

# CRISP Analytics Pvt. Ltd (Lumiq)

## Internship Work

By:- Ankit Kumar Singh

ID:- B418012

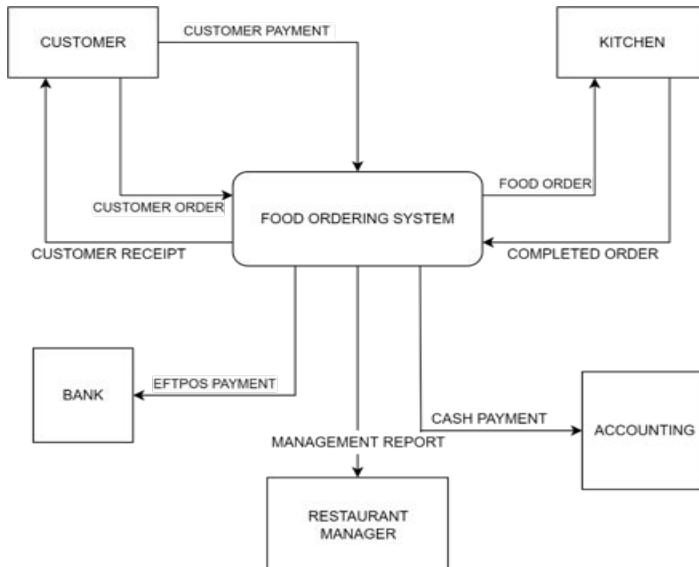
# Overview

The presentation mainly focuses on the work that I have done till now in my internship, the various domains that I have did hands-work like Git, API, Docker, Cloud(AWS), etc. Apart from that it also discusses about the project which I did that is chatbot using Python. Packages like chatterbot is used which uses different machine learning algorithms to produce different types of responses.

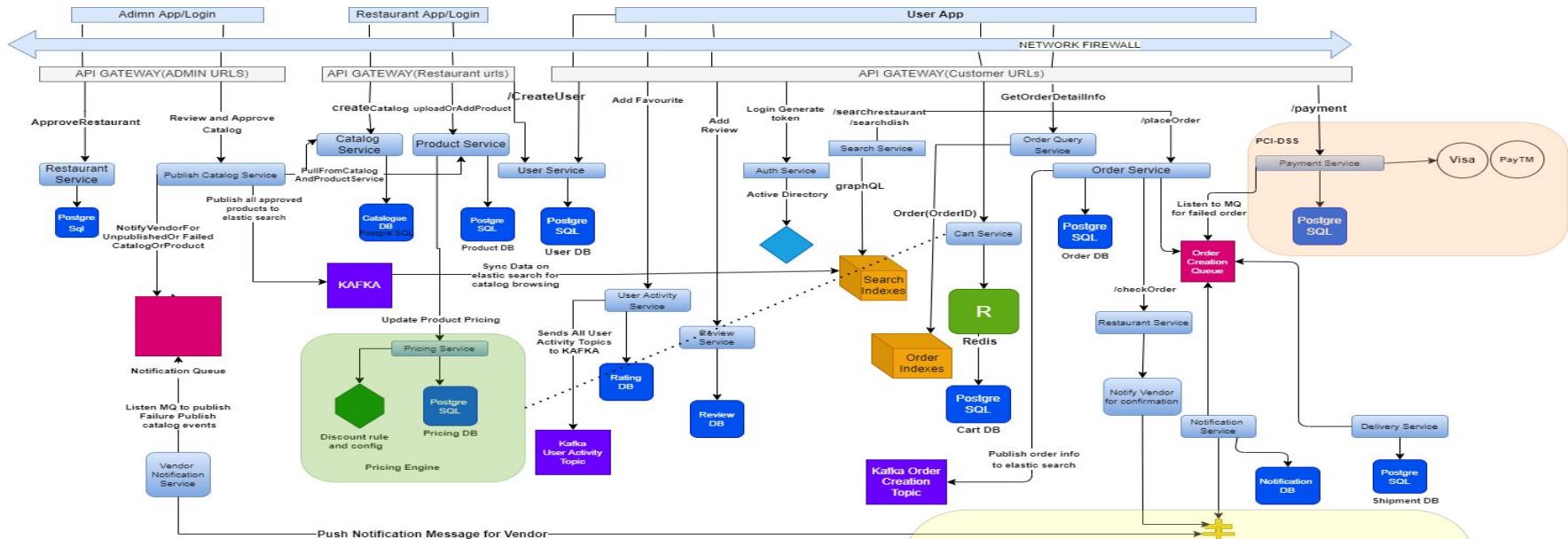
# Communication with diagram, HLD and LLD

This topic mainly deals with an approach to any problem and represent it with the help of Diagrams, Flow Charts and Documentations. It takes the high level view of the problem, understand what and why before proceeding to its how, correlating the problem statement with the desired solutions and finally validating all before proceeding.

Below are some of the HLD and LLD representations of an Online food ordering system called Zomato.



HIGH LEVEL DIAGRAM FOR ZOMATO



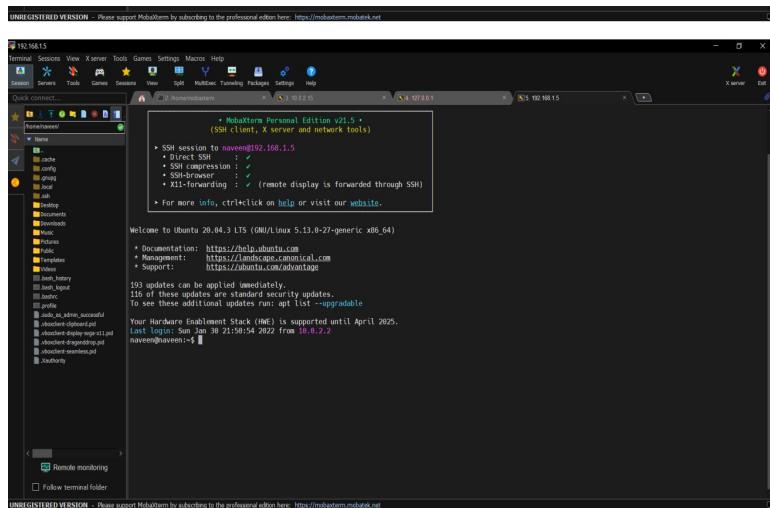
Low level Diagram for Zomato

# Internet

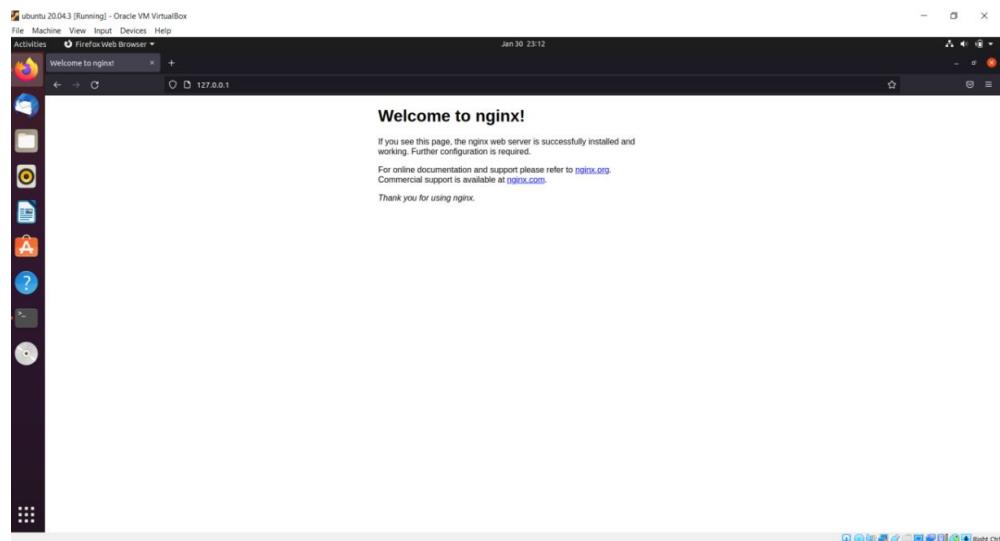
This relates to the topics like Virtualisation, Cloud, SSH, VPN, IP Addresses and Private and Public subnets. Virtualisation refers to the creation of a virtual machine over an existing operating system and hardware.

Cloud refers to the servers that are accessed over the internet and allow to run in those servers. SSH, or Secure shell is a remote administration protocol that allow users to control and modify their servers over the internet.

Some of the works are:-



SSH into the Linux system



Using Nginx for the web server

# Java

It is a general-purpose programming language intended to let programmers write once, run anywhere(WORA). It is used for developing Android Apps, Enterprise Software, is also used in Big Data Analytics and is also used for Server-Side Technologies like Apache, JBoss, Glass Fish, etc.

In Lumiq, Java is used for:-

- > EmPower Platform – it is used for financial service providers who want to take a proactive approach for handling and optimizing their data.
- > Aurum- NLP- driven real-time enterprise customer identification and deduplication solution.

The IDE we used for Java is Eclipse. We were given various questions based on various scenarios and we had to write codes as solution to those scenarios.

We also had to solve all easy, medium and hard level questions which were present on Hackerrank platform.

# HTML and CSS

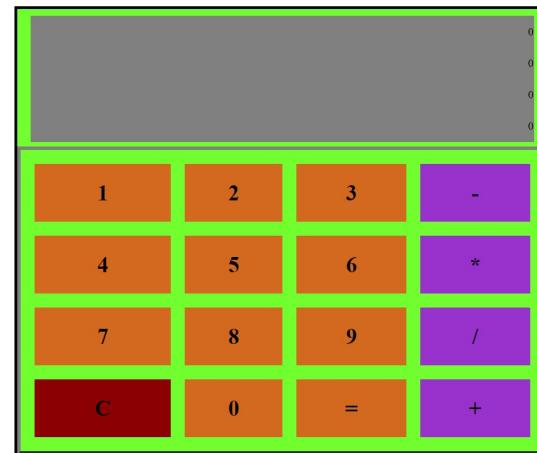
HTML stands for Hypertext Markup Language. It is a computer language used to create web pages. It is basically a text file that contains markup tags, such tags tell the Web Browser how to display a page. It is the skeletal framework of a page and defines the elements on the page.

CSS refers to Cascading Style Sheets. CSS describes how HTML elements are to be displayed on screen, paper, or in other media. It saves a lot of work. It can also control the layout of multiple web pages all at once.

Some of the works were:-



Visiting card



Calculator UI

## PROPOSAL FORM

Thank you for your trust in Canara HSBC Oriental Bank of Commerce Life Insurance Company. This is your proposal form for Canara HSBC Oriental Bank of Commerce Life Insurance iSelect Star Term Plan  
•The payment would be accepted in Indian Rupees ('') only from credit card or debit card or bank account owned by Proposer/Payer.  
•Kindly note that any non-disclosure or misrepresentation or suppression of information has a direct impact on the claim's decision. It may even lead to repudiation of the claim.

For office use only	
Bank Channel Name	Bank Channel Code
Client's Branch/DSP code	
Bank Account No.	
Customer Client No.	
BR Name	
BR Code	ISM Code
Customer Referred by Employee Name	
Referred by Employee No.	
Type of Insurance	Employer Employee <input type="checkbox"/> HUF <input type="checkbox"/> Individual <input type="checkbox"/> MWP <input type="checkbox"/> Partnership Firm <input type="checkbox"/> Salary Deduction <input type="checkbox"/> Key man <input type="checkbox"/>
Relationship with Bank	SB Account <input type="checkbox"/> CA Account <input type="checkbox"/> Deposit <input type="checkbox"/> Advance-Borrower <input type="checkbox"/> credit Card <input type="checkbox"/>
Are You Existing Customer of Canara HSBC Oriental Bank of Commerce Life Insurance Company Limited	
Are you a Staff Member	

Please affix recent Passport size photograph of Proposer and Sign across the photograph

DO NOT STAPLE THE PHOTOGRAPH

### Details of the Life to be Assured

#### 1. Personal Details

##### 1.1 a. Title

Mr.  Mrs.  Miss  Ms.

##### b. Name

First \_\_\_\_\_  
Middle \_\_\_\_\_  
Last \_\_\_\_\_

##### 1.3 a. Title

Mrs.  Miss  Ms.

##### Mother's name

First \_\_\_\_\_  
Middle \_\_\_\_\_  
Last \_\_\_\_\_

##### 1.4 Gender \_\_\_\_\_

##### 1.5 Date of Birth \_\_\_\_\_

##### 1.6 Marital Status \_\_\_\_\_

##### 1.7 a. Current Residential Address/Communication Address \_\_\_\_\_

- b. Permanent Residential Address \_\_\_\_\_  
c. Address Proof Document \_\_\_\_\_  
d. Mobile number with ISD code \_\_\_\_\_  
e. E-mail \_\_\_\_\_

(This email will be taken as registered email ID with us and you may correspond with us using this email ID)

##### 1.8 Educational/Professional Qualifications

##### 1.9 a. Occupation \_\_\_\_\_

##### b. Exact nature of duties/Occupation \_\_\_\_\_

(Specify if you are in money services/lottery/casino/gambling horse Jockey/NGO/Trust/Charity/Real Estate/Jewellery/Scrap Dealer/Diamond Dealer)

c. Name of the Employer/Organisation \_\_\_\_\_ d. Nature of industry of the Employer/Organisation \_\_\_\_\_

Employer/Organisation Address \_\_\_\_\_

Are there any risk associated with your Occupation? Eg. Working with Boiler, Explosives, Chemicals etc. \_\_\_\_\_ (Yes/No)

If yes, please provide details \_\_\_\_\_

##### 1.10 a. Annual Income(Rs) \_\_\_\_\_ b. Age Proof \_\_\_\_\_

c. PAN \_\_\_\_\_ d. Proof of profession of Aadhaar \_\_\_\_\_

CKYC Number(if applicable) \_\_\_\_\_

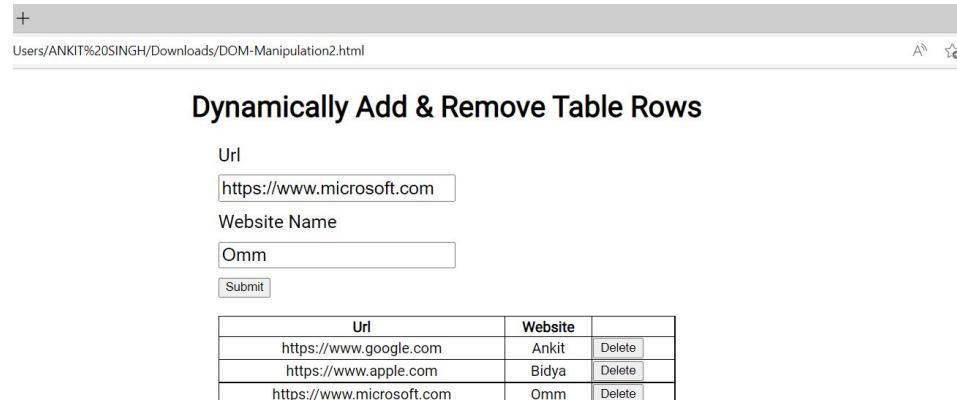
# Javascript

Javascript is a programming language that adds interactivity to the website. It is relatively compact, yet very flexible. Lots of frameworks like angular, reactjs, nodejs are developed over javascript.

Applications:-

- > Client side Validation
- > Manipulating HTML Pages
- > User-Notifications
- > Back-end Data Loading
- > Presentations
- > Server Applications

Some of the works under it were



The screenshot shows a web browser window with the URL `Users/ANKIT%20SINGH/Downloads/DOM-Manipulation2.html`. The page title is "Dynamically Add & Remove Table Rows". It contains a form with fields for "Url" (containing `https://www.microsoft.com`) and "Website Name" (containing "Omm"). A "Submit" button is present. Below the form is a table with three rows:

Url	Website	
<a href="https://www.google.com">https://www.google.com</a>	Ankit	<a href="#">Delete</a>
<a href="https://www.apple.com">https://www.apple.com</a>	Bidya	<a href="#">Delete</a>
<a href="https://www.microsoft.com">https://www.microsoft.com</a>	Omm	<a href="#">Delete</a>

1. DOM Manipulation on user clicks (table row add/delete/ script-add/delete, element add/delete, CSS changes) on clicks of button.



Functioning Calculator

Enter the city

New York

Temp - 294.02

Desc - haze

Using weather API to retrieve data and present the same

# Git

Git is a distributed Version Control System that is used to handle small to very large projects efficiently. Version Control System allows one to revert files back to its previous state. It is used to track changes in the source code, thus allowing multiple developers to come and work together. It runs locally and the files and their history are stored on the computer. Before Git, developers usually submitted their codes to the main server without keeping the copies of their own and the changes made to the source code remained unknown to other developers. But with its help, now it is being feasible to track the changes and updates and work as a team.

Some of the works under it were:-

```
git clone https://github.com/Ankitk12608/testing.git
Cloning into 'testing'...
remote: Enumerating objects: 4, done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
delta=0, reused=0, pack-reused=0
Branch 'main' set up to track 'origin/main'.
Ankitk12608@DESKTOP-0B10504W MINGW64 /d/git/testing
```

```
git add README.md
git add test.txt
git commit -m "First commit"
git push -u origin main
```

```
Ankitk12608@DESKTOP-0B10504W MINGW64 /d/git/testing
```

```
git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file: test.txt
```

```
Ankitk12608@DESKTOP-0B10504W MINGW64 /d/git/testing
```

```
git commit -m "First commit"
git push -u origin main
```

```
Ankitk12608@DESKTOP-0B10504W MINGW64 /d/git/testing
```

```
git status
On branch main
Your branch is up to date with 'origin/main'.
```

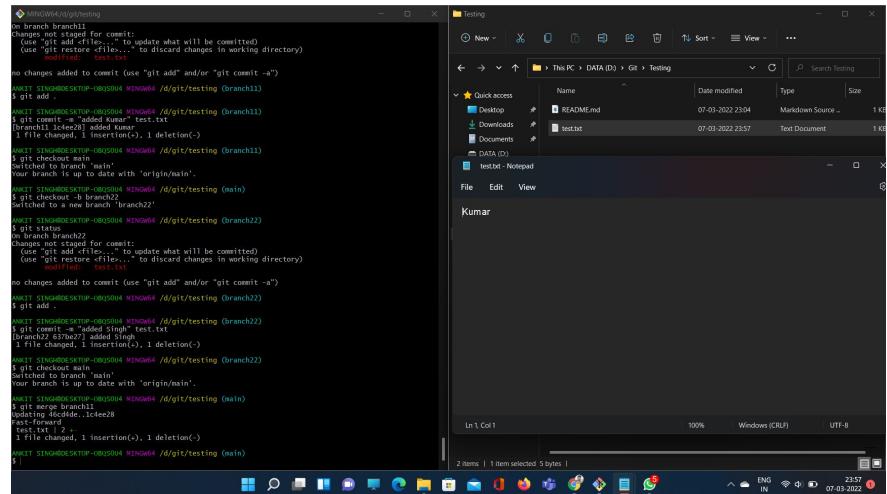
```
Ankitk12608@DESKTOP-0B10504W MINGW64 /d/git/testing
```

```
git add README.md
git add test.txt
git commit -m "First commit"
git push -u origin main
```

```
Ankitk12608@DESKTOP-0B10504W MINGW64 /d/git/testing
```

```
git status
On branch main
Your branch is up to date with 'origin/main'.
```

```
Ankitk12608@DESKTOP-0B10504W MINGW64 /d/git/testing
```



## Creating a Git repository and adding 2 files to it

Create multiple branches to add new features and merge those features into the main branch

```

MINGW64 /d/git/testing
MINGW64 /d/git/testing (issue1)
$ git checkout main
Switched to branch 'main'
branch 'main' set up to track 'origin/main'.
MINGW64 /d/git/testing (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
nothing to commit, working tree clean
changes not staged for commit:
  (use "git add" to update what will be committed)
    (use "git restore --staged <file>..." to discard changes in working directory)
      ...
no changes added to commit (use "git add" and/or "git commit -a")
MINGW64 /d/git/testing (main)
$ git add .
MINGW64 /d/git/testing (main)
$ git commit -m "main commit 2" test.txt
[main 3f7dbd8] main commit 2
 1 file changed, 1 insertion(+), 1 deletion(-)
MINGW64 /d/git/testing (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
MINGW64 /d/git/testing (main)
$ git branch
* main
  issue1
  issue2
MINGW64 /d/git/testing (main)
$ git merge issue1
Automatic merge failed; fix conflicts and then commit the result.
CONFLICT (content): Merge conflict in test.txt
Automatic merge failed; fix conflicts and then commit the result.
MINGW64 /d/git/testing (main MERGEING)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

```

```

MINGW64 /d/git/testing
MINGW64 /d/git/testing (main)
$ git commit -m "merge conflict solved"
[main 021629d] merge conflict solved
 1 file changed, 1 insertion(+)
MINGW64 /d/git/testing (main)
$ git push
remote: error: pathspec 'refs/heads/main' did not match any file(s) known to git
MINGW64 /d/git/testing (main)
$ git commit -m "main commit 2" test.txt
[main 3f7dbd8] main branch commit 2
 1 file changed, 1 insertion(+)
MINGW64 /d/git/testing (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

```

Ankitir2608 / Testing · Public

Just testing

0 stars 0 forks

[Star](#) [Unwatch](#)

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) ...

[main](#) ·

[Ankitir2608 merge conflict solved](#) 7 minutes ago

[View code](#)

README.md

Testing

Just testing

Releases

## Create a scenario of conflicts, solve those conflicts and push them.

```

MINGW64 /d/git/testing
MINGW64 /d/git/testing (main)
$ git merge issue1
Automatic merge failed; fix conflicts and then commit the result.
CONFLICT (content): Merge conflict in test.txt
Automatic merge failed; fix conflicts and then commit the result.
MINGW64 /d/git/testing (main MERGEING)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

```

Testing

Now ·

File Explorer

This PC > DATA (D) > Git > Testing

Name	Date modified	Type	Size
README.md	07-03-2022 23:04	Markdown Source	1 KB
test.txt	07-03-2022 23:14	Text Document	1 KB

## Revert changes and push the previous committee.

# Python

Python is one of the most popular languages, used in diverse range of places like in software systems used by NASA, in Google's search engines and web crawler, and by Industrial Light and Magic in the production of special effects for feature films. It is an interpreted and high-level general-purpose programming language, designed specially for code readability and use of significant indentation.

Its language constructs as well as its object- oriented approach aim to help programmers write clear, logical code for small and large-scale projects. It is dynamically typed and garbage-collected.

In this session we were taken through the advantages and disadvantages of using python , its real-time applications along with the softwares and programmes in lumiq which are based on Python. We were told the DO's and Dont's while writing the code. Then we were given the task of completing all questions of python from hackerrank.

# SQL

SQL refers to Structured Query Language. It is used for storing the data, manipulating it and storing it in databases. It allows users to communicate with Relational Databases and retrieve data from their tables. There are a number of sql languages like MariaDB, MySql, NoSQL, etc.

In the session we were demonstrated various types of sql queries and commands, along with indexing and also shown how to create databases. Then we were given tasks and questions from hackerrank.

The screenshot shows the HackerRank user interface. At the top, there is a navigation bar with the HackerRank logo, links for PREPARE (NEW), CERTIFY, COMPETE, a search bar, a 'Hiring developers?' link, and a 'Log In' button. Below the navigation bar, the user's profile information is displayed: a dark profile picture with the letters 'AK' in white, the name 'Ankit Kumar Singh', the handle '@b418012', and the location 'India'. To the right, under the heading 'Badges', are three yellow hexagonal badges for 'Java', 'Python', and 'SQL', each accompanied by a five-star rating icon.

# Databases and DBMS

Databases are a collection of related data stored in a manner that enables information to be retrieved as needed.

DBMS is Database management system. It is used for creating, maintaining and accessing databases. The part of the program that stores and retrieves data is called database engine.

There are various types of databases like Relational Databases, NoSQL Databases, Cloud Databases, Columnar Databases, etc.

Some of the works were:-

Create a data model for a use-case of your choice. The data model should contain at least 3 entities. Mention the primary key and watermark column (if any). Select a database that best fits your use-case and populate the data in the database according to your data model. 2-3 entries will suffice.

The screenshot shows the MySQL Workbench interface with two windows. The left window displays the 'client' table from the 'hospitaldb' schema. The table has columns ClientID (int PK) and Name (varchar(30)). Data rows include 1056 Ankit and 1058 Piyush. The right window shows the results of a query: 'SELECT \* FROM hospitaldb.client'. The results grid shows the same two rows: Ankit and Piyush. The status bar at the bottom indicates the date as 10-03-2022.

The screenshot shows the MySQL Workbench interface with two windows. The left window displays the 'client' table from the 'client' schema. The table has columns ClientID (int PK) and Name (varchar(30)). Data rows include 1056 Ankit. The right window shows the results of a query: 'SELECT \* FROM hospitaldb.client LIMIT 0, 1000'. The results grid shows the same row: Ankit. The status bar at the bottom indicates the date as 10-03-2022.

**Hospital Table**

The screenshot shows the MySQL Workbench interface with the hospital database selected. The Navigator pane shows the schema structure with tables: hospital, client, services, hospital, hospital, and client. The SQLAdditions pane contains a query: `1 ① SELECT * FROM hospitaldb.hospital;`. The Result Grid pane displays data from the hospital table:

VisitID	Visit Date	ClientID	Visit
12300	2022-05-12	1058	13:45:45
12496	2022-06-04	1058	12:46:23
13276	2022-06-14	1056	14:45:12
13789	2022-06-07	1058	09:12:09
15454	2022-06-24	1260	16:28:53
16318	2022-06-24	1058	16:28:53

The Table: client pane shows the client table structure with columns ClientID (int PK) and Name (varchar(30)). The Action Output pane shows several SELECT queries with their execution times and message counts.

**Service Table**

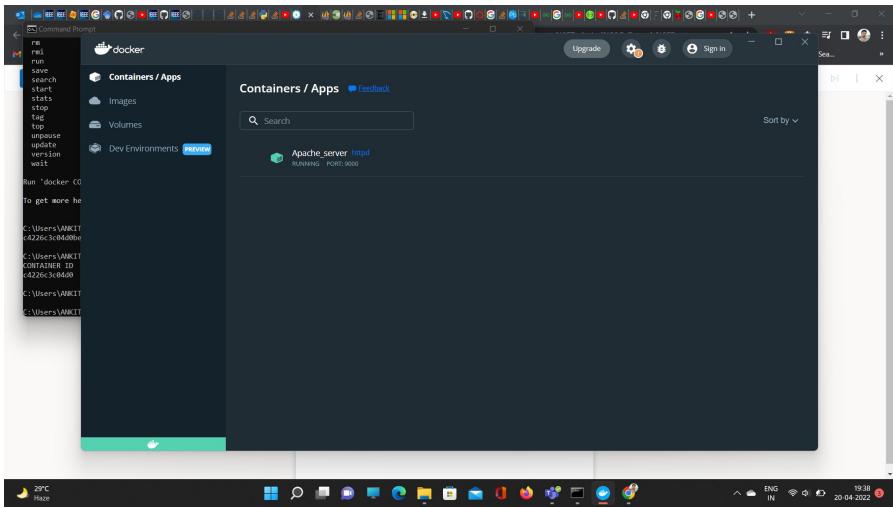
The screenshot shows the MySQL Workbench interface with the services database selected. The Navigator pane shows the schema structure with tables: hospital, client, hospital, hospital, services, and client. The SQLAdditions pane contains a query: `1 ① SELECT * FROM hospitaldb.services;`. The Result Grid pane displays data from the services table:

ServiceID	VisitID	Type
123546	12590	Heart checkup
123547	12590	Lung checkup
123548	12590	Eye checkup
145321	13270	Kidney checkup
147321	13270	Urine test
167543	15420	Skin treatment
167544	15420	Eye treatment

The Table: client pane shows the client table structure with columns ClientID (int PK) and Name (varchar(30)). The Action Output pane shows several SELECT queries with their execution times and message counts.

# Docker

Docker is an open source containerization platform which enables programmers to containerize and package their applications into specific containers which are some standardized executable components that combine application source code with the OS libraries and the required dependencies to run that code in any environment. It is widely used in DevOps teams and it bridges the gap between the developers team and the operations team. They are similar to virtual machines but are very light weight and very efficient.



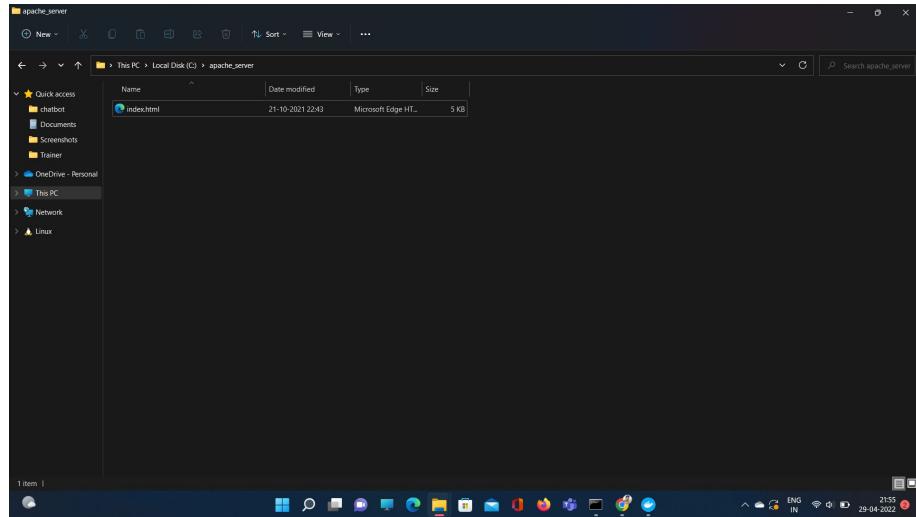
Httpd Docker-container with port exposed 9000,  
always running in background

```
C:\Users\ANKIT SINGH\Desktop>cd docker
C:\Users\ANKIT SINGH\Desktop\docker>docker build -t ankitimage:1.0 .
[+] Building 24.5s (6/6) FINISHED
  => [internal] load build definition from Dockerfile
  => => transferring dockerfile: 199B
  => [internal] load .dockerignore
  => => transferring context: 2B
  => [internal] load metadata for docker.io/library/httpd:latest
  => [1/2] FROM docker.io/library/httpd
  => => [2/2] RUN apt-get update
  => => exporting to image
  => => exporting layers
  => => writing image sha256:acead45f85eb715d94f1cdbdeeaaf55a78fe07072953e208bed9c16dfe060d73
  => => naming to docker.io/library/ankitimage:1.0
  => Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

C:\Users\ANKIT SINGH\Desktop\docker>docker images
REPOSITORY          TAG           IMAGE ID            CREATED             SIZE
ankitimage          1.0           acead45f85eb   18 seconds ago   161MB
pihole/pihole       latest        22872f4d204e  3 weeks ago     302MB
httpd               latest        118babfbff55  4 weeks ago     144MB
alpine/git          latest        b337a04161f7  6 weeks ago     38.2MB

C:\Users\ANKIT SINGH\Desktop\docker>
```

Dockerfile created along with building image on it and pushing it to the local repository



HTML Tutorial

**Hyper Text Markup Language**

Basic & A Very Important Web Technology

HTML Tutorial

HTML - Introduction HTML - Basic Tags HTML - Elements HTML - Formatting HTML - Phrase Tags HTML - Meta Tags HTML - Images HTML - Comments HTML - Tables HTML - Lists HTML - Text Links HTML - Image Links HTML - Email Links HTML - Frames HTML - iFrames HTML - Blocks HTML - Forms HTML - Embedded Multimedia HTML - Header HTML - Style Sheet HTML - Javascript HTML - Layouts HTML - Quiz

By ANKIT | B418012

Created a Docker volume and mounted the volume to the relevant httpd directory to serve the content. Placed all the html files at the mount path. The web server reflects the content in the HTML file.

## API

API stands for application programming interface between two applications that enables them to communicate with each other. The communication cannot happen between the applications unless one of them initiates it. Applications use APIs to send and receive data and content between each other. They work as the middle men, allowing developers to develop various functionalities and interactions between various applications. There are various types of APIs like REST API , HTTPS API, Web-Socket API.

Some of its works were:-

The screenshot shows a Windows desktop environment with several open windows:

- File Explorer:** Shows the project structure for "JDBC\_EXAMPLE".
- IntelliJ IDEA:** The main interface for Java development, displaying code for "client.java" and "client.java ~ client.java 5" (diff view). It includes tabs for "PROBLEMS", "OUTPUT", "TERMINAL", and "DEBUG CONSOLE".
- Terminal:** Shows the command "cd 'c:/Users/VANKIT SINGH/Desktop/my/jdbc\_exampl..." followed by JDBC connection setup and employee data insertion.
- MySQL Command Line Client:** Displays the creation of a database and table, and a query to select all rows from the "employee\_data" table.

At the bottom, the taskbar shows icons for File Explorer, Task View, Connect, Ankit's GitHub profile, Live Share, and various system status indicators like battery level (33% Haze), network, and date/time (12-04-2022).

CRUD (Create , Read , Update and Delete) operation

Maintained Error or Success response like e.g.: { code: 400 , message: 'Bad Request' , error: "Invalid email format", data: null} or success response: {code: 200 , message :'User created successfully' , error: false , data :null}

File Edit Selection View Go Run Terminal Help

client2.java - jdcs\_example - Visual Studio Code

OPEN EDITORS

- src > client2.java >  client >  addition
- client.java 5
- UserProfile.java
- x client2.java 1
- DBUtil.java

JAVA PROJECTS

- dbcs example
- src +
- App
- DBUtil
- UserProfile\_1
- client 5
- client

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE SQL CONSOLE

```
mysql> !cat < /tmp/c_towjpw/gpluhedazv5haf8.argfile > 'client2'
Enter number: 1- Create Table, 2-Add User Profile, to exit- 0
1
Enter number: 1- Create Table, 2-Add User Profile, to exit- 0
2
Enter ID: 1
Enter Name: ankit
Enter Age: 22
Enter Email: ankitkusingh22@gmail.com
Enter Mobile Number: 6370827634
Enter Birthday: 1999-08-26
Enter City: paradeep
Enter State: odisha
Enter Country: india
Enter Address1: bijayachandrapur
Enter Address2: tarinegada
User-Profile Added
Enter ID: 2
Enter Name: aadwac
Enter Gender: female
Enter Email: aadwac@gmail.com
Enter Birthday: 1999-08-23
Enter City: paradeep
Enter State: odisha
Enter Country: india
Enter Address1: bijayachandrapur
Enter Address2: tarinegada
User-Profile Added
Enter number: 1- Create Table, 2-Add User Profile, to exit- 0
```

In 00: Col 16 Spaces: 4 UTF-8 CR/LF Inv JavaSE-11 Premier

15:44 12-04-2022

MySQL 8.0 Command Line Client

```
1 row in set (0.00 sec)

mysql> drop table userprofile_data
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'table userprofile_data' at line 1
Query OK, 0 rows affected (0.43 sec)

mysql> select * from userprofile_data
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | Name | Age | email | Gender | Mobile_Number | Birthday | City | State | Country | Address1 | Address2 |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Ankit | 22 | ankitkusingh22@gmail.com | male | 6370827634 | 1999-08-26 | paradeep | odisha | india | bijayachandrapur | tarinegada |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.03 sec)

mysql> select * from userprofile_data
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | Name | Age | email | Gender | Mobile_Number | Birthday | City | State | Country | Address1 | Address2 |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Ankit | 22 | ankitkusingh22@gmail.com | male | 6370827634 | 1999-08-26 | paradeep | odisha | india | bijayachandrapur | tarinegada |
| 2 | Bidya | 19 | aadwac | female | 5623462364 | 1999-08-23 | paradeep | odisha | india | bijayachandrapur | tarinegada |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from userprofile_data
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | Name | Age | email | Gender | Mobile_Number | Birthday | City | State | Country | Address1 | Address2 |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Ankit | 22 | ankitkusingh22@gmail.com | male | 6370827634 | 1999-08-26 | paradeep | odisha | india | bijayachandrapur | tarinegada |
| 2 | Bidya | 19 | aadwac | female | 5623462364 | 1999-08-23 | paradeep | odisha | india | bijayachandrapur | tarinegada |
| 3 | Mohit | 19 | mohit@gmail.com | Male | 9478260956 | 2022-12-01 | Sambalpur | Odisha | India | Vissat | Barhmeegada |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.10 sec)

mysql>
```

Create User Profile with Name, Age, Email , Gender , Mobile Number, Birthday , City, State, Country, Address1 , Address 2

# Cloud Computing

Cloud computing gives us a way through which one could be able to grasp the applications as tools, on the internet. It enables us to configure , customize and create applications online.

With the help of Cloud computing programmers can acquire the resources in their databases through the internet from any corner of the world till the time they need, without thinking about any maintenance of actual resources or its management. Various Service models like PaaS, Saas and various deployment models like Private Cloud, Public Cloud, Hybrid Cloud and Community Cloud reside in the Cloud platform.

Some works were:-

The screenshot shows the ReqBin API testing tool. At the top, there's a navigation bar with links for Curl, Python, JavaScript, PHP, Java, JSON, XML, Contact, Login, and Sign Up. Below the navigation bar, there's a search bar and a dropdown menu for Authorization. The main area displays a POST request to the URL <https://rj7d6avq.execute-api.us-east-1.amazonaws.com>. The status is 200 OK, with a response time of 284 ms and a size of 0.02 kb. The response content is a JSON object: {"message": "sent from AWS API Gateway"}. At the bottom left, there's a section titled "What is API?" with a detailed explanation of what an Application Programming Interface is. The bottom right shows a taskbar with icons for curl 7.83.0, new user credentials.csv, and a weather widget showing 28°C Haze.

The screenshot shows the AWS Lambda console. The top navigation bar includes links for Services, Search for services, features, blogs, docs, and more, and a user profile for Ankit Singh. A green banner at the top indicates that the function "apiGatewaymessage" has been successfully updated. Below the banner, the function details page for "apiGatewaymessage" is shown. It lists the function ARN as arn:aws:lambda:us-east-1:764367764601:function:apiGatewaymessage and provides a Function URL link. The "Code" tab is selected, showing the Lambda code editor with the following code:

```
function handler(event) {  
    const response = {  
        statusCode: 200,  
        body: JSON.stringify({  
            message: "sent from AWS API Gateway"  
        })  
    };  
  
    return response;  
}
```

The "Test" tab is also visible, showing a test event and a successful execution result. The bottom of the screen shows a taskbar with curl 7.83.0, new user credentials.csv, and a weather widget showing 28°C Haze.

Send a message from aws api gateway to aws lambda eg: message can be "sent from AWS api gateway"

# Project: Chatbot using AI and Python

A chatbot is a computer coded program which is designed in a way to interact like humans. The system uses conversational artificial intelligence (AI) technology to replicate a discussion with a user in a human language via messaging applications, websites, mobile or apps. The earliest chatbot named Eliza, was built in 1966 at MIT's Artificial Intelligence Laboratory by Professor Joseph Weizenbaum to reproduce human conversations.

In this we will be extensively using Python and its packages for building the chatbot. We will be using Chatterbot library package, which generates responses for users. It comes with multiple pre-trained machine learning algorithms to serve a varied amount of responses.

For the UI part we will be using HTML and CSS.

We will deploy the chatbot with the help of Flask package of Python.

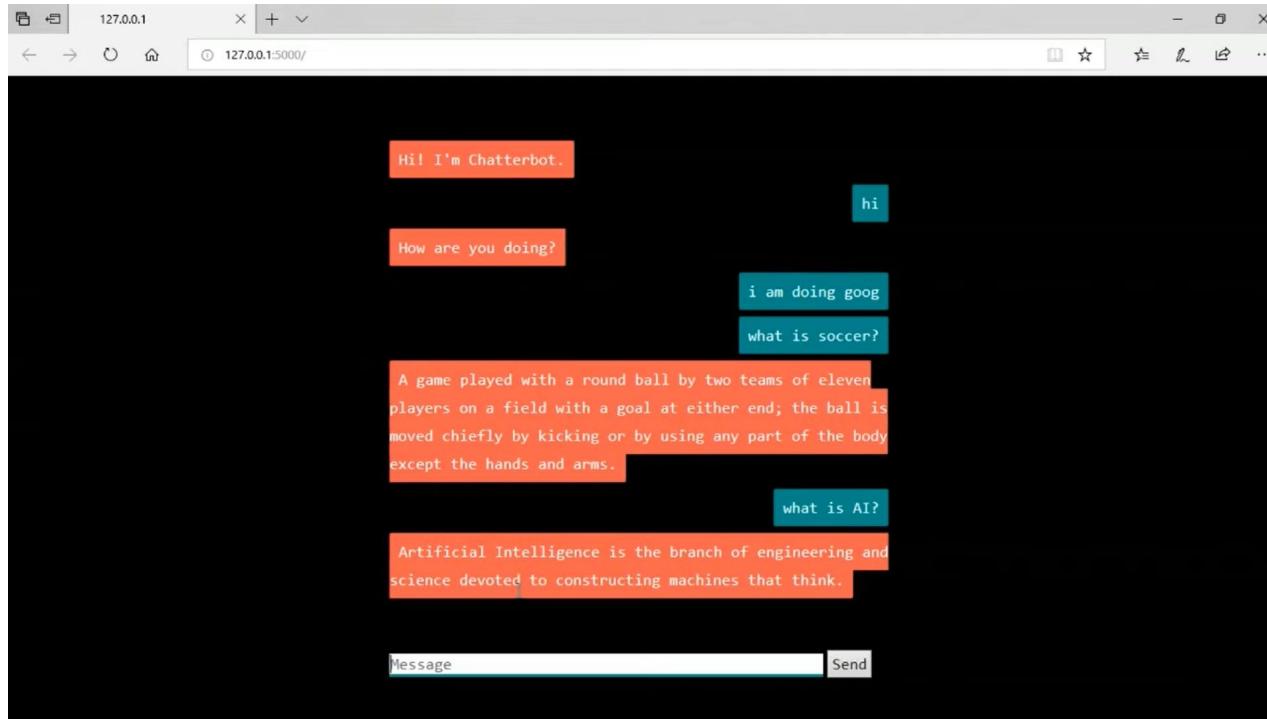
The screenshot shows a Visual Studio Code window titled "app.py - Visual Studio Code". The code editor displays a Python script named "app.py". The script imports Flask, render\_template, and request from flask, and ChatBot and ChatterBotCorpusTrainer from chatterbot. It creates a Flask app, initializes an English bot with a SQL storage adapter, and trains it on the English corpus. It defines routes for the home page and a get endpoint to handle user messages. The code ends with a run command if it's the main module.

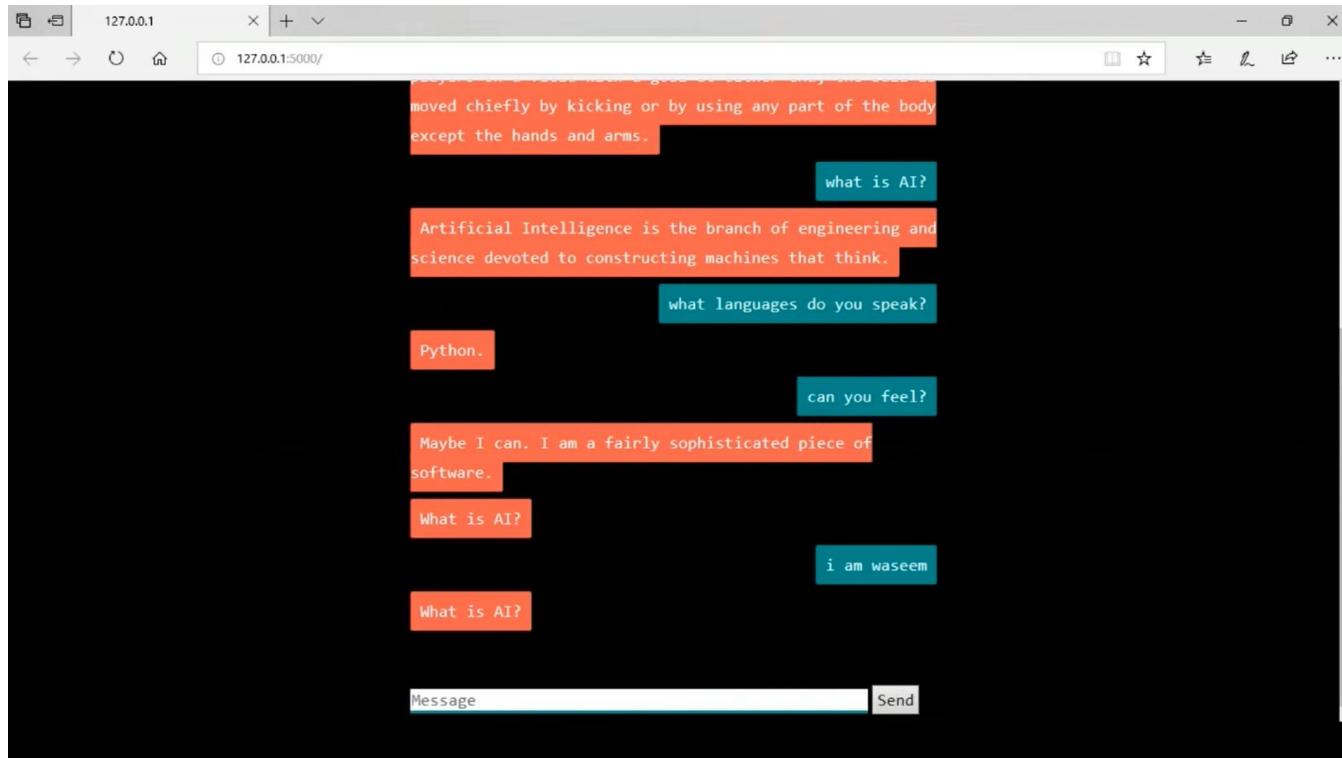
```
C:\> Users > ANKIT SINGH > Desktop > chatbot > app.py
1 # imports
2 from flask import Flask, render_template, request
3 from chatterbot import ChatBot
4 from chatterbot.trainers import ChatterBotCorpusTrainer
5
6 app = Flask(__name__)
7 # create chatbot
8 englishBot = ChatBot(
9     "chatterbot", storage_adapter="chatterbot.storage.SQLStorageAdapter")
10 trainer = ChatterBotCorpusTrainer(englishBot)
11 trainer.train("chatterbot.corpus.english") # train the chatter bot for english
12
13 # define app routes
14
15
16 @app.route("/")
17 def index():
18     return render_template("index.html")
19
20
21 @app.route("/get")
22 # function for the bot response
23 def get_bot_response():
24     userText = request.args.get('msg')
25     return str(englishBot.get_response(userText))
26
27
28 if __name__ == "__main__":
29     app.run()
30
```

The status bar at the bottom shows "Ln 1, Col 1" and "Spaces: 4" along with other system information like battery level, network, and date/time.

## Code: Python

# Screenshots:-





# Conclusion

The project in it has tried to put forward a basic chatbot which has the capability to maintain a casual conversation. The outputs that it provides are quite satisfactory though it gives some inaccurate answers, but that is due to the size of the chatterbot trainer package which is taken as small in our project for better hands-on and feasibility.

Today's tech-enthusiast consumers are always on the hunt for the perfect and most adequate consumer experiences. This becomes an impeccable task to get done as the quest of ever-developing demands is always on the rise. However, a solution, bright enough, seems to satisfy the modern consumers, is a chatbot. With it, companies can precisely give top-quality support and error to a solution whole 24/7, and for a varied variety of customers simultaneously.

The report has also extensively discussed the hand-on works that I have done in my internship in 13 weeks, ranging from Bash, Internet, Java to Databases, Git, API, Cloud and much more.

Thank You