

Block wise Explanation:-

```
from flask import Flask, request, render_template, url_for, redirect
import csv
app = Flask(__name__, template_folder = "templates")
```

- flask framework for the UI
- csv for handling comma separated values of a file
- 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 = True
        else:
            like = False
        fav_num = request.form.get("fav") # to get favourite number from form
        with open("cust_disc.csv", mode='a+', newline="") as file: # to append contents
            file_writer = csv.writer(file, delimiter=',') # to write
            file_reader = csv.reader(file, delimiter=',') # to read
            file.seek(0) # to search from beginning of the file
            for row in file_reader: # to traverse
                if row[0] == email: # if already exists
                    return render_template("result.html", alert_msg="Thank You, we already have your preference stored!")
            else: #else write in csv file
                file_writer.writerow([email, like, fav_num])
            return render_template("result.html", alert_msg="Thank You, Your preference has been stored successfully!")
```

- 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
- CSV file is opened in append and read mode
- Objects for writer and reader of the files are created with delimiter as comma
- File Pointer is moved to the beginning of the file
- The file is traversed row by row, in each row the 1st value is matched with the email we got as input from form, if it matches, we pass an appropriate message for the proper rendering of the result.html page
- If the email doesn't already exist, the input values are written in the csv file and, we pass an appropriate message and render the result.html page

```
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.