.0.3, however version 18.0 is available
```

```
You should consider upgrading via the 'python -m pip install --upgrade pip'
```

3. To check django version:

```
python -m django --version
```

Django Project vs Django Application:

A Django project is a collection of applications and configurations which forms a full web application.

Eg: Bank Project

A Django Application is responsible to perform a particular task in our entire web application.

Eg: loan app

registration app

polling app etc

django

Project = Several Applications + Configuration Information

Note:

- 1) **The Django applications can be plugged into other projects. ie these are reusable. (Pluggable Django Applications)**
- 2) **Without existing Django project there is no chance of existing Django Application. Before creating any application first we required to create project.**

How to create Django Project:

Once we installed django in our system, we will get 'django-admin' command line tool, which can be used to create our Django project.

```
django-admin startproject firstProject
```

```
D:\>mkdir djangoprojects
```

```
D:\>cd djangoprojects
```

```
D:\djangoprojects>django-admin start-project firstProject
```

The following project structure will be created

```
D:\djangoprojects>
```

```
|
```

```
+---firstProject
```

```
|
```

```
|---manage.py
```

```
|
```

```
+---firstProject
```

```
|---settings.py
```

```
|---urls.py
```

```
|--wsgi.py
```

```
|-- __init__.py
```

__init__.py:

It is a blank python script. Because of this special file name, Django treated this folder as python package.

Note: If any folder contains `__init__.py` file then only that folder is treated as Python package. But this rule is applicable until Python 3.3 Version.

settings.py:

In this file we have to specify all our project settings and configurations like installed applications, middleware configurations, database configurations etc

urls.py:

- **Here we have to store all our url-patterns of our project.**

- **For every view (web page), we have to define separate url-pattern.**
End user can use url-patterns to access our webpages.

wsgi.py:

- **wsgi → Web Server Gateway Interface.**
- **We can use this file while deploying our application in production on online server.**

manage.py:

- **The most commonly used python script is manage.py**
- **It is a command line utility to interact with Django project in various ways like to run development server, run tests, create migrations etc.**

How to Run Django Development Server:

We have to move to the manage.py file location and we have to execute the following command.

```
py manage.py runserver
```

```
D:\djangoprojects\firstProject>py manage.py startserver
```

```
Performing system checks...
```

```
System check identified no issues (0 silenced).
```

```
You have 13 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
```

```
Run 'python manage.py migrate' to apply them. August
```

```
03, 2018 - 15:38:59
```

```
Django version 1.11, using settings 'firstProject.settings'
```

```
Starting development server at http://127.0.0.1:8000/ Quit the server with CTRL-BREAK.
```

```
Now the server started.
```

How to Send First Request:

Open browser and send request:

```
http://127.0.0.1:8000/
```

The following should be response if everything goes fine.

```
-----
```

```
It worked!
```

```
Congratulations on your first Django-powered page.
```

```
Next, start your first app by running python manage.py startapp [app_label].
```

```
You're seeing this message because you have DEBUG = True in your Django settings file and you haven't configured any URLs. Get to work!
```

```
-----
```

Role of Web Server:

- **Web Server provides environment to run our web applications.**
- **Web Server is responsible to receive the request and forward request to the corresponding web component based on url-pattern and to provide response to the end user.**
- **Django framework is responsible to provide development server. Even Django framework provides one inbuilt database sqlite. Special Thanks to Django.**

Note: Once we started Server a special database related file will be generated in our project folder structure.

db.sqlite3

Creation of First Web Application:

Once we creates Django project, we can create any number of applications in that project.

The following is the command to create application.

```
python manage.py startapp firstApp
```

```
D:\djangoprojects\firstProject>python manage.py startapp firstApp
```

The following is the folder structure got created.

```
D:\djangoprojects>
├── firstProject
│   ├── db.sqlite3
│   └── manage.py
├── firstApp
│   ├── admin.py
│   ├── apps.py
│   ├── models.py
│   ├── tests.py
│   ├── views.py
│   └── __init__.py
├── migrations
│   └── __init__.py
└── firstProject
    ├── settings.py
    ├── urls.py
    ├── wsgi.py
    └── __init__.py
```

Note: Observe that Application contains 6 files and project contains 4 files+ one special file: manage.py

1) **__init__.py:**

It is a blank Python script. Because of this special name, Python treated this folder as a package.

2) **admin.py:**

We can register our models in this file. Django will use these models with Django's admin interface.

3) **apps.py:**

In this file we have to specify application's specific configurations.

4) **models.py:**

In this file we have to store application's data models.

5) **tests.py:**

In this file we have to specify test functions to test our code.

6) **views.py:**

In this file we have to save functions that handles requests and return required responses.

7) **Migrations Folder:**

This directory stores database specific information related to models.

Note: The most important commonly used files in every project are views.py and models.py

ORACLE

Data:

- Data is a raw collection of facts about people, places, and things.
- It is unprocessed one. It means, it is not meaningful form.
- Data can be divided into 2 types. They are:
 - i. Structured Data
 - ii. Unstructured Data

Structured Data:

It is made up of with letters, digits, and special symbols.

Example:

25 1-2-123/A Ramu

Unstructured Data:

Multimedia objects come under Unstructured data.

Example:

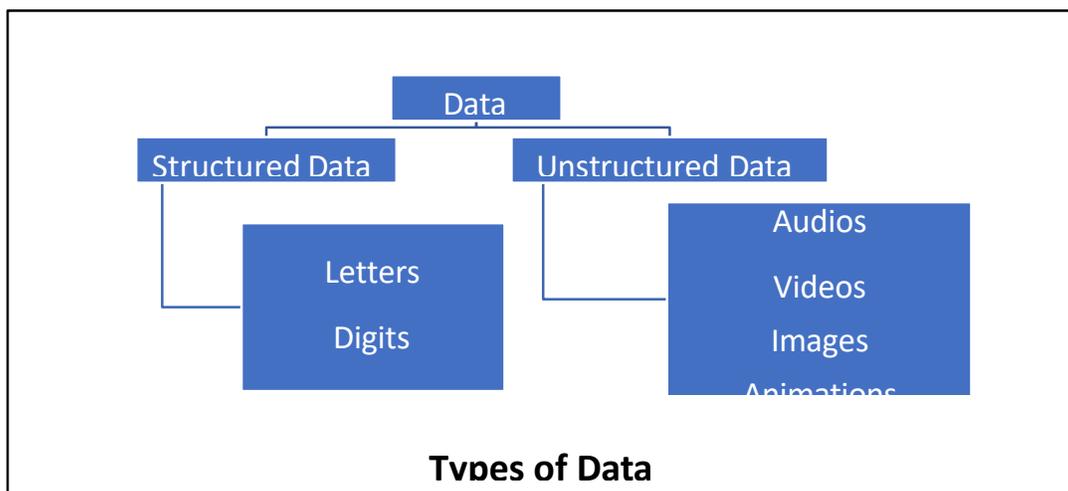
Audios

Videos

Images

Animations

Documents



Information:

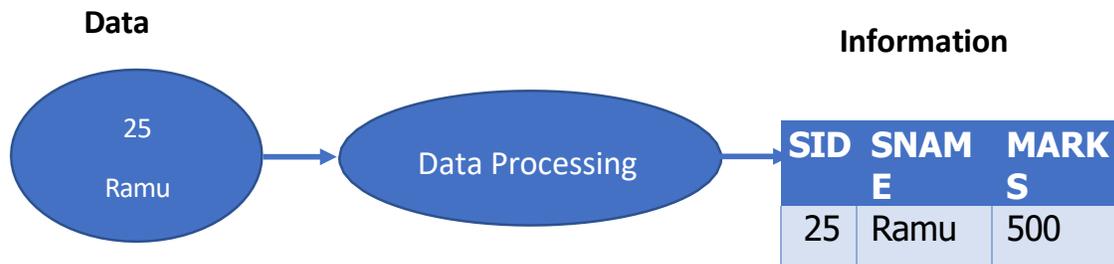
If the data is arranged in meaningful form, then it called “Information”. It is processed one. It is in meaningful form.

Example:**STUDENT**

STUDENT_ID	STUDENT_NA ME	MARKS
25	Ramu	500

Note:

If the data is processed by the process of data processing, it becomes Information.

**Database:**

- Database is a collection of interrelated data.
- It contains records and fields.
- Rows are called records and columns are fields.
- Field holds individual values.
- Record is a collection of field values.
- A row can be also called as record (or) tuple (or) entity instance.
- A column can be also called as field (or) attribute (or) property.
- Database can be divided into two types. They are:
 - i. OLTP
 - ii. OLAP

OLTP:

- OLTP stands for “OnLine Transaction Processing”.
- This database is used for day-to-day operations.
- CRUD operations are performed in day-to-day operations. CRUD operations are:

C =
Create
R =
Read

U =
Update
D = Delete

Example:

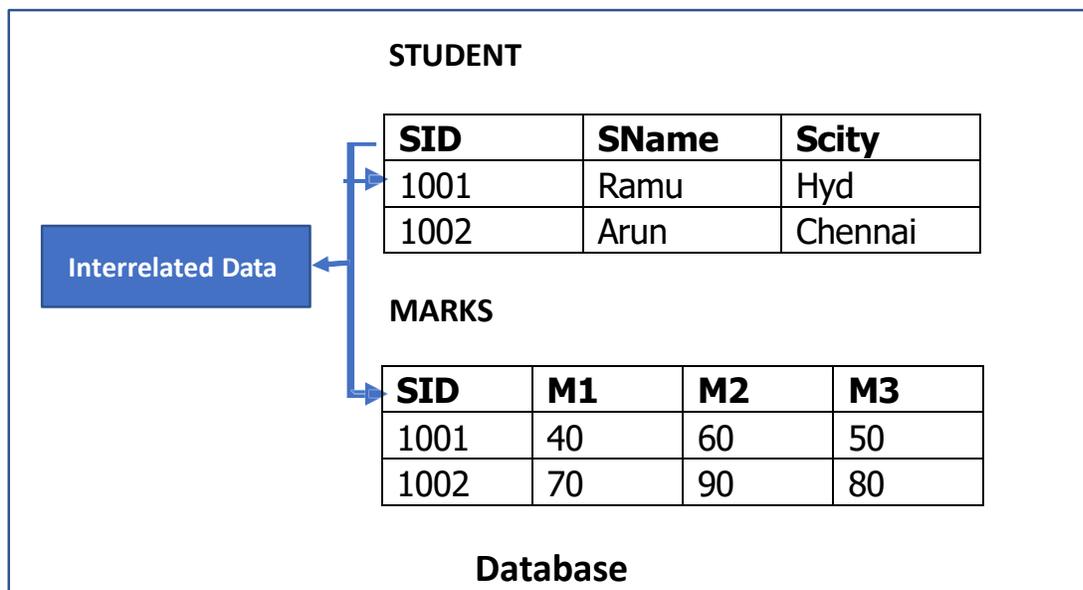
Maintaining day-to-day operations of Bank Customers like withdrawals, deposits, checking balance and fund transfers

OLAP:

- OLAP stands for "OnLine Analytical Processing".
- This database is used for Data Analysis.
- In this, we write complex queries for data analysis.
- It can be also called "Dataware Housing [DWH]" or "Decision Support System [DSS]".

Example:

Retrieving sales of past 5 years



DBMS:

- DBMS stands for DataBase Management System/Software.
- It is a software that is used to create and maintain the database.
- This software allows us to store, manipulate and retrieve the data of database. "Retrieve" means opening existing data. Manipulate means, Inserting, Deleting and Updating [modifying] the data.

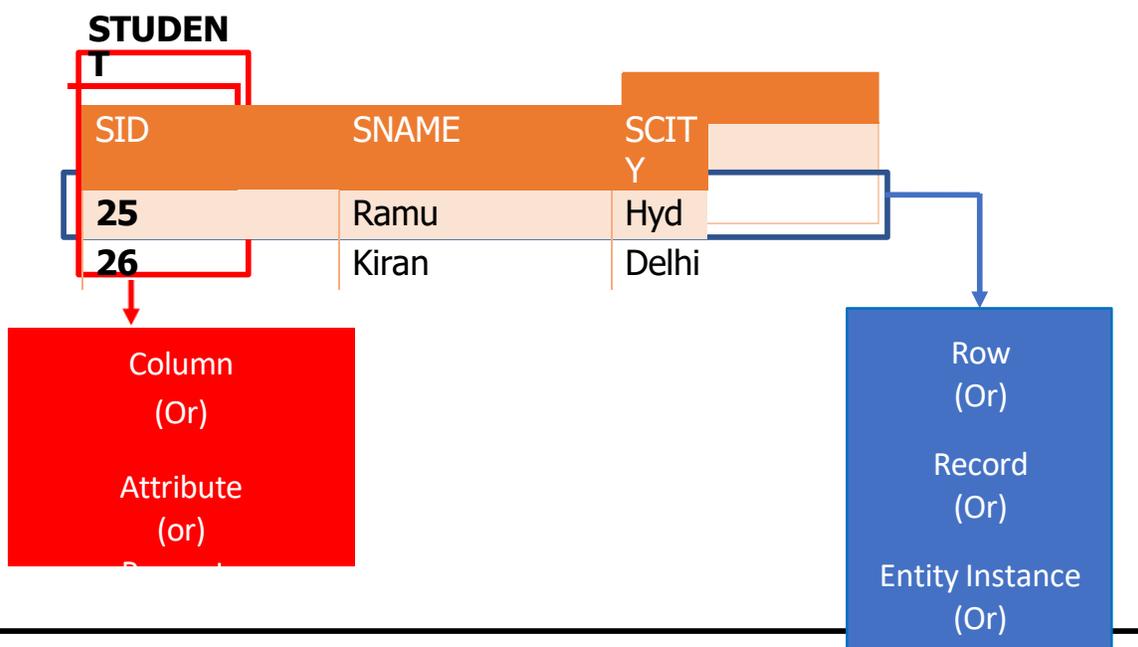
RDBMS:

- RDBMS stands for Relational DataBase Management System / Software.
- In this, we maintain database in the form of tables.

Example: Oracle, SQL Server, MY SQL, PostGre SQL, DB2

Table:

- Table is a collection of rows and columns.
- It can be also called as "Relation" (or) "Entity".
- Rows are called Records (or) Tuples (or) Entity Instances.
- Columns are called Fields (or) Attributes (Or) Fields.
- Field holds individual values.
- Record is a collection of field values.



Metadata:

- Metadata is the data about the data.
- It describes about the data.

Example:

Field
Name
Table
Name
Data Type
Field Size

File Management System [FMS]:

- It is a software which is used to create and maintain the data in flat files (text files).
- It was introduced in 1960s.
- It was the first method to store the business data in computer.
- In this, data and metadata will be stored in many locations. If we want to make any changes to data (or) metadata, we need to change in many places. If we changed in one place forgot to change in another places Data Inconsistency problem will occur.

Disadvantages of FMS:

- Data Redundancy
- Data Dependence
- No Security
- Limited Data Sharing
- Excessive Program Maintenance
- Lengthy Development Times

DataBase Management System [DBMS]:

- DBMS is a software that is create and maintain the database.
- It allows us to store, manipulate & retrieve the data of Database.
- In this, data and metadata will be stored in one location. If we want to make any changes, we need to change in one place only.

Advantages of DBMS:

- Minimal Data Redundancy
- Data Independence
- Security
- Improved Data Sharing
- Improved Data Consistency
- Improved Data Integrity
- Improved Data Accessibility and Responsiveness
- Increased Productivity

ORACLE:

- Oracle is a Relational Database management software.
- It is used to create & maintain the database.
- In this we maintain database in the form of tables.
- It allows us to store, manipulate and retrieve the data of database.
- Second version of "Oracle" introduced in 1979.
- Latest version is Oracle 19C. [c =Cloud].
- It is a product of "ORACLE" company. This company is established in 1977 by "Larry Ellison" with the name "Software Development Laboratories". Later it renamed as "Relation Software Inc." [Incorporation] in 1979. Later it renamed as "ORACLE CORPORATION" in 1983.

Tools of ORACLE:

Oracle provides following tools:

- SQL
- PL/SQL

SQL:

- SQL stands for “Structured Query Language”.
- It is a query language used to write the queries.
- Query is a request that is sent to Database Server.
- Query is used to communicate with the database.
- It is a Non-Procedural Language. We will not write any set of statements.
- It is a Fourth Generation Language [4GL]. In 4GLs, we much concentrate on what to do rather than how to do.
- It is a Unified Language. It is common for many relational databases such as SQL Server, My SQL, Postgre SQL ..etc.
- It provides operators to perform operations like +, -, >, Between and, In Like ..etc.
- It provides built-in functions to perform specific actions.
- It does not provide control structures. Because we cannot write set of statements in SQL. We have no need to control the flow of execution.
- It provides 5 sub languages. Every sub language provides commands to work with the database. They are:
 - i. DDL
 - ii. DML
 - iii. DQL /DRL
 - iv. DCL / ACL
 - v. TCL

Front-end Development

What is front-end?

- Software applications follow tier and layer architecture.
- Applications are built in layers.
- Applications run in tiers.

Tier : run application

Layer : build application

- **Developers work in 3 layers**

- UI Layer
- Business Layer
- Database Layer

- **Application will run in tiers**

- Database Tier
- Application Tier
- Presentation Tier

- Front End development is all about designing the **UI**.

- **In full stack we will learn building UI for 3 types of applications.**

- Web Application
- Distributed Application
- Mobile Native Application

What we need to learn for developing UI?

- HTML
- CSS
- LESS
- SASS
- Bootstrap
- JavaScript
- jQuery
- Angular JS
- React
- Ionic

<u>Why</u>	<u>Description</u>
------------	--------------------

HTML:	It is a presentation language used for preparing presentation for web, distributed and mobile.
--------------	--

CSS	It makes the presentation more attractive, interactive and responsive.
------------	--

LESS	It is a CSS pre-processor. Make your CSS better, simplified, reusable, and compact.
-------------	---

SASS	It is also a CSS pre-processor. Make your CSS better, simplified, reusable, and compact.
-------------	--

JavaScript	To reduce burden on server.
-------------------	-----------------------------

It manages several interactions client-

side. Security Issues, Blocked by

Browsers

JavaScript is language used client-side, server-side, database.

jQuery	It is a JavaScript library.
---------------	-----------------------------

Write less and do more.

It contains set of pre-defined functions. Functions are build with JavaScript.

Angular JS	It is a Front-end Framework.
-------------------	------------------------------

It can build and control the application flow. Google.

React	It is a JavaScript library. It is used with front-end framework like Cordova, Ionic, NativeScript. Facebook. More work is on UI
--------------	---

Ionic Angular.	Framework for building cross platform mobile application with React or
--------------------------	--

Web Application

- What is a network?

- It is a group of computers connecting with each other for sharing of information and resources.
- ARPANET [Advanced Research Projects Agency Network] is the first computer network introduced in 1960's by US-DOD.

- What are types of networks?

- **LAN**
- **MAN**
- **WAN**

- What is internet?

- It is a Wide Area Network connecting computers across the globe.
- International Network.

- What is Web?

- 1990's "Tim Berners Lee" introduced "Web".
- Web is a portion of Internet with restricted access.

- **Who maintains the standards of Web?**

- W3C
- WHATWG
- **Web Terminology**
- Server
- Website
- WebPage

Web Server

Server refers to a computer or software solution where we can host, process and handle requests.

- Web Server resembles both hardware and software.
- It satisfies the client request by sending and receiving data.
- Web Server is responsible for hosting your applications, process and handling requests in Web.
- Web Server is also known as Http Server.

- **The popular web server software**

- Microsoft IIS [Internet Information Services Manager]
- Apache Tomcat
- Node.js
- NGINX
- Lighttpd etc.

- We are going to use the webserver locally our machine, web build application, test, and then we can deploy on live servers.

Windows Web Server

- Windows OS comes with a webserver called "Internet Information Services"
- Locate the Web Server on your windows PC.
 - Open windows control panel
 - Switch to "Large Icon" view
 - Go to "Administrative Tools"
 - Look for "Internet Information Services Manager"
 - Add web server to your windows PC if not available
 - Go to Control Panel
 - Open "Programs and Features"
 - Click on "Turn Windows Features ON or OFF"
 - Select "Internet Information Services"
 - Click OK
- Test Web Server
 - Open any browser
 - Request the following

URL <http://localhost> or
<http://127.0.0.1>

Web Site

- Web is virtual directory on Web Server.
- It provides access to the resources.
- Web [Portion of Internet]
- Site [Location – usually in computing system location is referring to drive and directory]
- Create a new Website on your local server
 - Open IIS [run – inetmgr]
 - Expand local computer [http://localhost]
 - Expand "Sites" folder
 - Right click on "Default Web Site"
 - Select the option "Add Virtual Directory" Alias: Web Site Name [http://localhost/amazon] Physical Path: Resource Location [C:\amazon]

Note: You have to use "\" backslash for physical path and "/" forward slash for virtual path. For URL always use forward slash.

Web Page

- Web page provides a user interface (UI) from where user can interact with the resources in our application.
- Web page is a Hyper Text document that provides interface for interacting with resources.
- The term "Hyper" is derived from a Greek term, which means "beyond".
- Hyper Text document is a document that contains information beyond what is displaying.
- **The web pages are classified into 2 types**
 - Static Page
 - Dynamic Page

Static Page

- Static refers to continuous memory.
- The memory allocated for first request will continue for others.
- Static page contains information that will be same across any number of requests.
- Static page responds with the same content across any number of requests.
- **Static Pages will have extension**
 - .htm
 - .html

Dynamic Page

- Dynamic refers to discreet memory.
- The memory is newly allocated for every request.
- Dynamic page contains information that is customized according to the client request.
- Dynamic pages have extension
 - .aspx, .asp, .php, .jsp

Note: Every website that you designed starts with a default page called "index.html".

Elements in HTML

We need a markup language to design a web page. HTML

What is HTML?

Markup Language
Designing Static &
Dynamic Tim Berners
Lee WHATWG
HTML 4
HTML 5

How HTML Works?

Markup => Bytes => Chars => Tokens [Tags] => Node [Element] => DOM => Layout => Rendering => Painting

<p> => 0011011 => < p > => <p> => p [paragraph] => body->p => Layout [DOM] => generating output => painting output

- HTML is a collection of Elements arranged in a hierarchy called DOM. [Document Object Model]
- HTML presents static DOM.
- JavaScript, jQuery, React, Angular, Vue etc. convert the static DOM into dynamic DOM.
- Tags are used to build element.
- Tags are Tokens
- Tag => Element [Tokenization]

Types of Elements in HTML

1. Normal Elements
2. Void Elements
3. RC Data Elements
4. Raw Text Elements
5. Foreign Elements

Normal Elements:

- They return a presentation directly on callback without using additional attributes

- It can't end presentation.
- You have to end presentation explicitly by using end tag.
 Start Tag
End Tag

Void Elements:

- Void means no-return type.
- These elements will not return any presentation directly on callback.
- They require additional attributes.
- They can return specified content and end implicitly.
- They doesn't require end

tag. Ex.

RC Data Elements:

- These are the container elements used to present plain text content.
 - Rich Content is not allowed.
- <textarea> </textarea>

Raw Text Elements:

- These are the elements presented by using raw text instead of tokens. [tag]
- They don't have tags or tokens. ₑ
©
- They are presented by using "&".

Foreign Elements:

- These elements are not native to HTML, but can be used in HTML.
- They require additional library. Without library you can't use in a page. SVG, Canvas, MathML

FAQ: What is difference between and ?

Ans: There is no difference HTML or Browser can't understand .

It is just used by developers to suppress warnings in editor.

 indicates that it is a self ending element.

HTML Page Extension

- HTML page can be defined with 2 extensions

a) .htm b) .html

FAQ: What is difference between ".htm" & ".html"?

Ans: No difference, both are same.

".htm" is the extension given to file as per standards of Operating System.

Structure of HTML Page

- Every HTML page comprises of 2 sections at high level
 1. Document Declaration
 2. Document Scope

Document Declaration:

- It specifies the version of HTML used for Web Page.
- If document declaration is not defined then it is in HTML 4.
- To indicate that the page is designed in HTML 5, we need document declaration in the first line of page.

```
<!DOCTYPE html>
```

```
<!      comment - not a tag
```

- Document declaration contains meta data, information about HTML document like
 - a) Its version
 - b) Its culture
 - c) License etc.

FAQ: What is the purpose of <!DOCTYPE html>?

Ans: It describes HTML document meta data.
It comprises of version.

Document Scope:

- It specifies the scope of HTML document in page.
- It is defined by using <html> tag.

Synta

x:

```
<!DOCTYPE html>  
<html>  
</html>
```

FAQ: Can a page display multiple document?

Ans: Yes

- It is mandatory to define the culture type used in document. So that browser engine can understand the format of content in page.
- The language culture in page is defined by using "lang" attribute.

```
<html lang="en-in">  
  
</html>
```

lang - is an attribute of <html> tag

Sections in Document Scope

- Every document scope in HTML comprises of 2 sections at high level
 - a) Head <head> </head>
 - b) Body <body> </body>

Head Section:

- It comprises of content that is intended to load into browser memory. So that it can accessed and used by browser or by page whenever required.

- It is defined by using

```
<head>  
  
</head>
```

- Typically head section of web page contains following elements

1. <title>
2. <link>
3. <meta>
4. <script>
5. <style>

Title : It defines the title to display for page in the title bar of browser window. It is also used for bookmarking the page.

Syntax:

```
<!DOCTYPE html>  
<html lang="en-in">  
<head>  
<title> Amazon Shopping | India </title>  
</head>  
</html>
```

Link: It is used to link any external document to web page, usually we link favicon or stylesheet.

Ex: Link Favicon

1. Every favicon must have extension ".ico"
2. Make sure that favicon file size is between (16 x 16) to (32 x 32) pixels.
3. Goto project and add a new file in public folder by name "favicon.ico"
4. Open favicon.ico with any photo editing tool [Paint]
5. Design your icon and save.
6. Go to "index.html" and link

```
<head>
  <link rel="shortcut icon" href="favicon.ico">
</head>
```

rel : It specifies the relation type of external file. [stylesheet or shortcut icon]

href : It specifies the path and name of icon file.

Meta:

- Meta refers to "Meta Data".
- Meta Data means information about your page given to Web Spiders and Web Crawlers used in SEO. [Search Engine Optimization]
- Meta is one of the options used for SEO.
- Meta is also used for responsive

design. Syntax: **Responsive Meta**

Element

```
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
</head>
```

Meta in Head Section:

- Responsive Viewport

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

- Request Handling

```
<meta http-equiv="refresh" content="4">
```

This reloads entire page at regular time interval.

- Character Set

utf-8, utf-16, utf-32

Unicode Transformation Format 8, 16, 32 bit size

English - 8 Bit

Chinese, Jap, French, Korean -

16 bit Arabic, Regional Lang - 32

bit

```
<meta charset="utf-8">
```

- Keywords

They are used to find your page on Web.

```
<meta name="keywords" content="your keywords, words, words">
```

- Description

The summary of your website to display in search results.

```
<meta name="description" content="something about your site">
```

- Author

It specifies the author name if it is a blog.

```
<meta name="author" content="AuthorName">
```

Note: Web comprises of various types of content provides

a) Web Site b) Web Application c) Blog d) Vlog

e) MicroBlog f) Wiki g) Podcasting h) Widget

Web Site: It is a virtual directory. It comprises only static resources.

Web Application: It comprises of both static and dynamic resources.

Blog : The term blog is derived from "Web-Log" [blog]
It is published and maintained by individual users and updated periodically.

Vlog : Video Logs [YouTube Channel]

MicroBlog : Allows multiple users to post they
personal information on to single page.
Ex: Twitter

Wiki : The term wiki means
"Quick" [Hawain] A wiki
allows any user to edit its
content.
EX: Wikipedia, imdb etc..

Podcasting: Allows to broadcast media content on to network devices.
Ex: YouTube, Media Player, iTunes etc..

Widgets : Gadgets for web application or site.

Ex: Currency Converter, EMI Calculator etc..

Summary: Head Section

- title
- link
- meta
- **script :** It is used to embed client or server side script.
- **style :** It is used to embed styles in web page.

Ex:

```
<!DOCTYPE html>
<html lang="en-in">
  <head>
    <title>Amazon Shopping | India</title>
    <link rel="shortcut icon" href="favicon.ico">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta http-equiv="refresh" content="5">
    <meta charset="utf-8">
    <meta name="keywords" content="your keywords">
    <meta name="description" content="something about your site">
  </head>
</html>
```

Note:

- This is a **preview Python Full Stack e-Book** containing **only 30 pages**.
- It is provided to help you understand **how the Python Full Stack e-Book and is structured**.
- The **complete Python Full Stack** includes detailed concepts, real-world examples, and career guidance.
- Buy now from our official website: <https://topitcourses.com/python-full-stack-development/>