# Block wise Explanation:-

```
from flask import Flask, request, render_template, url_for, redirect
import mysql.connector as mc
app = Flask(__name__,template_folder = "templates")
```

- flask framework for the UI
- mysql.connector for connecting python with MySQL
- Object creation for the Flask application (templates is the name of the folder that consists of all html files

```
@app.route('/')
def home():
    return render_template("home.html")
```

- When the url(localhost:5555 in this case) is opened, home function gets invoked
- It renders home.html file

```
@app.route('/assessment', methods=['GET','POST'])
def assessment():
    if request.method == 'GET':
        return render_template("index.html") # to display form
    elif request.method == 'POST': # after submit
        email = request.form.get("email") # to get email from form
        like = request.form.get('like') # to get boolean value from form
        if like == "yes":
            like = 1
        else:
            like = 0
        fav_num = request.form.get("fav") # to get favourite number from form
        con = mc.connect(host='localhost', user='root', passwd='''put your mysql password''')
        cur = con.cursor()
        cur.execute('create database if not exists OrbDoc')
        cur.execute('use OrbDoc')
        cur.execute('create table if not exists Customer_Discovery(Email varchar(100) not null,\
Boolean_Value tinyint(1) not null, Favourite_Number int not null, primary key(email))')
        data =(email,like,fav_num)
        query ='insert into Customer_Discovery values (%s,%s,%s)'
        try: # execution of the query will pose in error, if email already exists
            cur.execute(query,data) # as email is the primary key
            con.commit()
            return render_template("result.html",alert_msg="Thank You, Your preference has been stored successfully!")
        except:
            return render_template("result.html",alert_msg="Thank You, we already have your preference stored!")
```

- When the url(localhost:5555/assessment in this case) is opened, assessment function gets invoked
- It renders index.html file , which consists of the form, upon whose submission the same url is invoked with post method
- In post method, all the form values are requested and stored in appropriate variables
- Connection is made with the in-device MySQL, with the password provided
- Cursor is created using which execution of queries take place
- The database and tables are created if they don't already exist
- New values are tried to be inserted into the table, they will have an error if the email is an existing one because it is the primary key of the table and must be unique, upon which, the result.html file is rendered with appropriate message
- If the email is unique, then the data gets inserted , we commit the database and render the result.html file with appropriate message

```
if __name__ == '__main__':
    app.run(host="0.0.0.0",port=5555, debug = False) #debug = true while developing
```

- The if condition makes sure that the implementation works only when it is run through the application
- Host and port can be changed according to requirement
- debug must be False while deploying, it can be kept True for developing purposes

## basic.html

This file consists of the basic syntax or template of any html file that is used in the application, including the style, so that it need not be retyped.

## home.html

This file just has a welcome message and a hyper link to the actual assessment page.

## index.html

This file consist the form that takes email, a Boolean value and favourite number as input.
Upon submission, email validations and required validations are performed, then using url_for() the assessment function is again called with the post method, and further processing is done.

## result.html

This file contains a message that is passed from the python program, which is simply displayed.