

## EXPERIMENT NO. 2

### Use GAE launcher to launch the web applications

**AIM:** Use GAE launcher to launch the web applications.

**Software Tools :** Python- 2.5.4, Jre-8u201-windows-xp, GoogleCloudSDK Inst, VS Code

**THEORY:** You can use Google App Engine to host a static website.

Static web pages can contain client-side technologies such as HTML, CSS, and JavaScript.

Hosting your static site on App Engine can cost less than using a traditional hosting provider, as App Engine provides a free tier. Sites hosted on App Engine are hosted on the REGION\_ID .r.appspot.com subdomain, such as [my-project-id].uc.r.appspot.com.

After you deploy your site, you can map your own domain name to your App Engine-hosted website. Create a new Cloud Console project or retrieve the project ID of an existing project to use: Goto the Project page2.

Install and then initialize the Google Cloud SDK. Download the SDK Creating a website to host on Google App EngineBasic structure for the project is guide use the following structure for the project:

app.yaml: Configure the settings of your App Engine application.

#### **Web app designing**

##### **HTML**

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of tags
- **HTML:**
- Hyper Text Markup Language
- Every tag has opening and closing bracket.

<html>---Opening tag

</html>---closing tag

## STEPS

1) Create one folder on desktop

Ex. TEST

2) Open VS Code and make the following four file app.yml, main.py, index.html, results.html in vs code and save it in folder TEST

1) app.yml (save as YAML)

```
runtime: python27
threadsafe: true

handlers:
- url: /
  script: main.app
```

2) main.py (save as python )



```
import os
import json
import urllib
import webapp2
from google.appengine.ext.webapp import template

class MainPage(webapp2.RequestHandler):
    def get(self):

        path = os.path.join(os.path.dirname(__file__), "index.html")
        template_values = {}
        self.response.out.write(template.render(path, template_values))

    def post(self):
        pincode = self.request.get('zipCode')
        if not pincode.isnumeric() or not len(pincode) == 6:
            template_values = {
                "error": "Incorrect Pin Code (String / False Code entered)"
            }
            path = os.path.join(os.path.dirname(__file__), 'index.html')
            return self.response.out.write(template.render(path, template_values))
        url = "https://api.postalpincode.in/pincode/" + pincode
        data = urllib.urlopen(url).read()
        data = json.loads(data)
        if(data[0]['Status'] == 'Success'):
            post_office = data[0]['PostOffice'][0]['State']
            name = data[0]['PostOffice'][0]['Name']
            block = data[0]['PostOffice'][0]['Block']
```

```

district = data[0]['PostOffice'][0]['District']
template_values = {
    "post_office": post_office,
    "name": name,
    "block": block,
    "district": district
}
path = os.path.join(os.path.dirname(__file__), 'results.html')
self.response.out.write(template.render(path, template_values))
else:
    template_values = {}
path = os.path.join(os.path.dirname(__file__), 'error.html')
self.response.out.write(template.render(path, template_values))

app = webapp2.WSGIApplication([(' / ', MainPage)], debug=True)

```

3) index.html (save as HTML)



```

<html>
<style>
.weatherText {
    font-family: 'Lato', 'sans-serif';
    font-size: 24px;
    text-align: center;
}

#weatherForm {
    padding: 20px;
}

#weatherSubmit {
    color: white;
    background-color: #083375;
    padding: 5px 20px;
    border-radius: 5px;
    margin-top: 20px;
}

#weatherSubmit:hover {
    cursor: pointer;
}
body {
    display: flex;
    justify-content: center;
    align-items: center;
}
.card {
    border: 2px solid black;
    width: 50%;
}

```

```

        justify-content: center;
        align-items: center;
    }
</style>
<head>
    <title class="alignct">Post Office Finder</title>
    <link

        href="https://fonts.googleapis.com/css2?family=Lato:wght@400;700&display=swap"
        rel="stylesheet"
    />
</head>
<body>
    <div class="card">
        <h2 class="weatherText">Post Office Finder Using WebApp</h2>
        <h1 id="error_head" style="display: none" value="{{error}}>{{error}}</h1>

        <form class="weatherText" id="weatherForm" action="/" method="post">
            Location Zip Code:
            <input
                class="weatherText"
                id="weatherInput"
                type="text"
                name="zipCode"
            /><br />
            <input
                class="weatherText"
                id="weatherSubmit"
                type="submit"
                value="Submit"
            />
            <button
                id="weatherSubmit"
                class="weatherText"
                onclick="document.getElementById(
                    'weatherInput').value = ''"
            >
                Clear
            </button>
        </form>
    </div>
    <!-- <script>
        let err = document.getElementById('error_head');
        function myFunction() {
            alert('Please Enter the Valid Pin Code!');
        }
        if (err) {
            myFunction();
        }
    </script> -->
</body>

```

```
</html>
```

4) results.html (save as HTML)

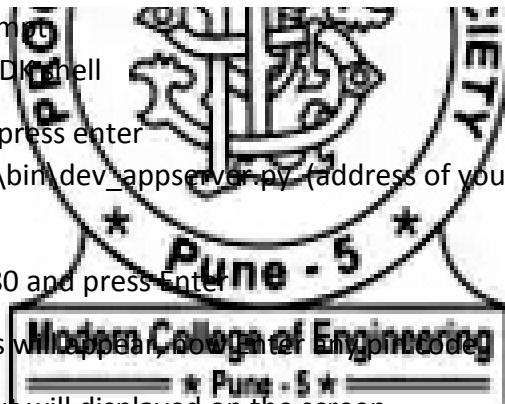
```
<!DOCTYPE html>
<html lang="en">
  <style>
    body {
      display: flex;
      justify-content: center;
      align-items: center;
    }
    #weatherResults {
      background-color: #83e9c2;
      font-family: 'Lato', sans-serif;
      font-size: 24px;
      padding: 30px;
      display: inline-block;
      text-align: center;
      margin: 20px;
      margin-top: 10%;
      border: 2px solid black;
      border-radius: 5px;
    }
  </style>
  <head>
    <meta charset="UTF-8" />
    <title>Post Office Information</title>
    <link href="https://fonts.googleapis.com/css2?family=Lato:wght@400;700&display=swap" rel="stylesheet"
    />
  </head>
  <body>
    <div id="weatherResults">
      <table>
        <tr>
          <th>
            <h3>State of Post Office :</h3>
          </th>
          <th>
            <h3>{{ post_office }}</h3>
          </th>
        </tr>
        <tr>
          <th>
            <h3>Name of Post Office :</h3>
          </th>
          <th>
            <h3>{{ name }}</h3>
          </th>
        </tr>
      </table>
    </div>
  </body>
</html>
```

```

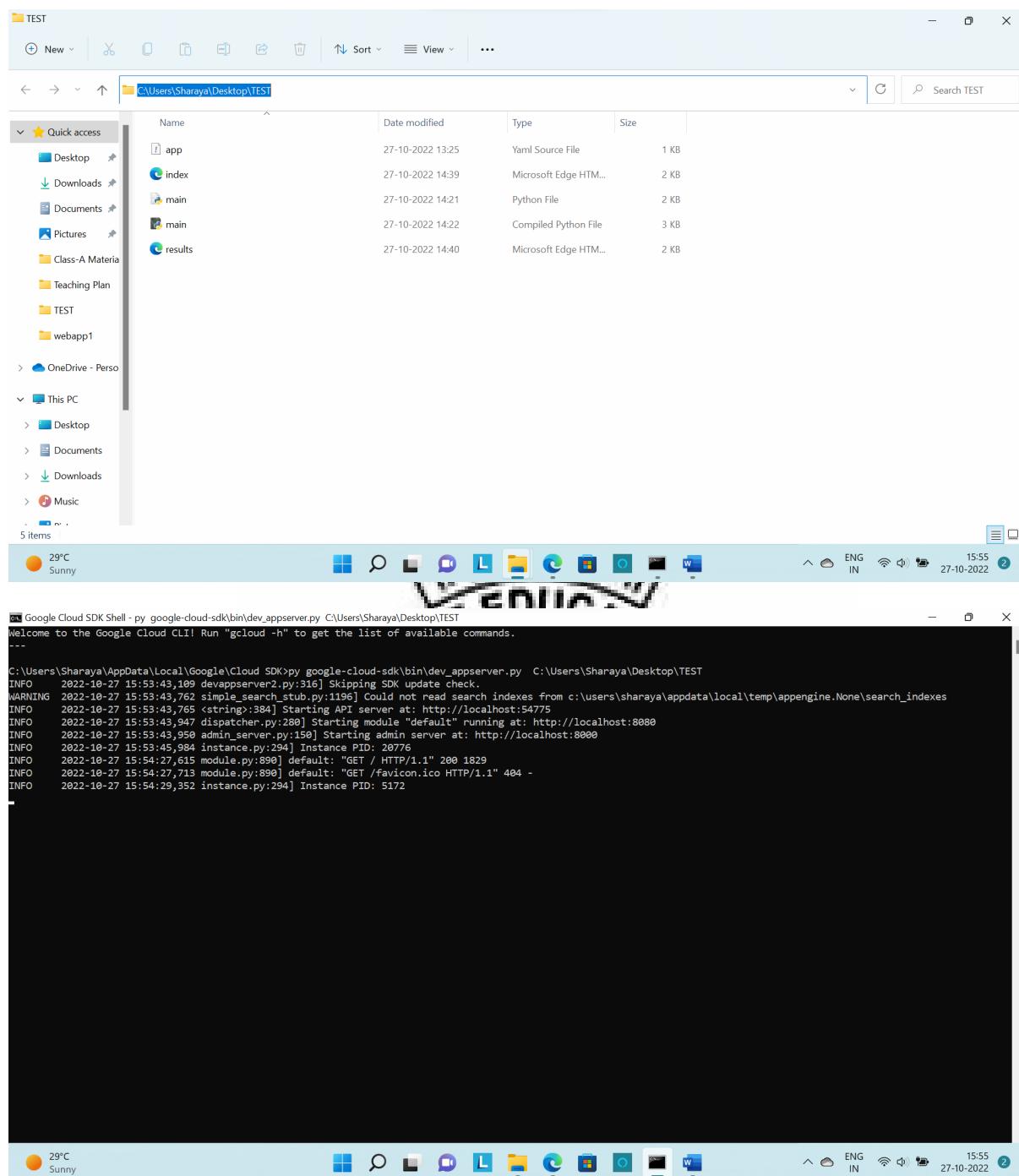
        </th>
    </tr>
    <tr>
        <th>
            <h3>Block of Post Office:</h3>
        </th>
        <th>
            <h3>{{ block }}</h3>
        </th>
    </tr>
    <tr>
        <th>
            <h3>District of Post Office:</h3>
        </th>
        <th>
            <h3>{{ district }}</h3>
        </th>
    </tr>
</table>
<a href="http://localhost:8080/"><h4>Back to the Home page</h4></a>
</div>
</body>
</html>

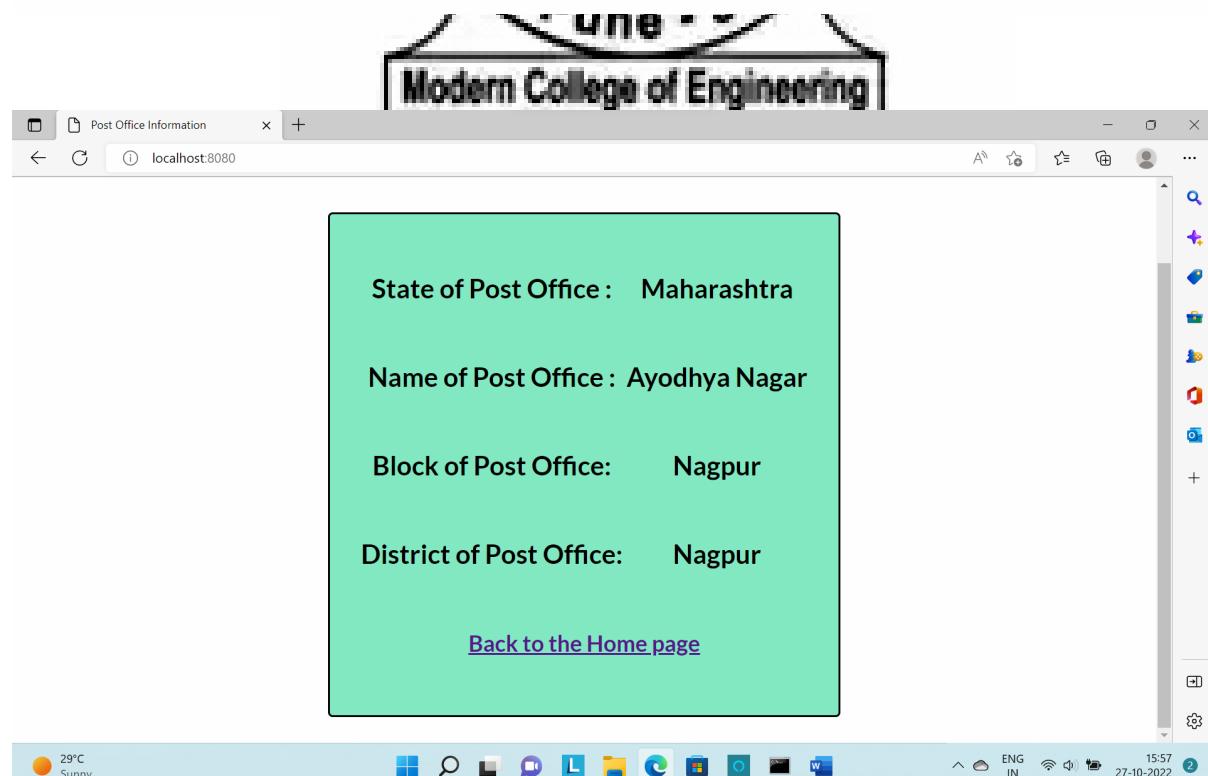
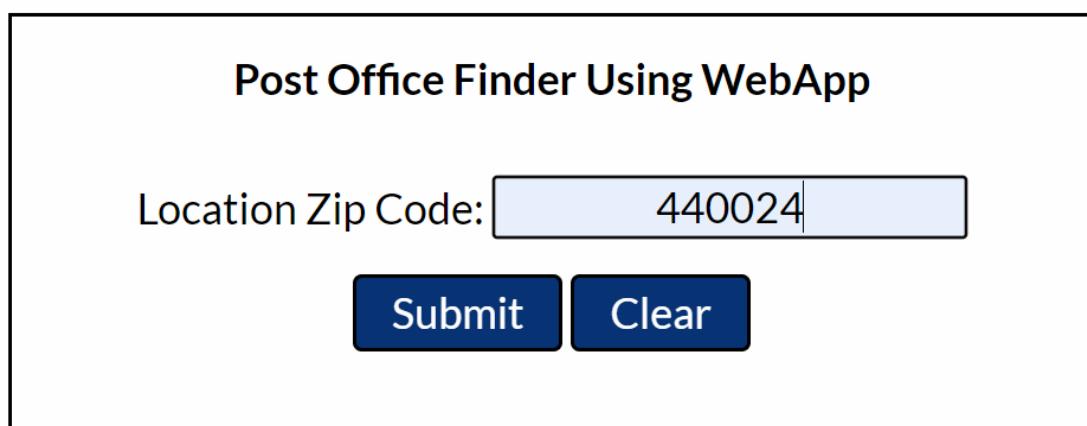
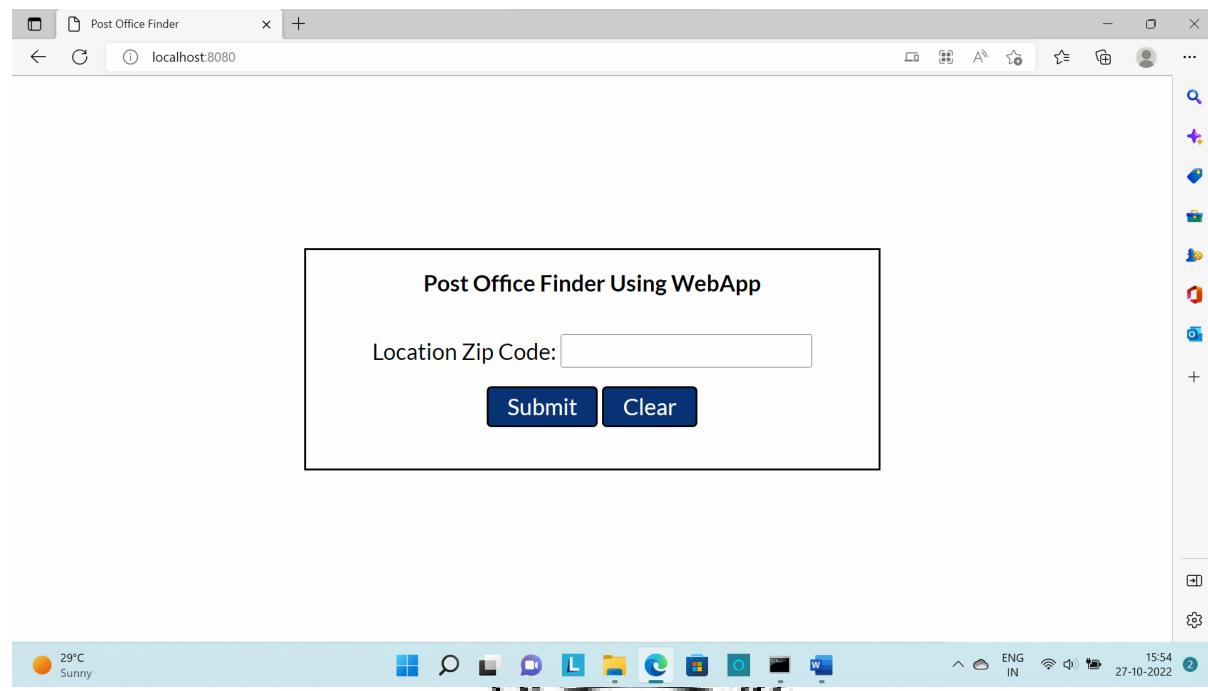
```

- 3) Go to command prompt
- 4) Open google cloud SDK shell
- 5) Type command and press enter  
py google-cloud-sdk\bin\dev\_appserver.py (address of your folder where you have saved these two files.)
- 6) Go to Local Host:8080 and press Enter
- 7) First output screen is will appear now enter Emp no code
- 8) After that final output will displayed on the screen.



## OUTPUT:





**Conclusion:**

