DyGraphs | Reporting Tool

A PROJECT REPORT

Submitted in partial fulfilment for the award of the degree

Of

Master of Technology

In

Information Technology

By

NAGESH REDDY
(17MIN0629)

Under the guidance of
Prof. Wasudev Rahane
Professor

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Information Technology and Engineering

November 2020

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Information Technology and Engineering

DECLARATION BY THE CANDIDATE

I hereby declare that the thesis entitled "**DyGraphs | Reporting Tool**" submitted by me to Vellore Institute of Technology, Vellore, in partial fulfilment of the requirement for the award of the degree of **Master of Technology** in **Information Technology** is a record of bonafide project work carried out by me under the supervision of **Prof. Wasudev Rahane**. I further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Pune

Date: **Signature of the Candidate**

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Information Technology and Engineering

BONAFIDE CERTIFICATE

This is to certify that the project work entitled “**DyGraphs | Reporting Tool**” by **NAGESH REDDY (17MIN0629)**, to Vellore Institute of Technology, Vellore, in partial fulfilment of the requirement for the award of the degree of **Master of Technology** in **Information Technology**, is a project bonafide work carried out by him/her under my supervision. The project fulfils the requirement as per the regulations of this Institute and in my opinion meets the necessary standards for submission. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this Institute or any other Institute or University.

Prof. Wasudev Rahane
Internal Supervisor
Professor
VIT University

Internal Examiner(s) **External Examiner(s)**

Table of Contents

ABSTRACT

"DyGraphs | Reporting Tool" is an attempt to create and develop a web platform to help in presenting dashboards based on dynamic data. The main objective of this project is creating a tool for showcasing data visually by using pre-processed data. Where tool takes the pre-processed data as input and gives the output in charts or graphs or geo-charts plus interactive format. The motto of this project is to accept any kind of data and provide the pivotal reports, in simple words accept dynamic data and provide dashboard on complete data set in terms of small stories and if possible, make future prediction. This is a web-based application, will be deployed in cloud.

In the current marketplace most of the visualization dashboards are coded only for particular data sets, if we want to use the script for different data then we have to modify the script a bit. This will be a problem for people who does not have coding knowledge. Even if the people who have coding knowledge, they can do it, but again it will be time consuming for them as well. Some of the tools does not provide future trend.

The motto of this project is to provide the visualised charts with interactivity. Any person irrespective of the person's knowledge on coding can import the pre-formatted data into application to see the data visually.

ACKNOWLEDGEMENT

This major project would have been possible without the valuable assistance of many people to whom we are indebted. The satisfaction that accompanies the successful of any task would be incomplete without the mention of people who made it possible whose is constant guidance of encouragement crowned our efforts with success in various stages.

My heartfelt gratitude goes to my guide Prof. Wasudev Rahane for his valuable and timely help, which boomed the project to this extent.

I would like to express my sincere gratitude and thanks to Vellore Institute of Technology and Wipro Technologies for the WISTA course and the encouragement and inspiration given throughout the course of study.

I extend my deep sense of gratefulness to The Management and The Faculty Members of WIPRO WISTA program for their pleasing support.

Special word of thanks is to our classmates for providing the moral support. Finally, yet importantly, I owe all the efforts to my beloved Family members and all my dear friends towards this endeavor, which in turn changed all the obscure today an explicit tomorrow.

NAGESH REDDY
(17MIN0629)

CHAPTER 1

1. INTRODUCTION

Data visualization is a form of communication that portrays dense and complex information in graphical form. The resulting visuals are designed to make it easy to compare data and use it to tell a story – both of which can help users in decision making.

Data visualization can express data of varying types and sizes: from a few data points to large multivariate datasets.

Data visualization can be expressed in different forms. Charts are a common way of expressing data, as they depict different data varieties and allow data comparison.

The type of chart you use depends primarily on two things: the data you want to communicate, and what you want to convey about that data. These guidelines provide descriptions of various types of charts and their use cases.

1.1 PURPOSE

A website that allows a user to see their business/data visually. The data can be dynamic, still user can fetch reports.

1.2 SCOPE

The project plan is to implement a website for report generation with user friendly UI. This website mainly consists of 1 module with majorly 2 sub-modules (Sample Reports and Designing Reports).

1.3 DEFINITIONS, ACRONYMS AND ABBREVIATIONS

1.3.1 Definitions:

- **SQLite**:

  SQLite is a relational database management system (RDBMS) contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program.

- **HTML**:

  Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

  Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

  HTML elements are the building blocks of HTML pages. With HTML constructs, images, and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes, and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img/> and <input/> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags but use them to interpret the content of the page.

- **MY-SQL**:

  MySQL, pronounced either "My S-Q-L" or "My Sequel," is an open source relational database management system. It is based on the structure query language

(SQL), which is used for adding, removing, and modifying information in the database. Standard SQL commands, such as ADD, DROP, INSERT, and UPDATE can be used with MySQL. MySQL can be used for a variety of applications but is most commonly found on Web servers. A website that uses MySQL may include Web pages that access information from a database. These pages are often referred to as "dynamic," meaning the content of each page is generated from a database as the page loads. Websites that use dynamic Web pages are often referred to as database-driven websites. Many database-driven websites that use MySQL also use a Web scripting language like Java Script to access information from the database. MySQL commands can be incorporated into the Java Script code, allowing part or all of a Web page to be generated from database information. Because MySQL and Java Script are both open source (meaning they are free to download and use), the Java Script/MySQL combination has become a popular choice for database-driven websites.

- **Python :**

Python is an interpreted, high-level and general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

1.3.2 Acronyms and Abbreviations

ACRONYM	EXPANSION
SQLite	Embeddable Relational Database Management System which is written in ANSI-C.
HTML	**HTML** (**H**yper **T**ext **M**arkup **L**anguage)
MY-SQL	**MY SQL** (**S**tructured **Q**uery **L**anguage)
Gunicorn Web Server	**Green Unicorn** (Also pronounced as gun – i – corn) – Django's default webserver

1.4 OVERVIEW

The rest of this SRS document describes the various system requirements, interfaces, features and functionalities in detail.

CHAPTER 2

Literature Survey of the Proposed System

2. LITERATURE SURVEY

In a conference Jeff Sellers & Kathy Gosa, SST Facilitators on Monday, February 13, 2017 said:

Dynamic reports are created at runtime. Each time a dynamic report is run, it gathers the most recent data in the Data Warehouse. Only the report definition, which remains the same over time, is stored.

Static reports are run immediately upon request, and then stored with the data in the Completed Reports module.

Previous State of Reporting

- Majority of Reporting Are Operational
- Silo-Based Reporting
- Lack of Enterprise Reporting Strategy
  - Internal vs. External
- No Self-Service Reporting Options
- Custom .NET Dashboards Development

Enterprise-class business analytics platforms that can scale up to hundreds of thousands of users.

- Supports user choice of data architecture
- Create operational and analytical reporting
- Self-service reporting – Natural path from report consumer to report developer
- Offers a fast, in-memory Data Engine that is optimized for analytics

Pros

- Support Operational and Analytical Reporting
- Stunning Data Visualizations Options
- Interactive Discovery Solution
    - Can drill down from summarized view to detail and underlying data source
- Data Source Integration
    - Can blend data from multiple sources
    - Can connect to your data no matter where it lives
- Supports Mobile Devices
- Drag-and-Drop Report Design Interface

Cons

- Cost Prohibitive
    - Core licensing model for Tableau Server
- Initial Data Preparation
    - Requires strong technical skills to build initial structure
- Complexity of Advanced Dashboard Design
    - High-level or technical expertise required
    - Will require IT intervention
- Data Management
    - Works best with Data Extracts vs. live connections
    - IT management of another redundant data repository
- Security for External Users
    - No one-stop authentication mechanism for external and internal users
- Change Management
    - No concept of report version

CHAPTER 3

3.1 HARDWARE REQUIREMENTS:

Memory	- 4 GB RAM and above for better speed.
Windows	- 32bit/64bit
Interface	- Mouse, keyboard
Processor	- Intel(R) Core (TM) i3-6006U CPU @ 2.00GHz, 2000 Mhz, 2 Core(s)

3.2 SOFTWARE REQUIREMENTS:

Operating System	- Windows 7(ultimate, enterprise) or higher version
Browser	- Any of Chrome, Mozilla, Internet Explorer etc.
Front end	- HTML, Bootstrap, Django, CSS.
Back end	- Django, Python Pandas.

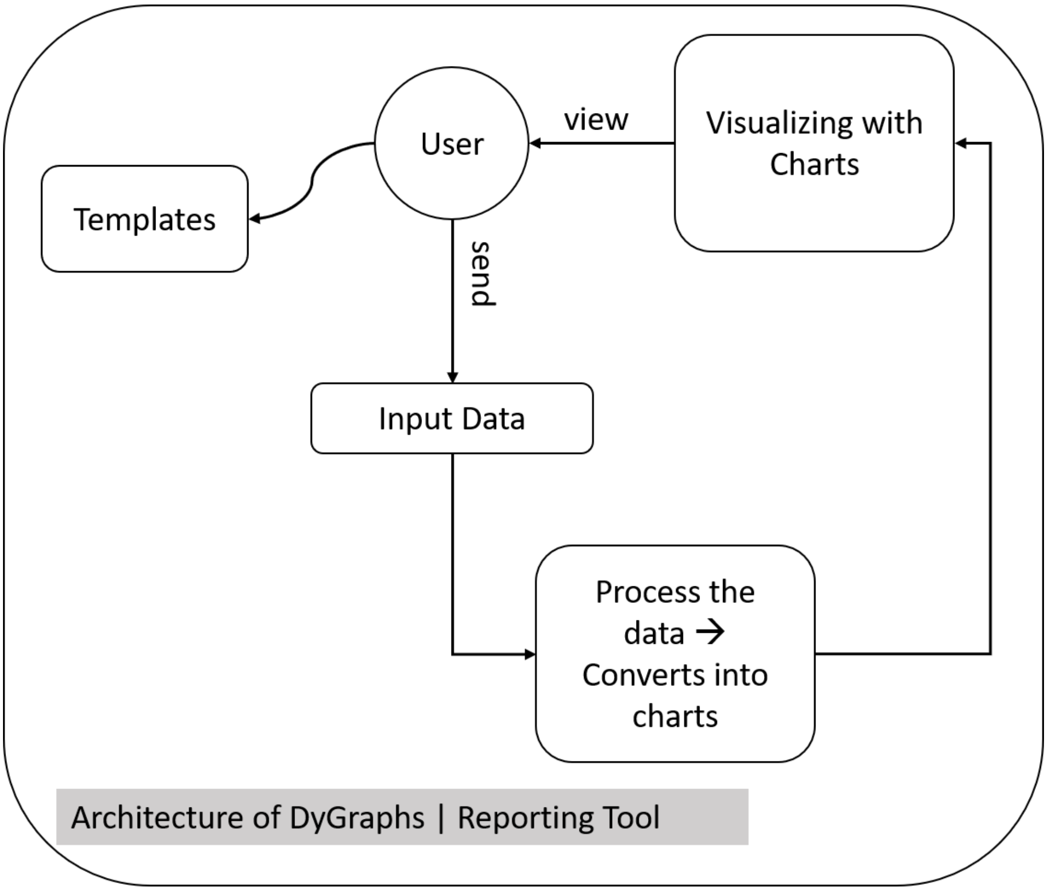
CHAPTER 4

Detailed Design of the System

4.1System Architecture

Below is the architecture diagram for the **DyGraphs | Reporting Tool**. This project designed in python web application. It uses Django framework for web interface which sends the http request to the models and get the response from the models and shows in html.

- We have a Web Interface designed in HTML and CSS.
- Django Framework includes http web server and template engine to process request and give response.
- Web Interface takes input data.
- Python libraries/Frameworks Processes data and provides data out graphically/visually.



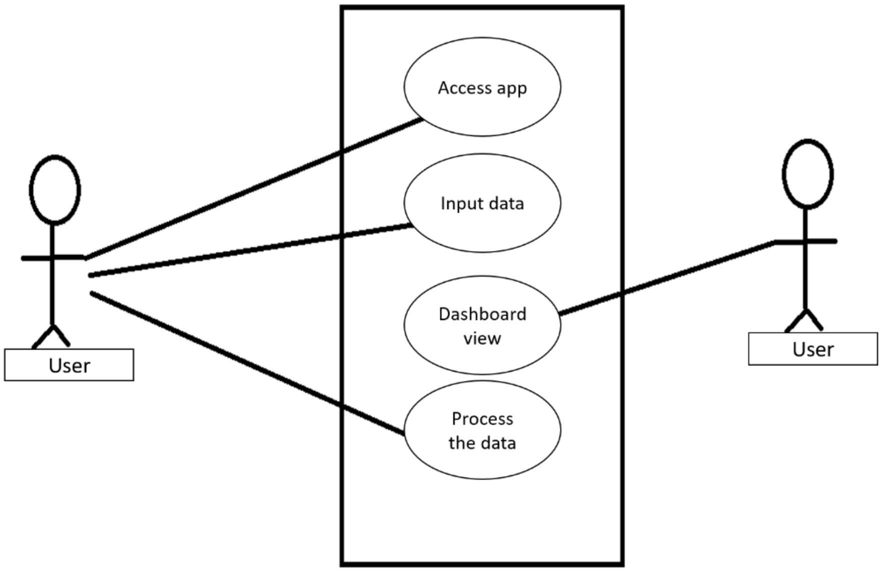
Architecture of DyGraphs | Reporting Tool

4.2 UML Diagrams

The Unified Modelling Language is a standard language for specifying, process analysis and design, visualization, constructing and documenting the artefacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing objects-oriented software and the software development process. UML uses mostly graphical notations to express the design of software projects. Different UML diagrams using is this project are namely Use case Diagram, Activity Diagram, Sequence Diagram, Collaboration Diagram, Component Diagram, and Deployment Diagram.

4.3Use Case Diagram

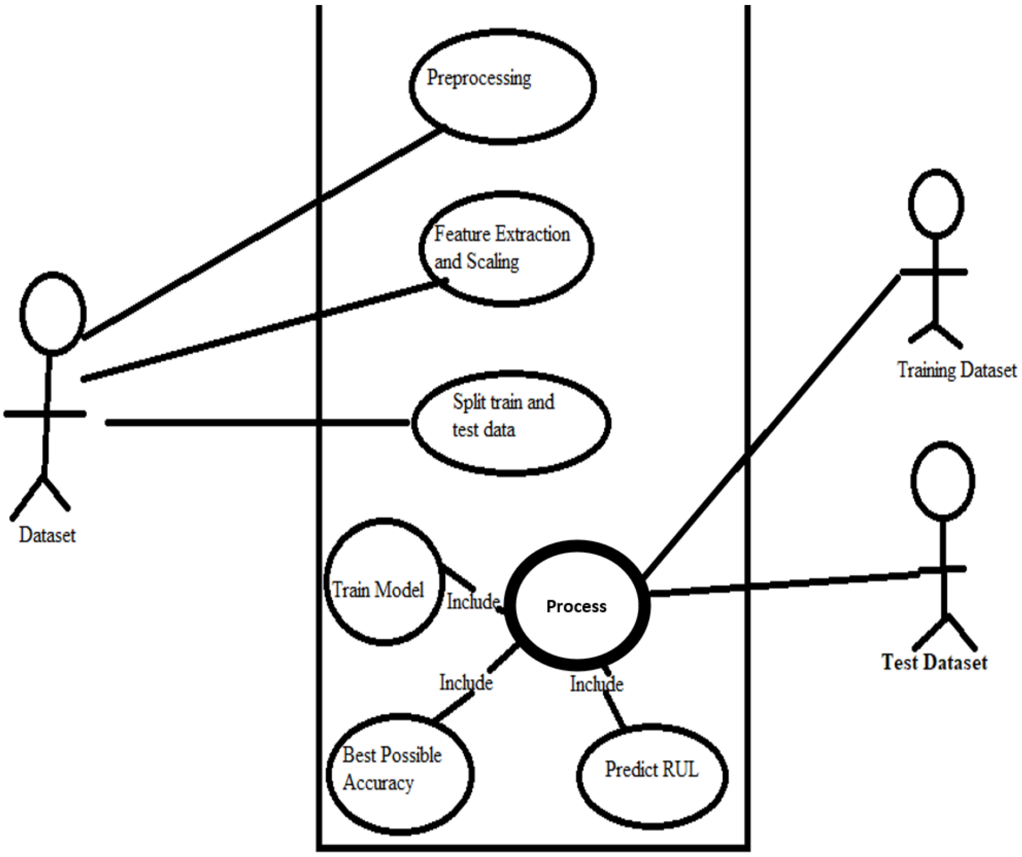


First mode takes the dataset and do the cleaning and pre-processing of the data.

Next sept is feature extraction and scaling of data.

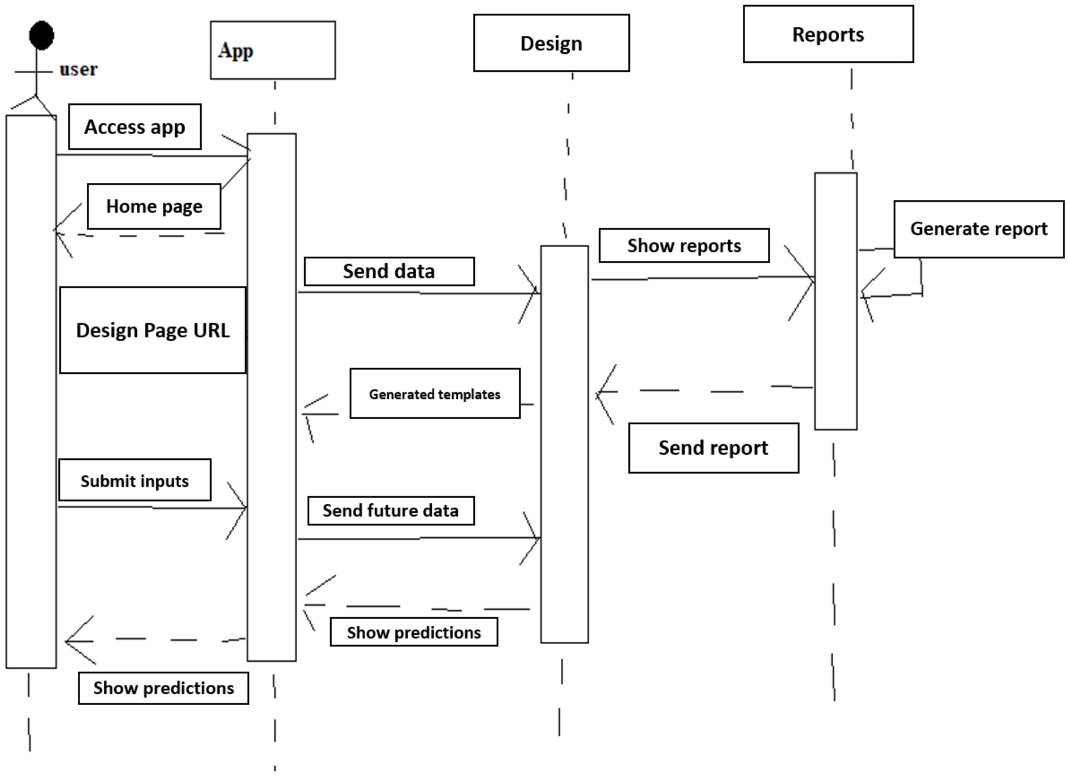
Model splits the data into test and train datasets.

Then the programs will be trained to find the best possible accuracy and predict the result based on test dataset.



4.4Sequence Diagram

The below diagram represents the interaction between user and the system objects in the sequential order.



Below is the sequence of interactions done by the system.

- Here user starts the interaction by accessing the app.
- App sends the home page.
- User puts the inputs with required data set.
- Design shows the number of template previews.
- User sends the future data or new data to get the future prediction.
- Reports page generates the reports based on the data sent and processed in design page.

4.5 TESTING

The objective of testing is to find errors or faults. Testing is the process of trying to discover every possible fault or bug in a software product. In software development life cycle (SDLC), it gives us a way to investigate the functionality of components, sub-modules, modules and/or a completed product. Testing is the process of exercising software with the intent of ensuring that the Software system meets its functional requirements and user expectations and does not break down in an unacceptable manner. There are various types of testing. Each type addresses a specific testing requirement.

The success of the testing process in determining the errors is mostly depends upon the test case criteria, for testing any software needs a description of the expected behaviour of the system and method of determining whether the observed behaviour confirmed to the expected behaviour.

There are two basic approaches for testing. They are:

1. Functional Testing
2. Structural Testing

In Functional Testing test cases are decided mainly on the basis of requirements of the system or module and the internals of the system or modules are not considered for selection of test cases. This is also called Black Box Testing.

In Structural Testing test cases are generated on actual code of the program or module to be tested. This is called White Box Testing.

LEVELS OF TESTING

Since the errors or faults in the software can be damaged at any phase. So, the testing process needs to be performed at different levels during the development phase. The primary levels of testing are Unit, Integration, System and Acceptance Testing.

Unit testing

Unit testing involves the design of test cases that checks that the internal program logic is working properly, and that program inputs provides valid outputs. All decision branches and internal code flow should be checked. It is the testing of individual software modules or units of the system. It will be carried out after the completion of an individual unit ahead of integration. This is a structural testing, that build on knowledge of its development and is invasive. Unit tests perform basic tests at module level and test a specific business process, application, and/or system configuration.

Integration testing

Integration tests are created to test integrated software components to find if they run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests determine that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is especially aimed at exposing the problems that arise from the combination of components.

System Testing

System testing assures that the entire integrated software system meets requirements. It tests a configuration to ensure known and certain results. An example of system testing is the configuration-oriented system integration test. System testing is based on process definitions and flows, highlight pre-driven process links and integration points.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires important participation by the end user. It also assures that the system meets the functional requirements.

Output Testing

After performing arts, the validation testing, succeeding step is output testing of the projected system, since no system may be helpful if it doesn't turn out such output within the specified format. Asking the users regarding the format needed by them tests the outputs generated or displayed by the system into account. Thus, the output format is considered in two ways in which one is on screen and another in written format.

CHAPTER 5

IMPLEMENTATION OF THE SYSTEM

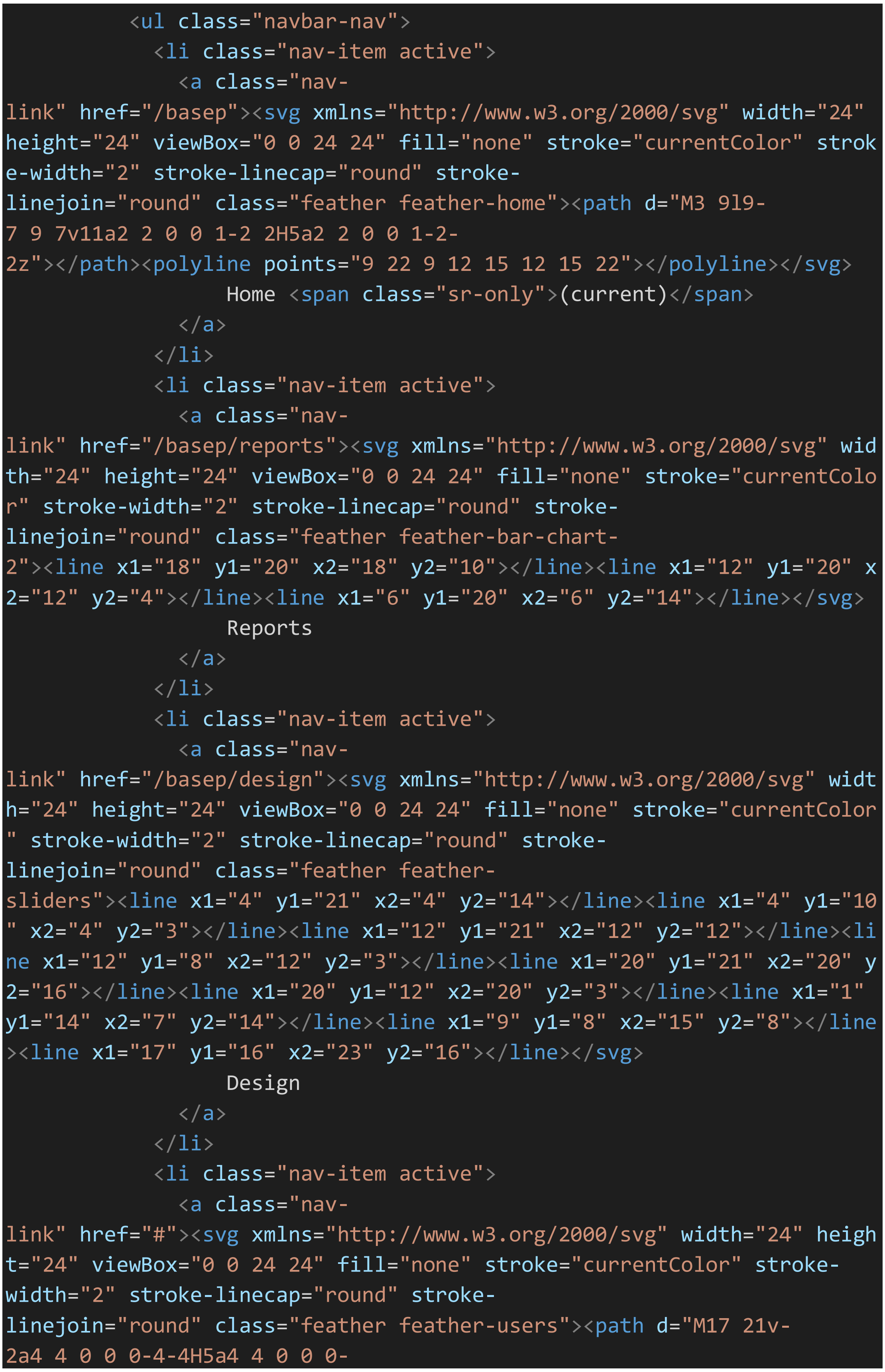
SCRIPTING

#--#

global.html

#--#

```
<!DOCTYPE html>
<html lang="en">
  <script src="https://cdn.jsdelivr.net/npm/chart.js@2.8.0"></script>
    <head>
        <link rel="stylesheet"
        href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css"
        integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2xXr2"
        crossorigin="anonymous">

        {% block head %}
        <title>Global</title>
        {% endblock %}
    </head>
    <div class="container" style="background-color: black; margin-top:1%">
      <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <a class="navbar-brand" href="/basep"><b>DyGraphs | Reporting Tool</b></a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarNav">

        </div>
        <div id="navbar"  style="float:right;">
```



```
          <ul class="navbar-nav">
            <li class="nav-item active">
              <a class="nav-
link" href="/basep"><svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" strok
e-width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-home"><path d="M3 9l9-
7 9 7v11a2 2 0 0 1-2 2H5a2 2 0 0 1-2-
2z"></path><polyline points="9 22 9 12 15 12 15 22"></polyline></svg>
                  Home <span class="sr-only">(current)</span>
              </a>
            </li>
            <li class="nav-item active">
              <a class="nav-
link" href="/basep/reports"><svg xmlns="http://www.w3.org/2000/svg" wid
th="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColo
r" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-bar-chart-
2"><line x1="18" y1="20" x2="18" y2="10"></line><line x1="12" y1="20" x
2="12" y2="4"></line><line x1="6" y1="20" x2="6" y2="14"></line></svg>
                  Reports
              </a>
            </li>
            <li class="nav-item active">
              <a class="nav-
link" href="/basep/design"><svg xmlns="http://www.w3.org/2000/svg" widt
h="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor
" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-
sliders"><line x1="4" y1="21" x2="4" y2="14"></line><line x1="4" y1="10
" x2="4" y2="3"></line><line x1="12" y1="21" x2="12" y2="12"></line><li
ne x1="12" y1="8" x2="12" y2="3"></line><line x1="20" y1="21" x2="20" y
2="16"></line><line x1="20" y1="12" x2="20" y2="3"></line><line x1="1"
y1="14" x2="7" y2="14"></line><line x1="9" y1="8" x2="15" y2="8"></line
><line x1="17" y1="16" x2="23" y2="16"></line></svg>
                  Design
              </a>
            </li>
            <li class="nav-item active">
              <a class="nav-
link" href="#"><svg xmlns="http://www.w3.org/2000/svg" width="24" heigh
t="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-
width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-users"><path d="M17 21v-
2a4 4 0 0 0-4-4H5a4 4 0 0 0-
```

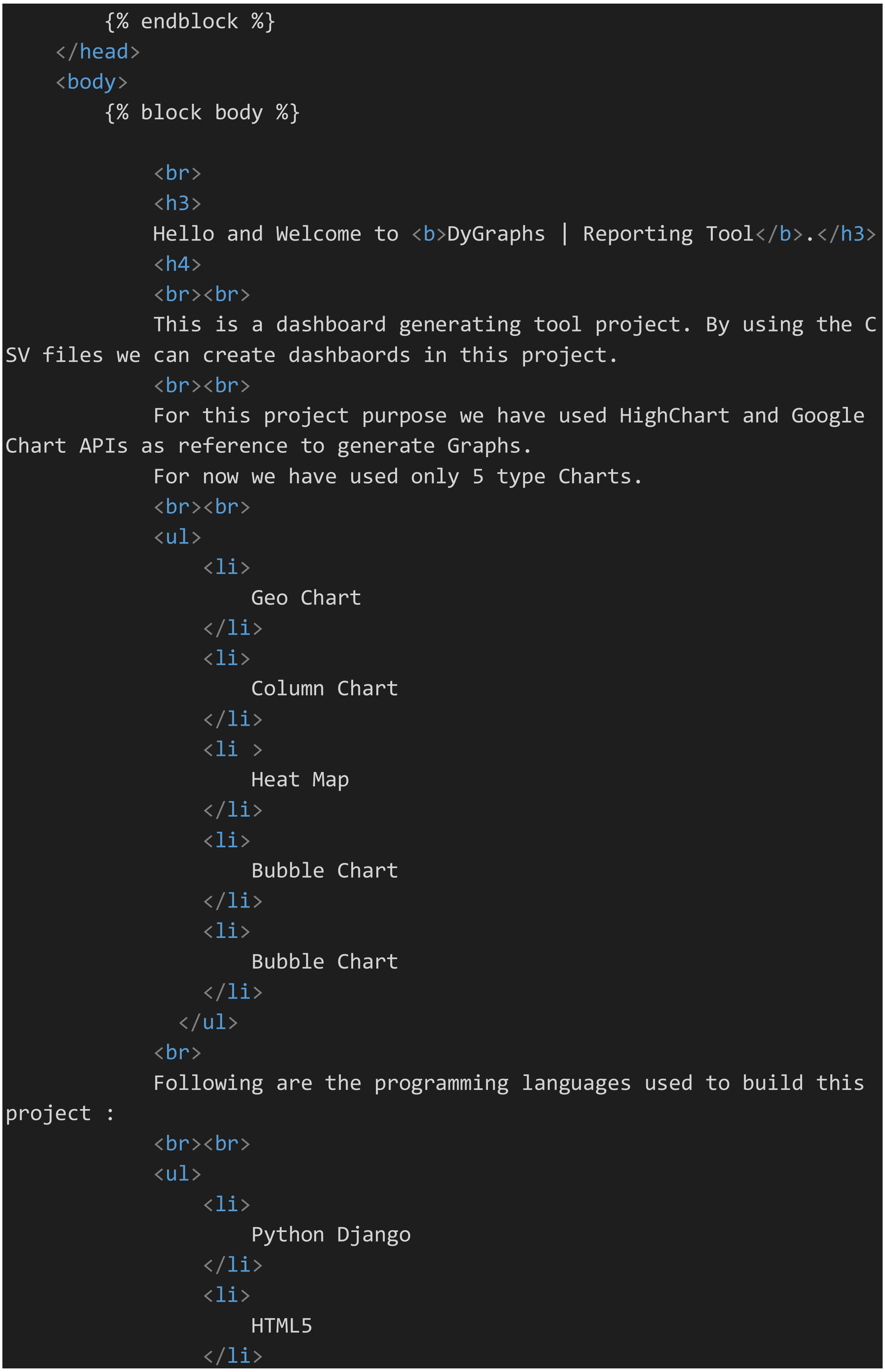
```
4 4v2"></path><circle cx="9" cy="7" r="4"></circle><path d="M23 21v-
2a4 4 0 0 0-3-
3.87"></path><path d="M16 3.13a4 4 0 0 1 0 7.75"></path></svg>
                  People
              </a>
            </li>
            <!--
                <li class="nav-item active">
                    <a class="nav-link" href="/basep/graphs">Graphs</a>
                </li>
                <li class="nav-item active">
                    <a class="nav-link" href="#">Design</a>
                </li>
              -->
          </ul>
        </div>
      </nav>
    </div>

    <body>
    <br>

        <div class="container" style="margin-top:1%">
        {% block body %}
        <h1>Global Body</h1>
        {% endblock %}



        </div>
    </body>
</html>
```

#---#

home.html

#---#

```
{% extends 'global.html' %}

<html>
    <head>
        {% block head %}
        <title>Home</title>
```



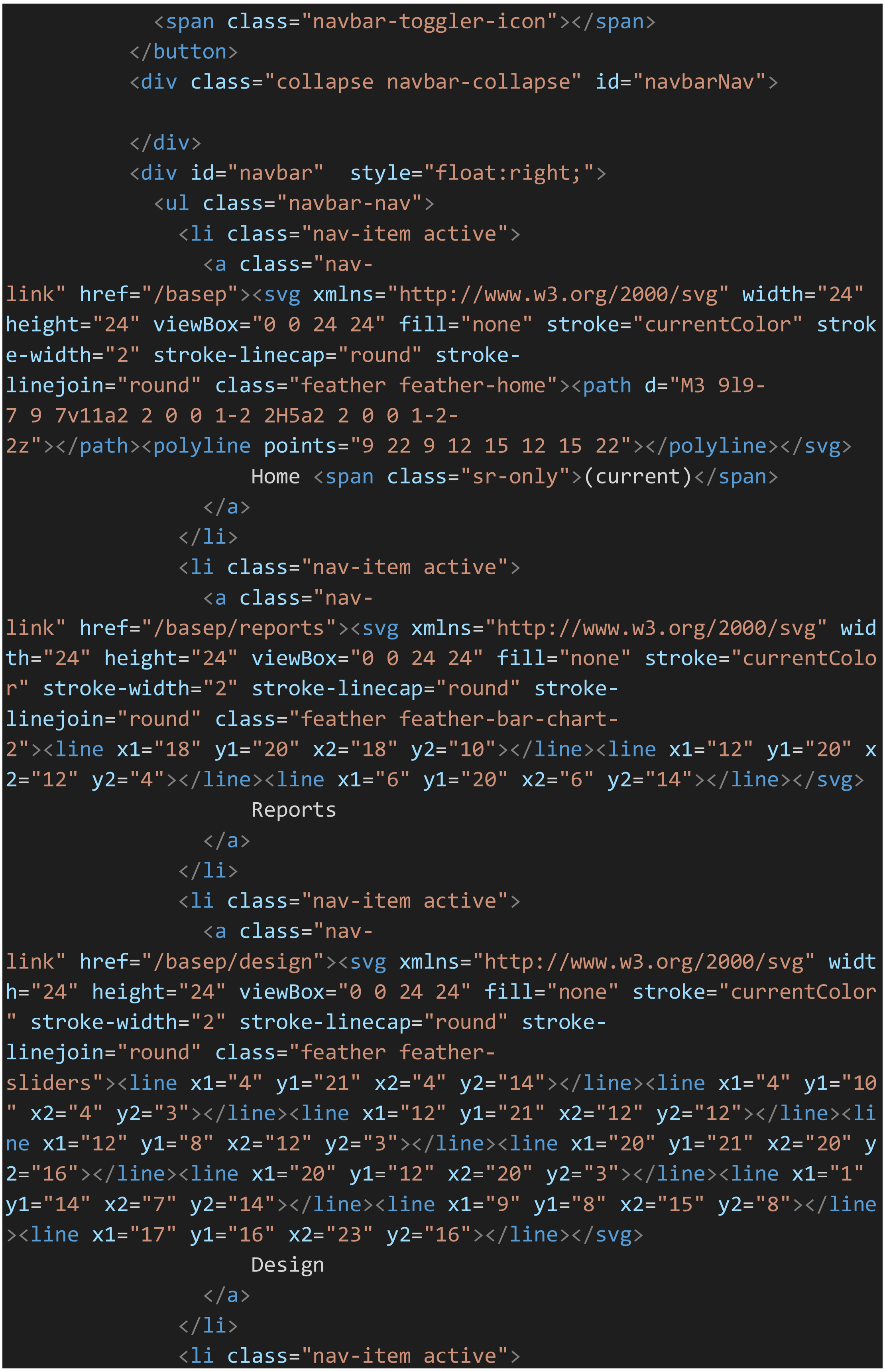
```
        {% endblock %}
    </head>
    <body>
        {% block body %}

            <br>
            <h3>
            Hello and Welcome to <b>DyGraphs | Reporting Tool</b>.</h3>
            <h4>
            <br><br>
            This is a dashboard generating tool project. By using the C
SV files we can create dashbaords in this project.
            <br><br>
            For this project purpose we have used HighChart and Google
Chart APIs as reference to generate Graphs.
            For now we have used only 5 type Charts.
            <br><br>
            <ul>
                <li>
                    Geo Chart
                </li>
                <li>
                    Column Chart
                </li>
                <li >
                    Heat Map
                </li>
                <li>
                    Bubble Chart
                </li>
                <li>
                    Bubble Chart
                </li>
              </ul>
            <br>
            Following are the programming languages used to build this
project :
            <br><br>
            <ul>
                <li>
                    Python Django
                </li>
                <li>
                    HTML5
                </li>
                <li >
                    CSS Styles
                </li>

              </ul>
        </h4>
        {% endblock %}
    </body>
</html>
```

#---#

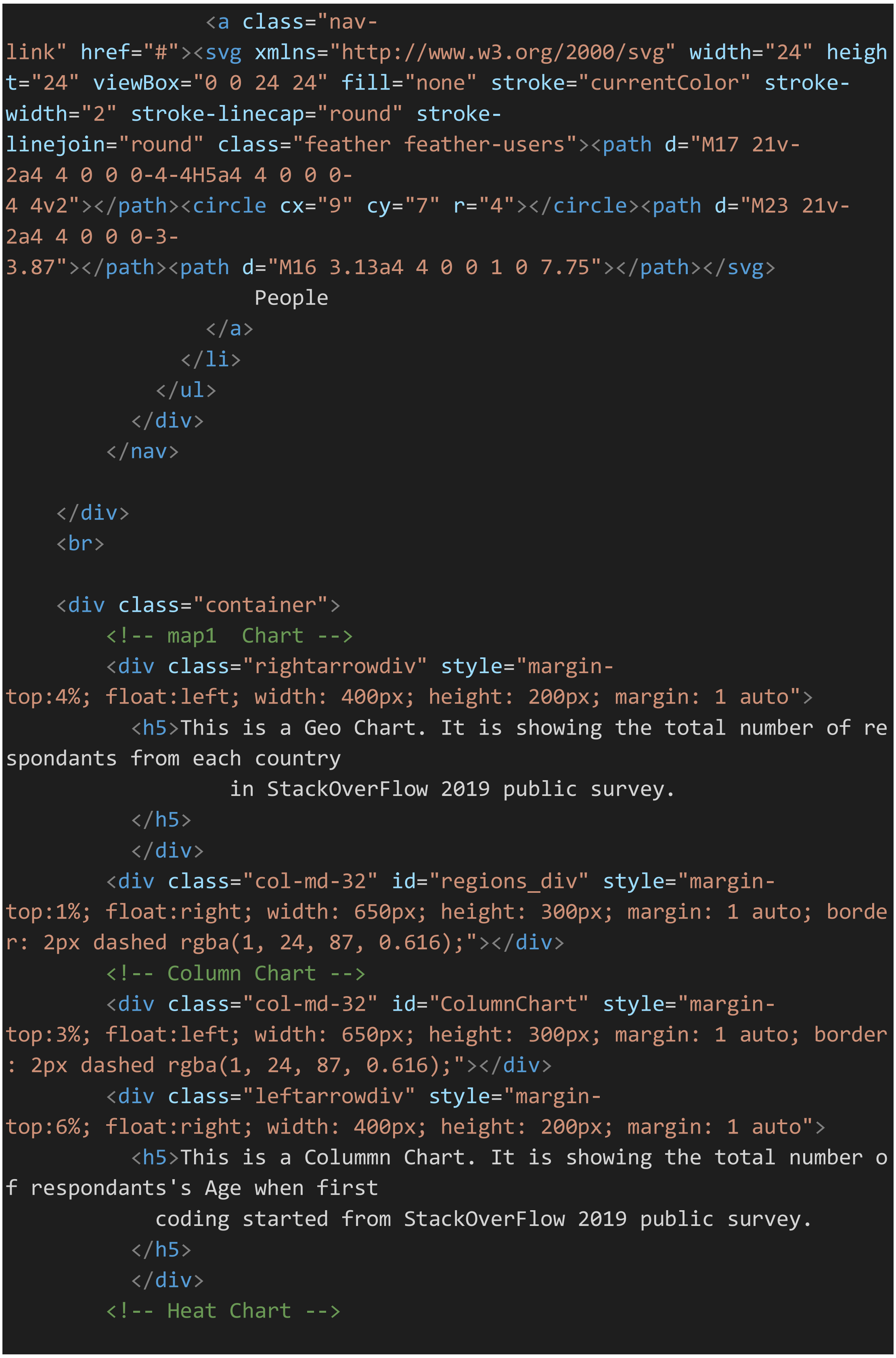
reports.html

#---#

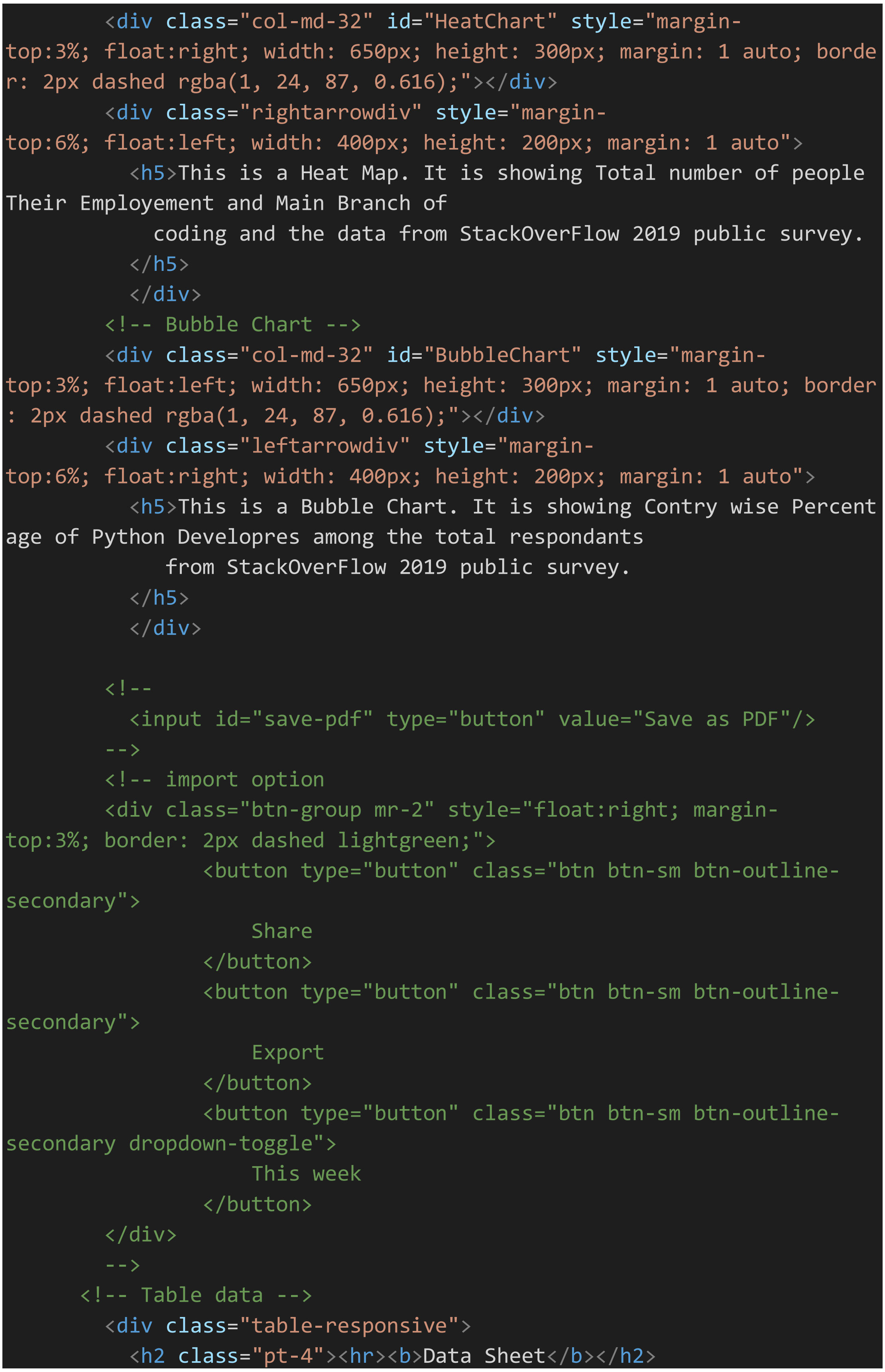
```
<!doctype html>
<html lang="en">
  <head>
    <title>Reports | DyGraphs</title>
    <meta charset="utf-8" />

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
    <script src="https://files.codepedia.info/files/uploads/iScripts/html2canvas.js"></script>
    <!-- Bootstrap core CSS -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/css/bootstrap.min.css" integrity="sha384-GJzZqFGwb1QTTN6wy59ffF1BuGJpLSa9DkKMp0DgiMDm4iYMj70gZWKYbI706tWS" crossorigin="anonymous">

  </head>
  <body>

    <!-- Top Bar -->
    <div class="container" style="background-color: black; margin-top:1%">
        <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
          <a class="navbar-brand" href="/basep"><b>DyGraphs | Reporting Tool</b></a>
          <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
```



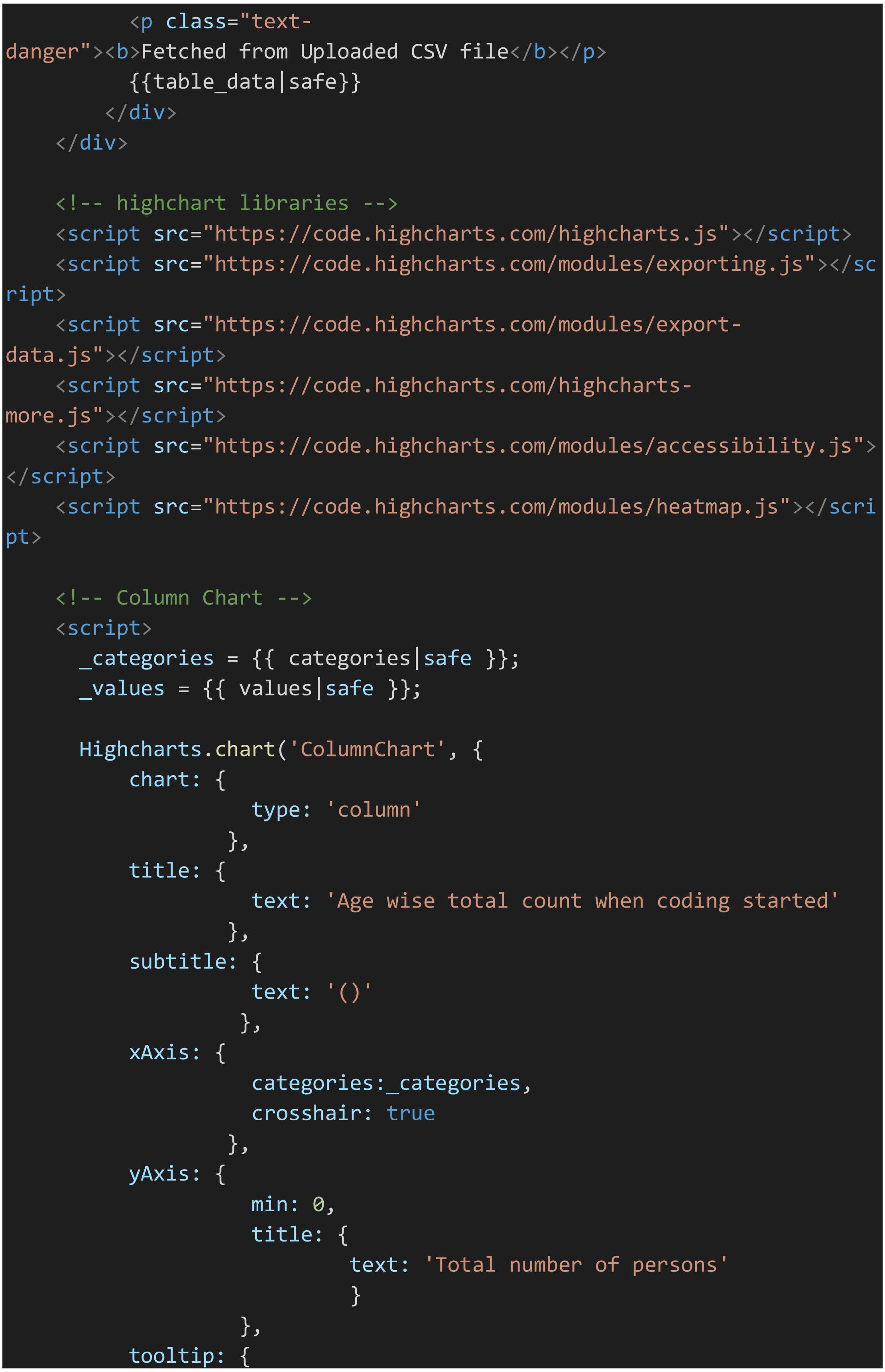
```
      <span class="navbar-toggler-icon"></span>
     </button>
     <div class="collapse navbar-collapse" id="navbarNav">

     </div>
     <div id="navbar"  style="float:right;">
      <ul class="navbar-nav">
       <li class="nav-item active">
        <a class="nav-
link" href="/basep"><svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" strok
e-width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-home"><path d="M3 9l9-
7 9 7v11a2 2 0 0 1-2 2H5a2 2 0 0 1-2-
2z"></path><polyline points="9 22 9 12 15 12 15 22"></polyline></svg>
          Home <span class="sr-only">(current)</span>
        </a>
       </li>
       <li class="nav-item active">
        <a class="nav-
link" href="/basep/reports"><svg xmlns="http://www.w3.org/2000/svg" wid
th="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColo
r" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-bar-chart-
2"><line x1="18" y1="20" x2="18" y2="10"></line><line x1="12" y1="20" x
2="12" y2="4"></line><line x1="6" y1="20" x2="6" y2="14"></line></svg>
          Reports
        </a>
       </li>
       <li class="nav-item active">
        <a class="nav-
link" href="/basep/design"><svg xmlns="http://www.w3.org/2000/svg" widt
h="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor
" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-
sliders"><line x1="4" y1="21" x2="4" y2="14"></line><line x1="4" y1="10
" x2="4" y2="3"></line><line x1="12" y1="21" x2="12" y2="12"></line><li
ne x1="12" y1="8" x2="12" y2="3"></line><line x1="20" y1="21" x2="20" y
2="16"></line><line x1="20" y1="12" x2="20" y2="3"></line><line x1="1"
y1="14" x2="7" y2="14"></line><line x1="9" y1="8" x2="15" y2="8"></line
><line x1="17" y1="16" x2="23" y2="16"></line></svg>
          Design
        </a>
       </li>
       <li class="nav-item active">
```



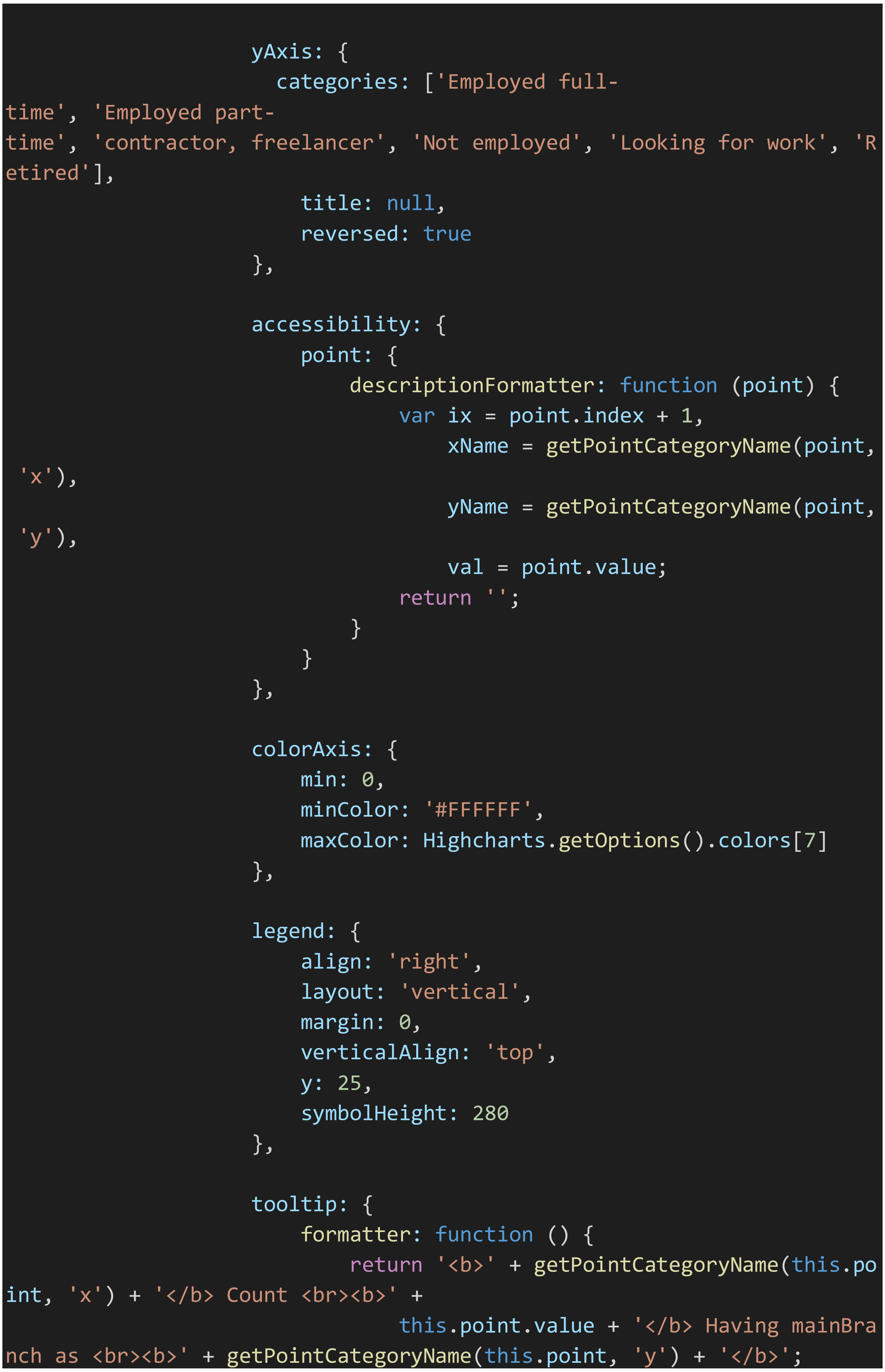
```
                <a class="nav-
link" href="#"><svg xmlns="http://www.w3.org/2000/svg" width="24" heigh
t="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-
width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-users"><path d="M17 21v-
2a4 4 0 0 0-4-4H5a4 4 0 0 0-
4 4v2"></path><circle cx="9" cy="7" r="4"></circle><path d="M23 21v-
2a4 4 0 0 0-3-
3.87"></path><path d="M16 3.13a4 4 0 0 1 0 7.75"></path></svg>
                    People
                </a>
              </li>
            </ul>
          </div>
        </nav>

    </div>
    <br>

    <div class="container">
        <!-- map1  Chart -->
        <div class="rightarrowdiv" style="margin-
top:4%; float:left; width: 400px; height: 200px; margin: 1 auto">
          <h5>This is a Geo Chart. It is showing the total number of re
spondants from each country
                  in StackOverFlow 2019 public survey.
          </h5>
          </div>
        <div class="col-md-32" id="regions_div" style="margin-
top:1%; float:right; width: 650px; height: 300px; margin: 1 auto; borde
r: 2px dashed rgba(1, 24, 87, 0.616);"></div>
        <!-- Column Chart -->
        <div class="col-md-32" id="ColumnChart" style="margin-
top:3%; float:left; width: 650px; height: 300px; margin: 1 auto; border
: 2px dashed rgba(1, 24, 87, 0.616);"></div>
        <div class="leftarrowdiv" style="margin-
top:6%; float:right; width: 400px; height: 200px; margin: 1 auto">
          <h5>This is a Colummn Chart. It is showing the total number o
f respondants's Age when first
            coding started from StackOverFlow 2019 public survey.
          </h5>
          </div>
        <!-- Heat Chart -->
```



```
    <div class="col-md-32" id="HeatChart" style="margin-
top:3%; float:right; width: 650px; height: 300px; margin: 1 auto; borde
r: 2px dashed rgba(1, 24, 87, 0.616);"></div>
    <div class="rightarrowdiv" style="margin-
top:6%; float:left; width: 400px; height: 200px; margin: 1 auto">
     <h5>This is a Heat Map. It is showing Total number of people
Their Employement and Main Branch of
      coding and the data from StackOverFlow 2019 public survey.
     </h5>
     </div>
    <!-- Bubble Chart -->
    <div class="col-md-32" id="BubbleChart" style="margin-
top:3%; float:left; width: 650px; height: 300px; margin: 1 auto; border
: 2px dashed rgba(1, 24, 87, 0.616);"></div>
    <div class="leftarrowdiv" style="margin-
top:6%; float:right; width: 400px; height: 200px; margin: 1 auto">
     <h5>This is a Bubble Chart. It is showing Contry wise Percent
age of Python Developres among the total respondants
       from StackOverFlow 2019 public survey.
     </h5>
     </div>

    <!--
     <input id="save-pdf" type="button" value="Save as PDF"/>
    -->
    <!-- import option
    <div class="btn-group mr-2" style="float:right; margin-
top:3%; border: 2px dashed lightgreen;">
        <button type="button" class="btn btn-sm btn-outline-
secondary">
          Share
        </button>
        <button type="button" class="btn btn-sm btn-outline-
secondary">
          Export
        </button>
        <button type="button" class="btn btn-sm btn-outline-
secondary dropdown-toggle">
          This week
        </button>
    </div>
    -->
   <!-- Table data -->
    <div class="table-responsive">
     <h2 class="pt-4"><hr><b>Data Sheet</b></h2>
```



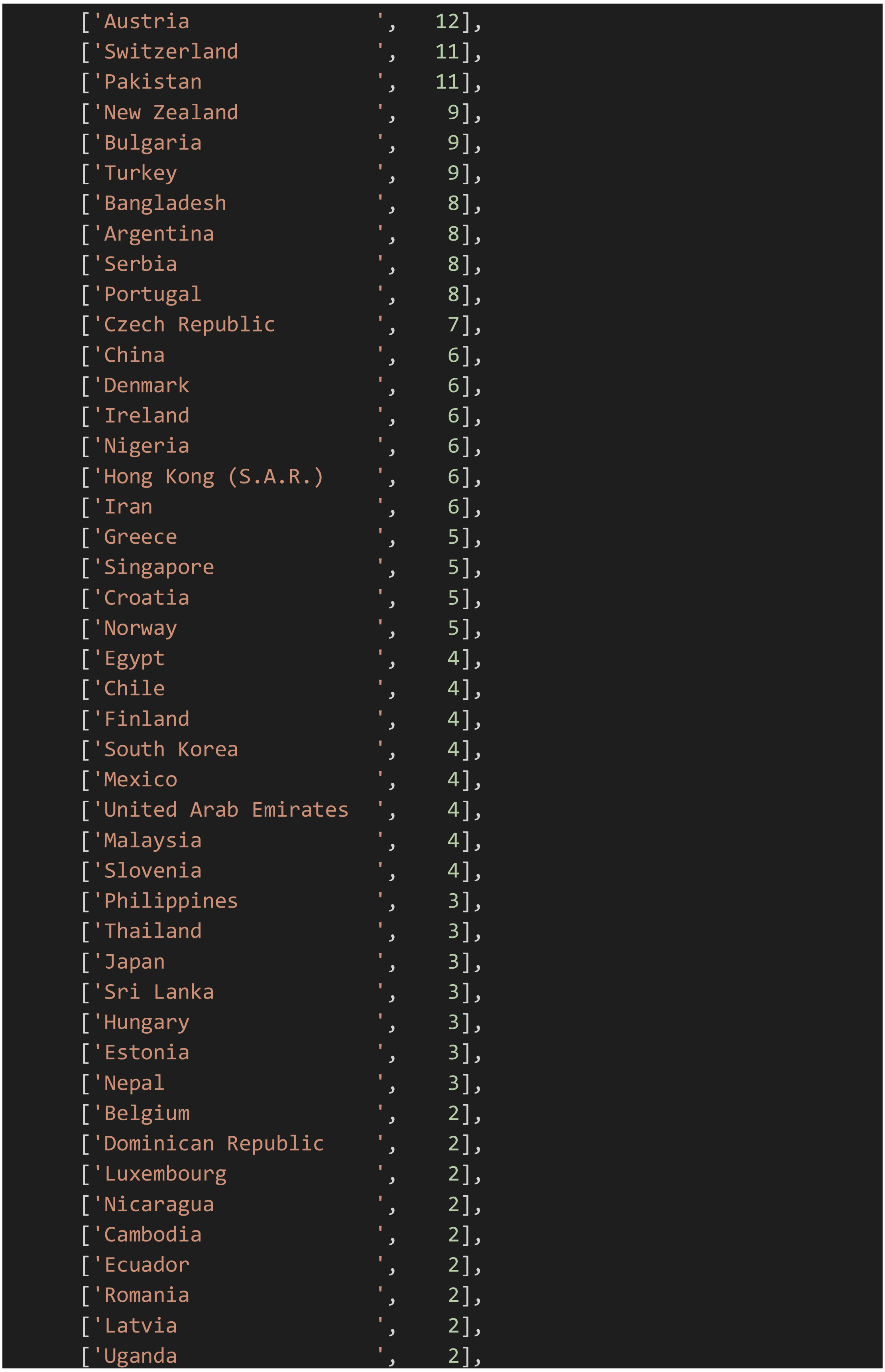
```
          <p class="text-
danger"><b>Fetched from Uploaded CSV file</b></p>
          {{table_data|safe}}
        </div>
    </div>

    <!-- highchart libraries -->
    <script src="https://code.highcharts.com/highcharts.js"></script>
    <script src="https://code.highcharts.com/modules/exporting.js"></sc
ript>
    <script src="https://code.highcharts.com/modules/export-
data.js"></script>
    <script src="https://code.highcharts.com/highcharts-
more.js"></script>
    <script src="https://code.highcharts.com/modules/accessibility.js">
</script>
    <script src="https://code.highcharts.com/modules/heatmap.js"></scri
pt>

    <!-- Column Chart -->
    <script>
      _categories = {{ categories|safe }};
      _values = {{ values|safe }};

      Highcharts.chart('ColumnChart', {
          chart: {
                    type: 'column'
                  },
          title: {
                    text: 'Age wise total count when coding started'
                  },
          subtitle: {
                    text: '()'
                   },
          xAxis: {
                    categories:_categories,
                    crosshair: true
                  },
          yAxis: {
                    min: 0,
                    title: {
                            text: 'Total number of persons'
                            }
                   },
          tooltip: {
            headerFormat: '<span style="font-
size:20px">{series.name} {point.key}</span><table>',
              pointFormat: '<tr><td style="padding:0"> is : </td>' +
                             '<td style="padding:0; font-
size:20px"><b>{point.y:.f} </b></td></tr>',
              footerFormat: '</table>',
              shared: true,
              useHTML: true
              },
    plotOptions: {
              column: {
                  pointPadding: 0.2,
                  borderWidth: 0
                        }
              },
          series: [{
                    name: 'Age when coding started',
                    data: _values,
                    color: '#808080'

          }]
      });
      </script>


      <!-- Bubble Chart -->
      <script>
        Highcharts.chart('BubbleChart', {

            chart: {
                type: 'bubble',
                plotBorderWidth: 1,
                zoomType: 'xy'
            },

            legend: {
                enabled: false
            },

            title: {
                text: 'Percent of Python Developers By Country'
            },

            subtitle: {
                text: '---------------------------------------'
            },

            accessibility: {

            },

            xAxis: {
                gridLineWidth: 1,
                title: {
                    text: 'Total number of respondants from Each Countr
y'
                },
                labels: {
                    format: '{value}'
                },

                accessibility: {
                    rangeDescription: ''
                }
            },

            yAxis: {
                startOnTick: false,
                endOnTick: false,
                title: {
                    text: 'Number of Respondents Knows Python Language'
                },
                labels: {
                    format: '{value}'
                },
                maxPadding: 0.2,
                accessibility: {
                    rangeDescription: ''
                }
            },

            tooltip: {
                useHTML: true,
                headerFormat: '<table>',
                pointFormat: '<tr><th colspan="2"><h3>{point.country}</
h3></th></tr>' +
                    '<tr><th>#Respondents:</th><td>{point.x}</td></tr>'
 +
                    '<tr><th>#KnowsPython:</th><td>{point.y}</td></tr>'
 +
                    '<tr><th>%KnowsPython:</th><td>{point.z}%</td></tr>
',
                footerFormat: '</table>',
                followPointer: true
            },

            plotOptions: {
                series: {
                    dataLabels: {
                        enabled: true,
                        format: '{point.name}'
                    }
                }
            },

            series: [{
                data: [
                    { x: 253, y: 131,  z: 51.778656,  name: 'US' ,  cou
ntry: 'United States'  },
                    { x: 88,  y: 37,   z: 42.045455,  name: 'Ind',  cou
ntry: 'India'          },
                    { x: 63,  y: 29,   z: 46.031746,  name: 'UK',   cou
ntry: 'United Kingdom'   },
                    { x: 54,  y: 26,   z: 48.148148,  name: 'DE',   cou
ntry: 'Germany'        },
                    { x: 43,  y: 19,   z: 44.186047,  name: 'CAN',  cou
ntry: 'Canada'     },
                    { x: 35,  y: 15,   z: 42.857143,  name: 'FR',   cou
ntry: 'France'     },
                    { x: 31,  y: 13,   z: 41.935484,  name: 'NL',   cou
ntry: 'Netherlands'      },
                    { x: 28,  y: 13,   z: 46.428571,  name: 'AUS',  cou
ntry: 'Australia'      },
                    { x: 24,  y: 8,  z: 33.333333,  name: 'RU',   count
ry: 'Russian'      },
                    { x: 19,  y: 5,  z: 26.315789,  name: 'SPN',  count
ry: 'Spain'            },
                    { x: 18,  y: 7,  z: 38.888889,  name: 'BR',   count
ry: 'Brazil'     },
                    { x: 18,  y: 10,   z: 55.555556,  name: 'PL',   cou
ntry: 'Poland'     },
                    { x: 17,  y: 10,   z: 58.823529,  name: 'IT',   cou
ntry: 'Italy'      },
                    { x: 16,  y: 8,  z: 50.000000,  name: 'SW',   count
ry: 'Sweden'     },
                    { x: 14,  y: 2,  z: 14.285714,  name: 'IS',   count
ry: 'Israel'     },
                    { x: 12,  y: 3,  z: 25.000000,  name: 'UKR',  count
ry: 'Ukraine'      },
                    { x: 12,  y: 3,  z: 25.000000,  name: 'SS',   count
ry: 'South Africa'     },
                    { x: 12,  y: 5,  z: 41.666667,  name: 'ATR',  count
ry: 'Austria'        },
                    { x: 11,  y: 5,  z: 45.454545,  name: 'PK',   count
ry: 'Pakistan'     },
                    { x: 11,  y: 2,  z: 18.181818,  name: 'SLND', count
ry: 'Switzerland'      }
                      ]
                  }]

                  });
      </script>

    <!-- Heat Chart -->
    <script>
          function getPointCategoryName(point, dimension) {
            var series = point.series,
                isY = dimension === 'y',
                axis = series[isY ? 'yAxis' : 'xAxis'];
            return axis.categories[point[isY ? 'y' : 'x']];
                }

                Highcharts.chart('HeatChart', {

                    chart: {
                        type: 'heatmap',
                        marginTop: 40,
                        marginBottom: 80,
                        plotBorderWidth: 1
                    },


                    title: {
                        text: 'Employement and MainBranch'
                    },

                    xAxis: {
                      categories: ['Developer', 'Not a developer', 'Was
 a developer', 'Student', 'As a hobby']
                    },
```



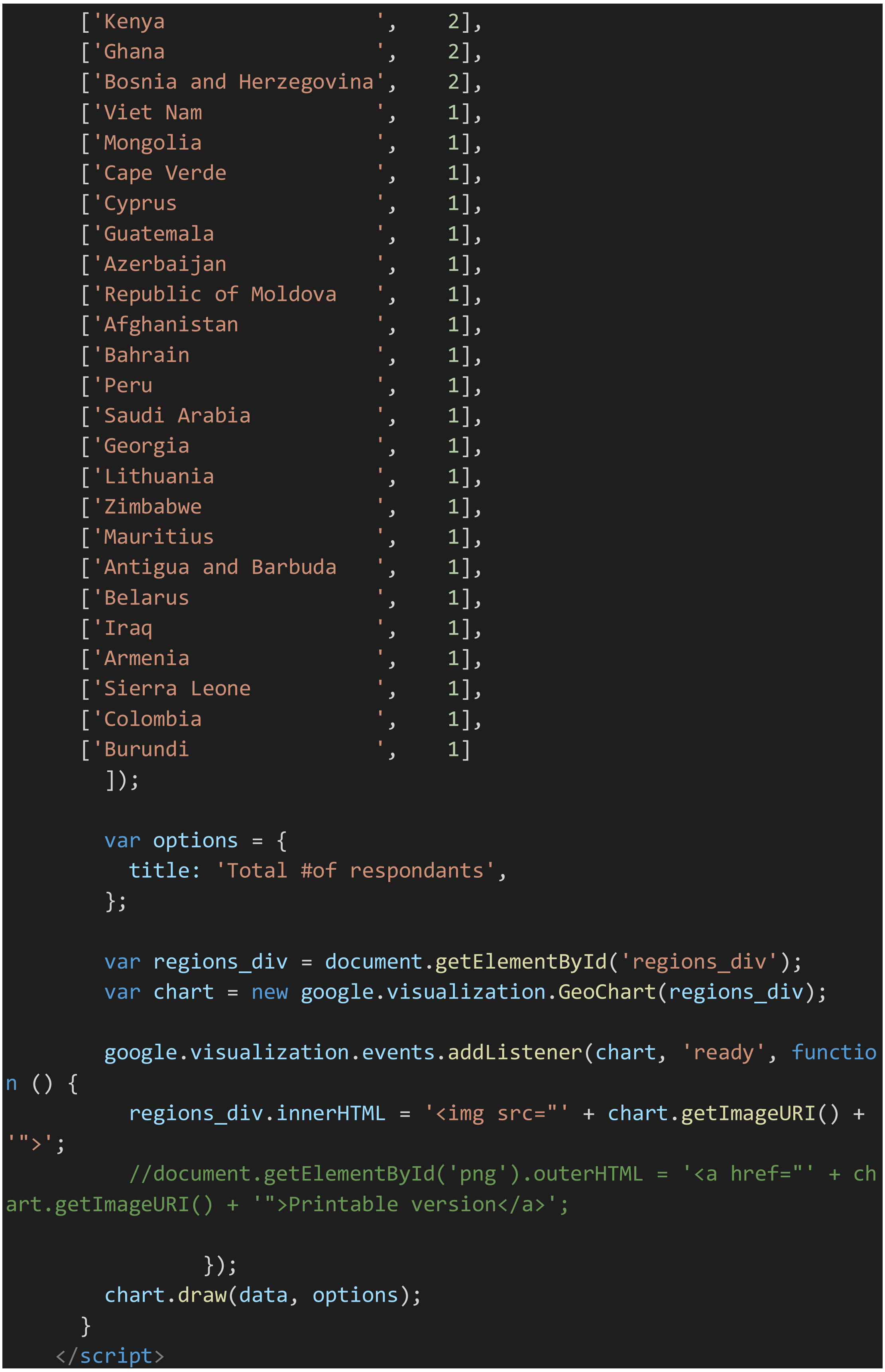
```
                    yAxis: {
                      categories: ['Employed full-
time', 'Employed part-
time', 'contractor, freelancer', 'Not employed', 'Looking for work', 'R
etired'],
                        title: null,
                        reversed: true
                    },

                    accessibility: {
                        point: {
                            descriptionFormatter: function (point) {
                                var ix = point.index + 1,
                                    xName = getPointCategoryName(point,
 'x'),

                                    yName = getPointCategoryName(point,
 'y'),

                                    val = point.value;
                                return '';
                            }
                        }
                    },

                    colorAxis: {
                        min: 0,
                        minColor: '#FFFFFF',
                        maxColor: Highcharts.getOptions().colors[7]
                    },

                    legend: {
                        align: 'right',
                        layout: 'vertical',
                        margin: 0,
                        verticalAlign: 'top',
                        y: 25,
                        symbolHeight: 280
                    },

                    tooltip: {
                        formatter: function () {
                            return '<b>' + getPointCategoryName(this.po
int, 'x') + '</b> Count <br><b>' +
                                this.point.value + '</b> Having mainBra
nch as <br><b>' + getPointCategoryName(this.point, 'y') + '</b>';
```

```
                        }
                    },

                    series: [{
                        name: 'Sales per employee',
                        borderWidth: 1,
                        data: [[0,0,65], [0,1,32], [0,2,51], [0,3,1],
[0,4,10], [0,5,0], [1,0,5], [1,1,57], [1,2,9], [1,3,0], [1,4,5], [1,5,0
], [2,0,10], [2,1,1], [2,2,2], [2,3,1], [2,4,2],  [2,5,1], [3,0,9], [3,
1,24], [3,2,8], [3,3,24], [3,4,36], [3,5,0], [4,0,3],  [4,1,4], [4,2,6]
, [4,3,7], [4,4,6], [4,5,45]],
                        dataLabels: {
                            enabled: true,
                            color: '#000000'
                        }
                    }],

                    responsive: {
                        rules: [{
                            condition: {
                                maxWidth: 500
                            },
                            chartOptions: {
                                yAxis: {
                                    labels: {
                                        formatter: function () {
                                            return this.value.charAt(0)
;
                                        }
                                    }
                                }
                            }
                        }]
                    }

                });
    </script>

    <script src="https://code.jquery.com/jquery-
3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" cross
origin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.
6/umd/popper.min.js" integrity="sha384-
```

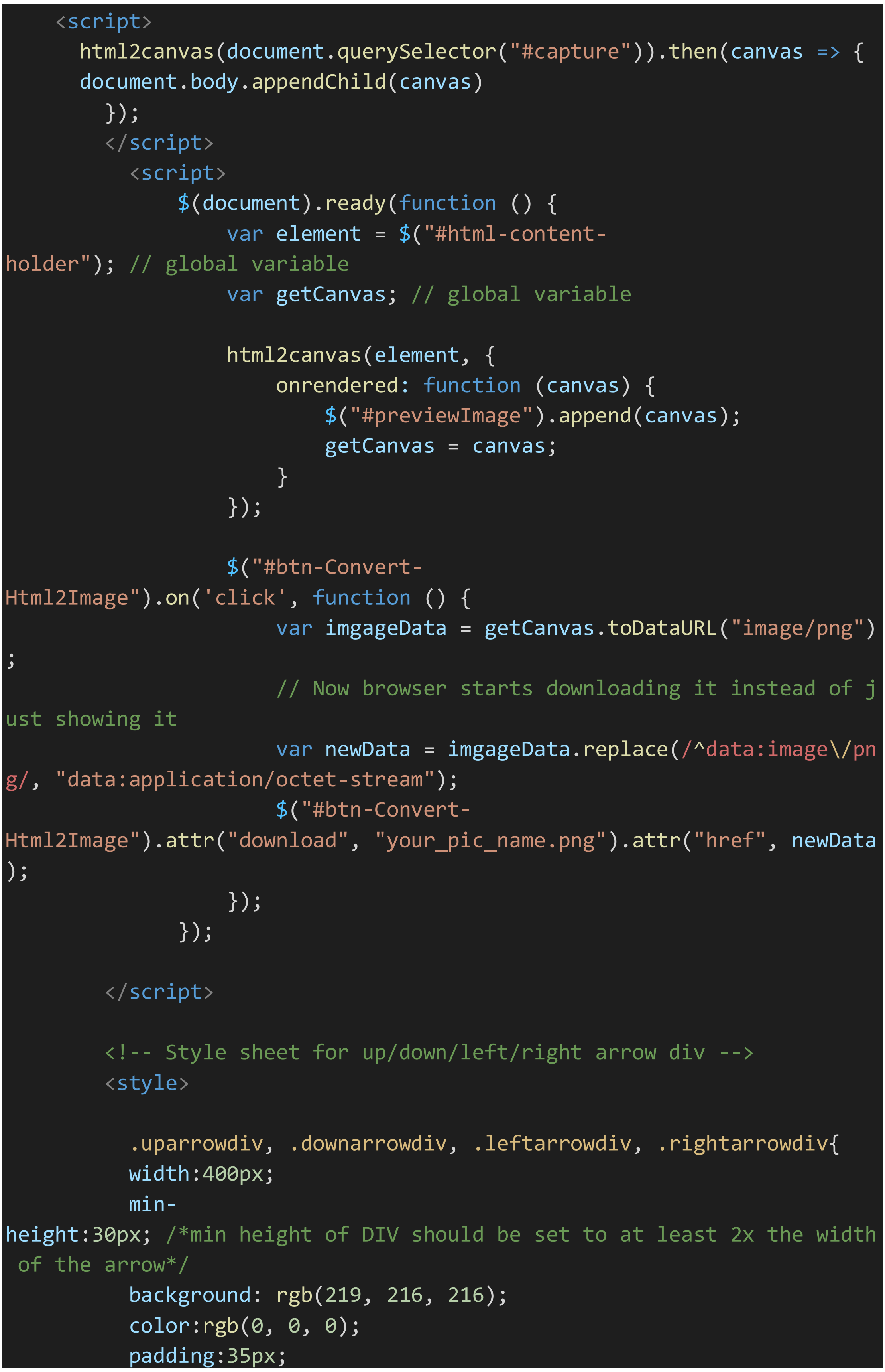
```
wHAiFfRlMFy6i5SRaxvfOCifBUQy1xHdJ/yoi7FRNXMRBu5WHdZYu1hA6ZOblgut" cross
origin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/js/
bootstrap.min.js" integrity="sha384-
B0UglyR+jN6CkvvICOB2joaf5I4l3gm9GU6Hc1og6Ls7i6U/mkkaduKaBhlAXv9k" cross
origin="anonymous"></script>


    <!-- Google map library Chart -->
    <script type="text/javascript" src="https://www.gstatic.com/charts/
loader.js"></script>
    <script type="text/javascript" src="https://www.google.com/jsapi?au
toload={'modules':[{'name':'visualization','version':'1.0','packages':[
'geochart']},{'name':'visualization','version':'1.0','packages':['contr
ols']}]}"></script>
    <script type="text/javascript">
      google.charts.load('current', {
        'packages':['geochart'],

        // See: https://developers.google.com/chart/interactive/docs/ba
sic_load_libs#load-settings
        'mapsApiKey': 'AIzaSyD-9tSrke72PouQMnMX-a7eZSW0jkFMBWY'
      });
      google.charts.setOnLoadCallback(drawRegionsMap);

      function drawRegionsMap() {
        var data = google.visualization.arrayToDataTable([
      ['Country', '#Respondants'],
      ['United States          ',  253],
      ['India                  ',   88],
      ['United Kingdom         ',   63],
      ['Germany                ',   54],
      ['Canada                 ',   43],
      ['France                 ',   35],
      ['Netherlands            ',   31],
      ['Australia              ',   28],
      ['Russian Federation     ',   24],
      ['Spain                  ',   19],
      ['Poland                 ',   18],
      ['Brazil                 ',   18],
      ['Italy                  ',   17],
      ['Sweden                 ',   16],
      ['Israel                 ',   14],
      ['Ukraine                ',   12],
      ['South Africa           ',   12],
```



```
    ['Austria               ',   12],
    ['Switzerland           ',   11],
    ['Pakistan              ',   11],
    ['New Zealand           ',    9],
    ['Bulgaria              ',    9],
    ['Turkey                ',    9],
    ['Bangladesh            ',    8],
    ['Argentina             ',    8],
    ['Serbia                ',    8],
    ['Portugal              ',    8],
    ['Czech Republic        ',    7],
    ['China                 ',    6],
    ['Denmark               ',    6],
    ['Ireland               ',    6],
    ['Nigeria               ',    6],
    ['Hong Kong (S.A.R.)    ',    6],
    ['Iran                  ',    6],
    ['Greece                ',    5],
    ['Singapore             ',    5],
    ['Croatia               ',    5],
    ['Norway                ',    5],
    ['Egypt                 ',    4],
    ['Chile                 ',    4],
    ['Finland               ',    4],
    ['South Korea           ',    4],
    ['Mexico                ',    4],
    ['United Arab Emirates  ',    4],
    ['Malaysia              ',    4],
    ['Slovenia              ',    4],
    ['Philippines           ',    3],
    ['Thailand              ',    3],
    ['Japan                 ',    3],
    ['Sri Lanka             ',    3],
    ['Hungary               ',    3],
    ['Estonia               ',    3],
    ['Nepal                 ',    3],
    ['Belgium               ',    2],
    ['Dominican Republic    ',    2],
    ['Luxembourg            ',    2],
    ['Nicaragua             ',    2],
    ['Cambodia              ',    2],
    ['Ecuador               ',    2],
    ['Romania               ',    2],
    ['Latvia                ',    2],
    ['Uganda                ',    2],
```



```
      ['Kenya                 ',    2],
      ['Ghana                 ',    2],
      ['Bosnia and Herzegovina',    2],
      ['Viet Nam              ',    1],
      ['Mongolia              ',    1],
      ['Cape Verde            ',    1],
      ['Cyprus                ',    1],
      ['Guatemala             ',    1],
      ['Azerbaijan            ',    1],
      ['Republic of Moldova   ',    1],
      ['Afghanistan           ',    1],
      ['Bahrain               ',    1],
      ['Peru                  ',    1],
      ['Saudi Arabia          ',    1],
      ['Georgia               ',    1],
      ['Lithuania             ',    1],
      ['Zimbabwe              ',    1],
      ['Mauritius             ',    1],
      ['Antigua and Barbuda   ',    1],
      ['Belarus               ',    1],
      ['Iraq                  ',    1],
      ['Armenia               ',    1],
      ['Sierra Leone          ',    1],
      ['Colombia              ',    1],
      ['Burundi               ',    1]
        ]);

        var options = {
          title: 'Total #of respondants',
        };

        var regions_div = document.getElementById('regions_div');
        var chart = new google.visualization.GeoChart(regions_div);

        google.visualization.events.addListener(chart, 'ready', function () {
          regions_div.innerHTML = '<img src="' + chart.getImageURI() + '">';
          //document.getElementById('png').outerHTML = '<a href="' + chart.getImageURI() + '">Printable version</a>';

                });
        chart.draw(data, options);
      }
    </script>
```



```
  <script>
   html2canvas(document.querySelector("#capture")).then(canvas => {
   document.body.appendChild(canvas)
    });
    </script>
     <script>
       $(document).ready(function () {
         var element = $("#html-content-
holder"); // global variable
         var getCanvas; // global variable

         html2canvas(element, {
           onrendered: function (canvas) {
             $("#previewImage").append(canvas);
             getCanvas = canvas;
           }
         });

         $("#btn-Convert-
Html2Image").on('click', function () {
           var imgageData = getCanvas.toDataURL("image/png")
;
           // Now browser starts downloading it instead of j
ust showing it
           var newData = imgageData.replace(/^data:image\/pn
g/, "data:application/octet-stream");
           $("#btn-Convert-
Html2Image").attr("download", "your_pic_name.png").attr("href", newData
);
         });
       });

    </script>

    <!-- Style sheet for up/down/left/right arrow div -->
    <style>

     .uparrowdiv, .downarrowdiv, .leftarrowdiv, .rightarrowdiv{
     width:400px;
     min-
height:30px; /*min height of DIV should be set to at least 2x the width
 of the arrow*/
     background: rgb(219, 216, 216);
     color:rgb(0, 0, 0);
     padding:35px;
```

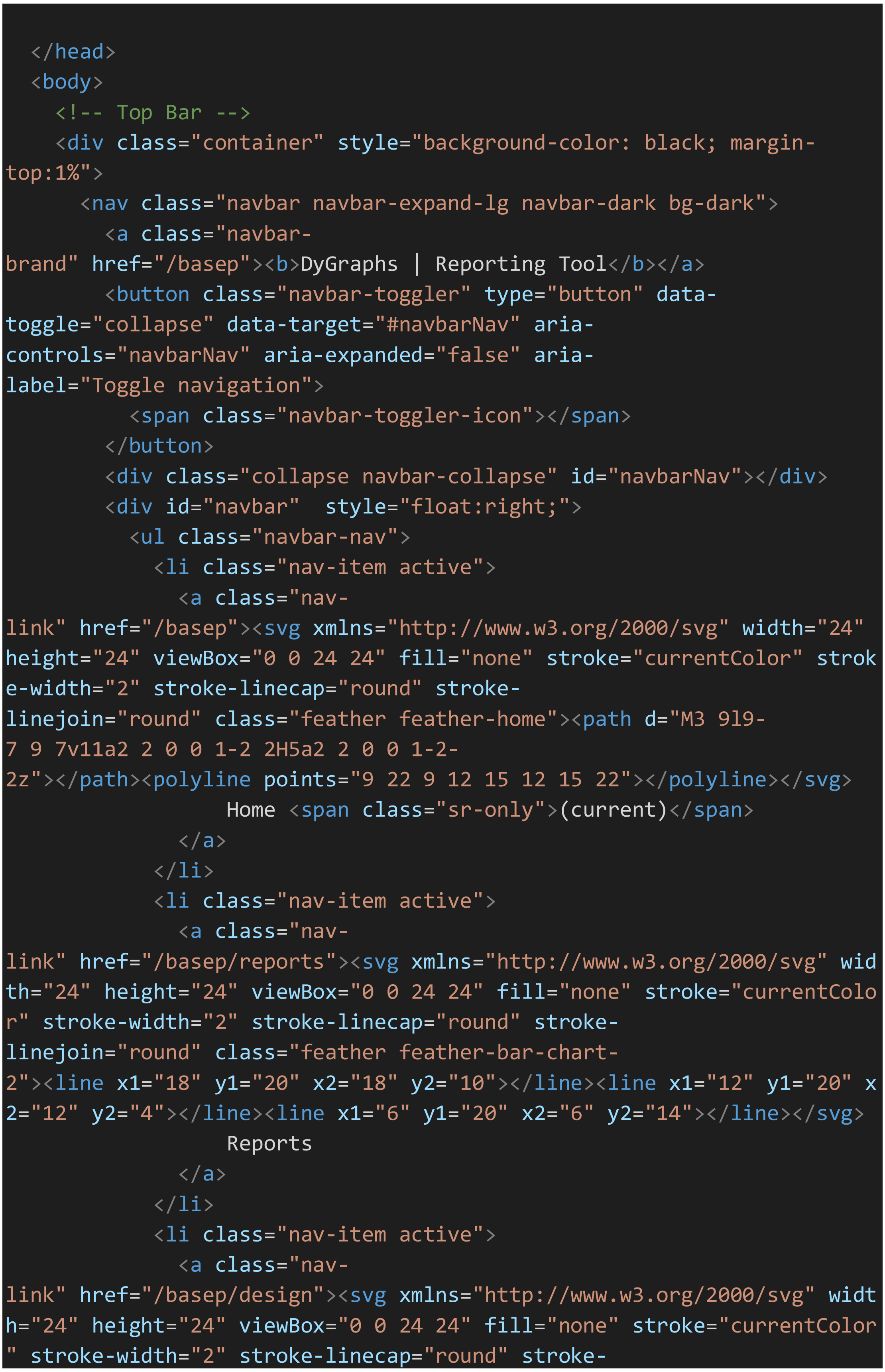
```
          position:relative;
          word-wrap:break-word;
          -moz-border-radius:5px; /*add some nice CSS3 round corners*/
          -webkit-border-radius:5px;
          border-radius:5px;
          margin-bottom:2em;
          }

          .uparrowdiv:after{ /*arrow added to uparrowdiv DIV*/
          content:'';
          display:block;
          position:absolute;
          top:-20px; /*should be set to -border-width x 2 */
          left:30px;
          width:0;
          height:0;
          border-
color: transparent transparent black transparent; /*border color should
 be same as div div background color*/
          border-style: solid;
          border-width: 10px;

          }

          .downarrowdiv:after{ /*arrow added to downarrowdiv DIV*/
          content:'';
          display:block;
          position:absolute;
          top:100%; /*should be set to 100% */
          left:30px;
          width:0;
          height:0;
          border-
color: black transparent transparent transparent; /*border color should
 be same as div div background color*/
          border-style: solid;
          border-width: 10px;
          }

          .leftarrowdiv:after{ /*arrow added to leftarrowdiv DIV*/
          content:'';
          display:block;
          position:absolute;
          top:10px;
          left:-60px; /*should be set to -border-width x 2 */
          width:0;
          height:0;
          border-color: transparent rgb(219, 216, 216) transparent transparent; /*border color should be same as div div background color*/
          border-style: solid;
          border-width: 30px;
          }

          .rightarrowdiv:after{ /*arrow added to rightarrowdiv DIV*/
          content:'';
          display:block;
          position:absolute;
          top:10px;
          left:100%; /*should be set to 100% */
          width:0;
          height:0;
          border-color: transparent transparent transparent rgb(219, 216, 216); /*border color should be same as div div background color*/
          border-style: solid;
          border-width: 30px;
          }

          </style>

  </body>
</html>
```

#--#

Design.html

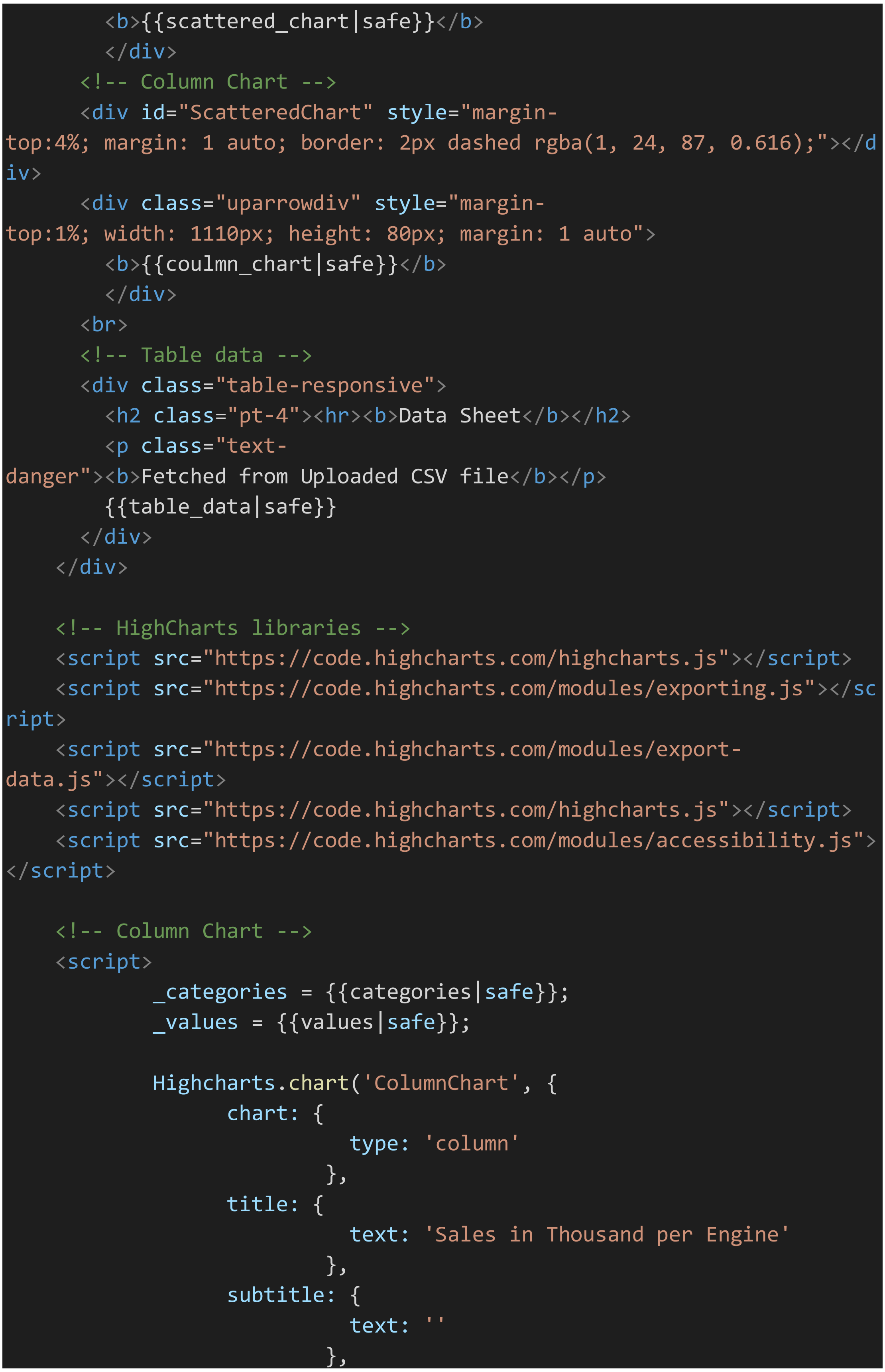
#--#

```
<!doctype html>
<html lang="en">
  <head>
    <title>Design | DyGraphs</title>

    <!-- Bootstrap core CSS -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/css/bootstrap.min.css" integrity="sha384-GJzZqFGwb1QTTN6wy59ffF1BuGJpLSa9DkKMp0DgiMDm4iYMj70gZWKYbI706tWS" crossorigin="anonymous">
```

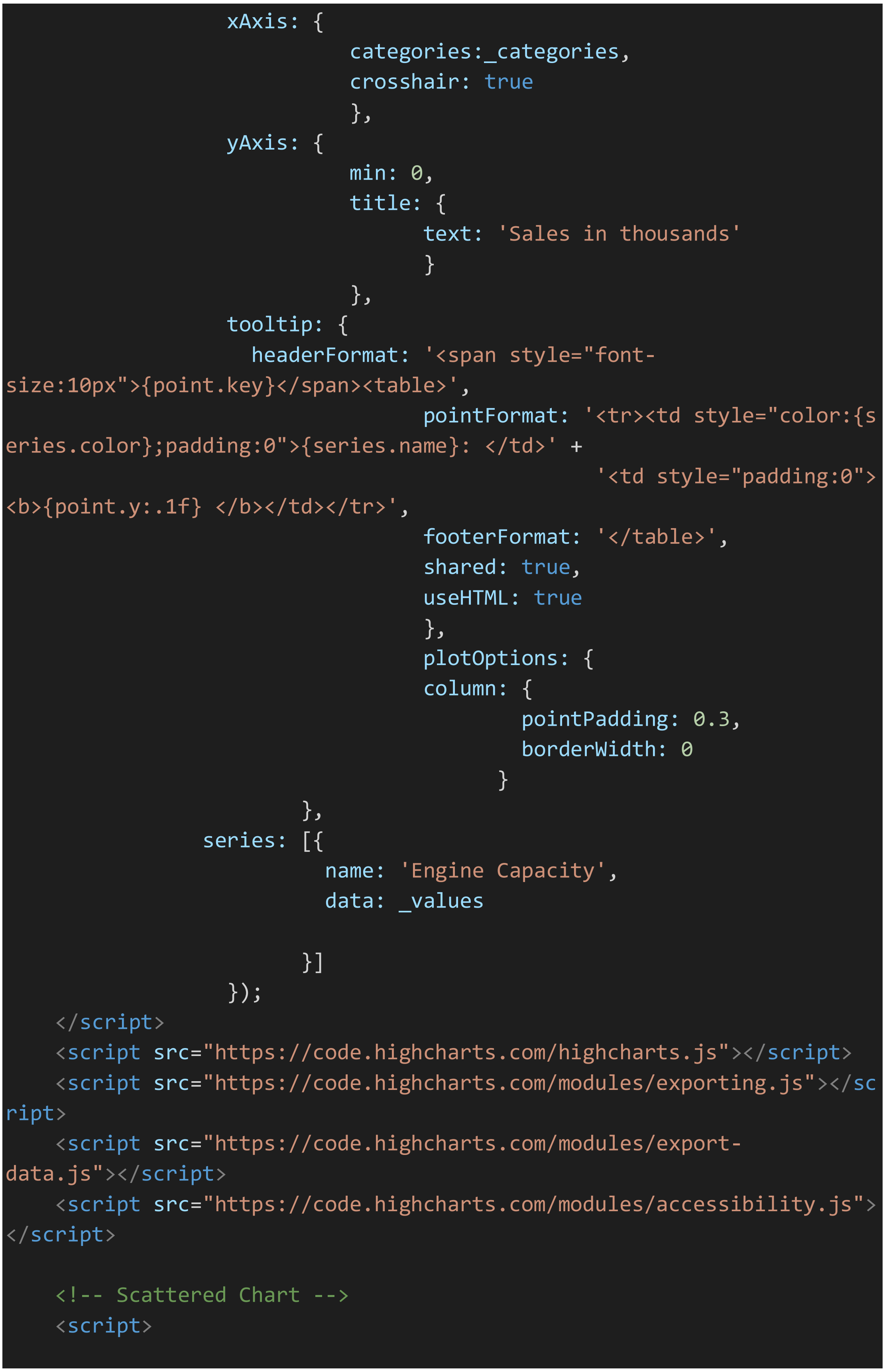


```
  </head>
  <body>
    <!-- Top Bar -->
    <div class="container" style="background-color: black; margin-
top:1%">
      <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <a class="navbar-
brand" href="/basep"><b>DyGraphs | Reporting Tool</b></a>
        <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarNav" aria-
controls="navbarNav" aria-expanded="false" aria-
label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarNav"></div>
        <div id="navbar"  style="float:right;">
          <ul class="navbar-nav">
            <li class="nav-item active">
              <a class="nav-
link" href="/basep"><svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" strok
e-width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-home"><path d="M3 9l9-
7 9 7v11a2 2 0 0 1-2 2H5a2 2 0 0 1-2-
2z"></path><polyline points="9 22 9 12 15 12 15 22"></polyline></svg>
                  Home <span class="sr-only">(current)</span>
              </a>
            </li>
            <li class="nav-item active">
              <a class="nav-
link" href="/basep/reports"><svg xmlns="http://www.w3.org/2000/svg" wid
th="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColo
r" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-bar-chart-
2"><line x1="18" y1="20" x2="18" y2="10"></line><line x1="12" y1="20" x
2="12" y2="4"></line><line x1="6" y1="20" x2="6" y2="14"></line></svg>
                  Reports
              </a>
            </li>
            <li class="nav-item active">
              <a class="nav-
link" href="/basep/design"><svg xmlns="http://www.w3.org/2000/svg" widt
h="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor
" stroke-width="2" stroke-linecap="round" stroke-
```

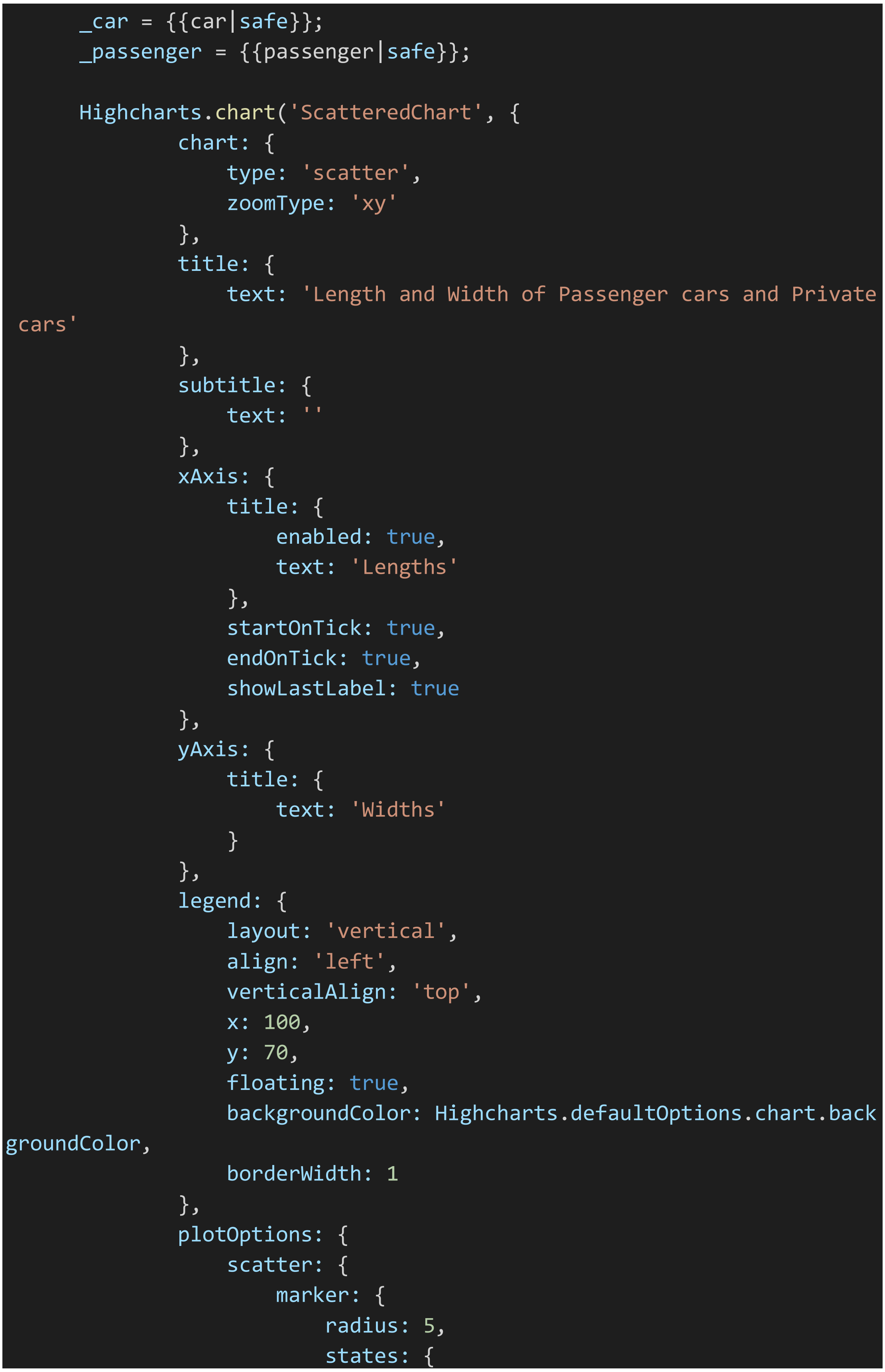
```
linejoin="round" class="feather feather-
sliders"><line x1="4" y1="21" x2="4" y2="14"></line><line x1="4" y1="10
" x2="4" y2="3"></line><line x1="12" y1="21" x2="12" y2="12"></line><li
ne x1="12" y1="8" x2="12" y2="3"></line><line x1="20" y1="21" x2="20" y
2="16"></line><line x1="20" y1="12" x2="20" y2="3"></line><line x1="1"
y1="14" x2="7" y2="14"></line><line x1="9" y1="8" x2="15" y2="8"></line
><line x1="17" y1="16" x2="23" y2="16"></line></svg>
                  Design
              </a>
            </li>
            <li class="nav-item active">
              <a class="nav-
link" href="#"><svg xmlns="http://www.w3.org/2000/svg" width="24" heigh
t="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-
width="2" stroke-linecap="round" stroke-
linejoin="round" class="feather feather-users"><path d="M17 21v-
2a4 4 0 0 0-4-4H5a4 4 0 0 0-
4 4v2"></path><circle cx="9" cy="7" r="4"></circle><path d="M23 21v-
2a4 4 0 0 0-3-
3.87"></path><path d="M16 3.13a4 4 0 0 1 0 7.75"></path></svg>
                  People
              </a>
            </li>
          </ul>
        </div>
      </nav>
    </div>
    <br>
    <div class="container">
      <div class="btn-group mr-2" style="float:right;">
        <form action = "" method = "get">
          <label for="Upload_File">                  Provide CSV file p
ath: </label>
          <input id="Upload_File" type="text" name="Upload_File">
          <input type="submit" value="submit">
        </form>
      </div>
      <br>
      <!-- Highcharts display -->
      <!-- Scattered Chart -->
      <div class="col-md-32" id="ColumnChart" style="margin-
top:4%; margin: 1 auto; border: 2px dashed rgba(1, 24, 87, 0.616);"></d
iv>
      <div class="uparrowdiv" style="margin-
top:1%; width: 1110px; height: 80px; margin: 1 auto">
```



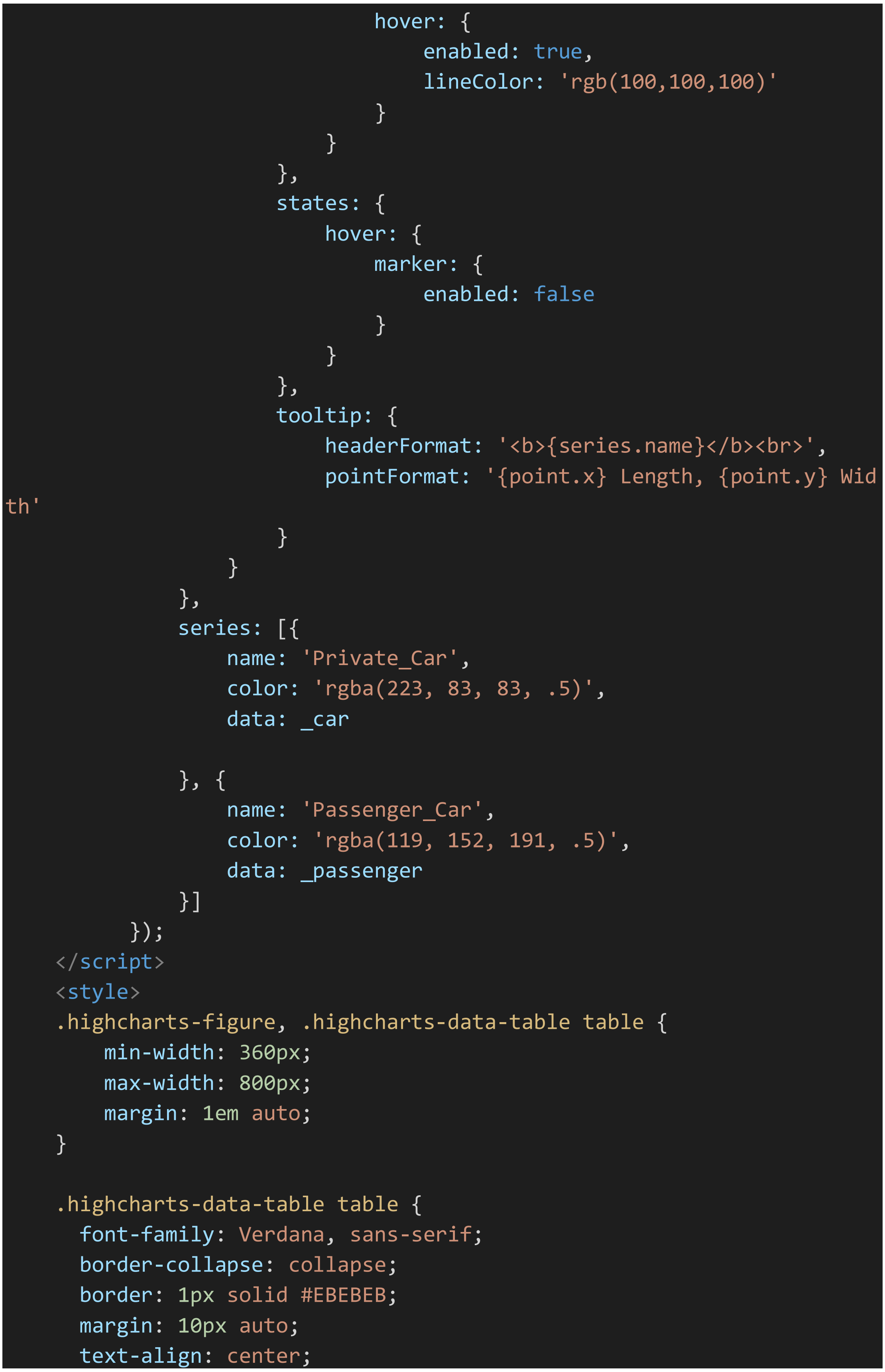
```
        <b>{{scattered_chart|safe}}</b>
        </div>
      <!-- Column Chart -->
      <div id="ScatteredChart" style="margin-
top:4%; margin: 1 auto; border: 2px dashed rgba(1, 24, 87, 0.616);"></d
iv>
      <div class="uparrowdiv" style="margin-
top:1%; width: 1110px; height: 80px; margin: 1 auto">
        <b>{{coulmn_chart|safe}}</b>
        </div>
      <br>
      <!-- Table data -->
      <div class="table-responsive">
        <h2 class="pt-4"><hr><b>Data Sheet</b></h2>
        <p class="text-
danger"><b>Fetched from Uploaded CSV file</b></p>
        {{table_data|safe}}
      </div>
    </div>

    <!-- HighCharts libraries -->
    <script src="https://code.highcharts.com/highcharts.js"></script>
    <script src="https://code.highcharts.com/modules/exporting.js"></sc
ript>
    <script src="https://code.highcharts.com/modules/export-
data.js"></script>
    <script src="https://code.highcharts.com/highcharts.js"></script>
    <script src="https://code.highcharts.com/modules/accessibility.js">
</script>

    <!-- Column Chart -->
    <script>
            _categories = {{categories|safe}};
            _values = {{values|safe}};

            Highcharts.chart('ColumnChart', {
                  chart: {
                            type: 'column'
                          },
                  title: {
                            text: 'Sales in Thousand per Engine'
                          },
                  subtitle: {
                            text: ''
                          },
```



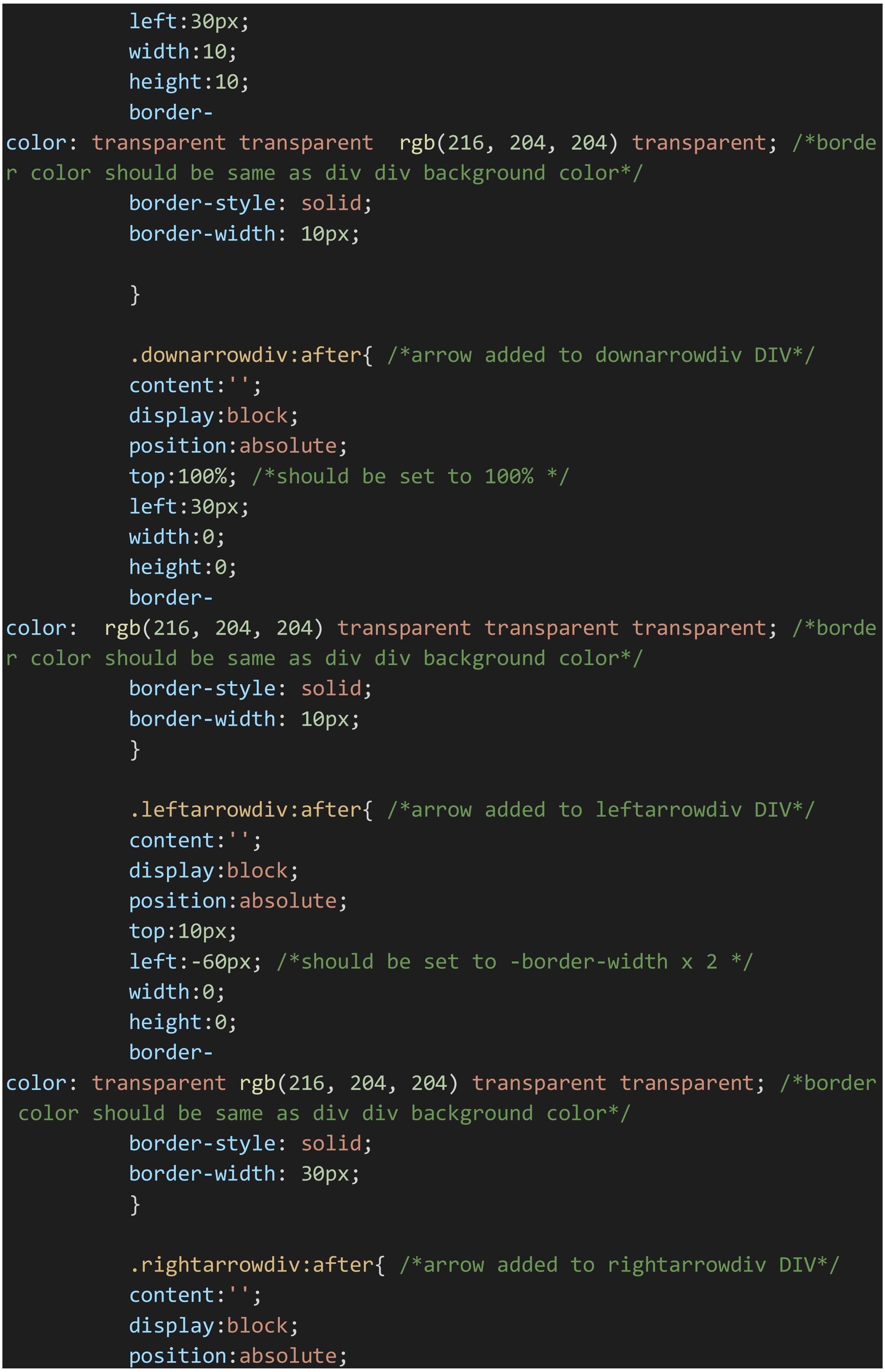
```
                  xAxis: {
                            categories:_categories,
                            crosshair: true
                            },
                  yAxis: {
                            min: 0,
                            title: {
                                  text: 'Sales in thousands'
                                  }
                            },
                  tooltip: {
                    headerFormat: '<span style="font-
size:10px">{point.key}</span><table>',
                                  pointFormat: '<tr><td style="color:{s
eries.color};padding:0">{series.name}: </td>' +
                                                '<td style="padding:0">
<b>{point.y:.1f} </b></td></tr>',
                                  footerFormat: '</table>',
                                  shared: true,
                                  useHTML: true
                                  },
                                  plotOptions: {
                                  column: {
                                          pointPadding: 0.3,
                                          borderWidth: 0
                                        }
                        },
                series: [{
                          name: 'Engine Capacity',
                          data: _values

                        }]
                  });
    </script>
    <script src="https://code.highcharts.com/highcharts.js"></script>
    <script src="https://code.highcharts.com/modules/exporting.js"></sc
ript>
    <script src="https://code.highcharts.com/modules/export-
data.js"></script>
    <script src="https://code.highcharts.com/modules/accessibility.js">
</script>

    <!-- Scattered Chart -->
    <script>
```



```
      _car = {{car|safe}};
      _passenger = {{passenger|safe}};

      Highcharts.chart('ScatteredChart', {
              chart: {
                  type: 'scatter',
                  zoomType: 'xy'
              },
              title: {
                  text: 'Length and Width of Passenger cars and Private
 cars'
              },
              subtitle: {
                  text: ''
              },
              xAxis: {
                  title: {
                      enabled: true,
                      text: 'Lengths'
                  },
                  startOnTick: true,
                  endOnTick: true,
                  showLastLabel: true
              },
              yAxis: {
                  title: {
                      text: 'Widths'
                  }
              },
              legend: {
                  layout: 'vertical',
                  align: 'left',
                  verticalAlign: 'top',
                  x: 100,
                  y: 70,
                  floating: true,
                  backgroundColor: Highcharts.defaultOptions.chart.back
groundColor,
                  borderWidth: 1
              },
              plotOptions: {
                  scatter: {
                      marker: {
                          radius: 5,
                          states: {
```



```
                              hover: {
                                  enabled: true,
                                  lineColor: 'rgb(100,100,100)'
                              }
                          }
                      },
                      states: {
                          hover: {
                              marker: {
                                  enabled: false
                              }
                          }
                      },
                      tooltip: {
                          headerFormat: '<b>{series.name}</b><br>',
                          pointFormat: '{point.x} Length, {point.y} Wid
th'
                      }
                  }
              },
              series: [{
                  name: 'Private_Car',
                  color: 'rgba(223, 83, 83, .5)',
                  data: _car

              }, {
                  name: 'Passenger_Car',
                  color: 'rgba(119, 152, 191, .5)',
                  data: _passenger
              }]
          });
    </script>
    <style>
    .highcharts-figure, .highcharts-data-table table {
        min-width: 360px;
        max-width: 800px;
        margin: 1em auto;
    }

    .highcharts-data-table table {
      font-family: Verdana, sans-serif;
      border-collapse: collapse;
      border: 1px solid #EBEBEB;
      margin: 10px auto;
      text-align: center;
      width: 100%;
      max-width: 500px;
    }
    .highcharts-data-table caption {
        padding: 1em 0;
        font-size: 1.2em;
        color: #555;
    }
    .highcharts-data-table th {
      font-weight: 600;
        padding: 0.5em;
    }
    .highcharts-data-table td, .highcharts-data-table th, .highcharts-
data-table caption {
        padding: 0.5em;
    }
    .highcharts-data-table thead tr, .highcharts-data-table tr:nth-
child(even) {
        background: #f8f8f8;
    }
    .highcharts-data-table tr:hover {
        background: #f1f7ff;
    }

    .uparrowdiv, .downarrowdiv, .leftarrowdiv, .rightarrowdiv{
          width:400px;
          min-
height:30px; /*min height of DIV should be set to at least 2x the width
 of the arrow*/
          background: rgb(219, 216, 216);
          color:rgb(0, 0, 0);
          padding:25px;
          position:relative;
          word-wrap:break-word;
          -moz-border-radius:5px; /*add some nice CSS3 round corners*/
          -webkit-border-radius:5px;
          border-radius:5px;
          margin-bottom:2em;
          }

          .uparrowdiv:after{ /*arrow added to uparrowdiv DIV*/
          content:'';
          display:block;
          position:absolute;
          top:-20px; /*should be set to -border-width x 2 */
```



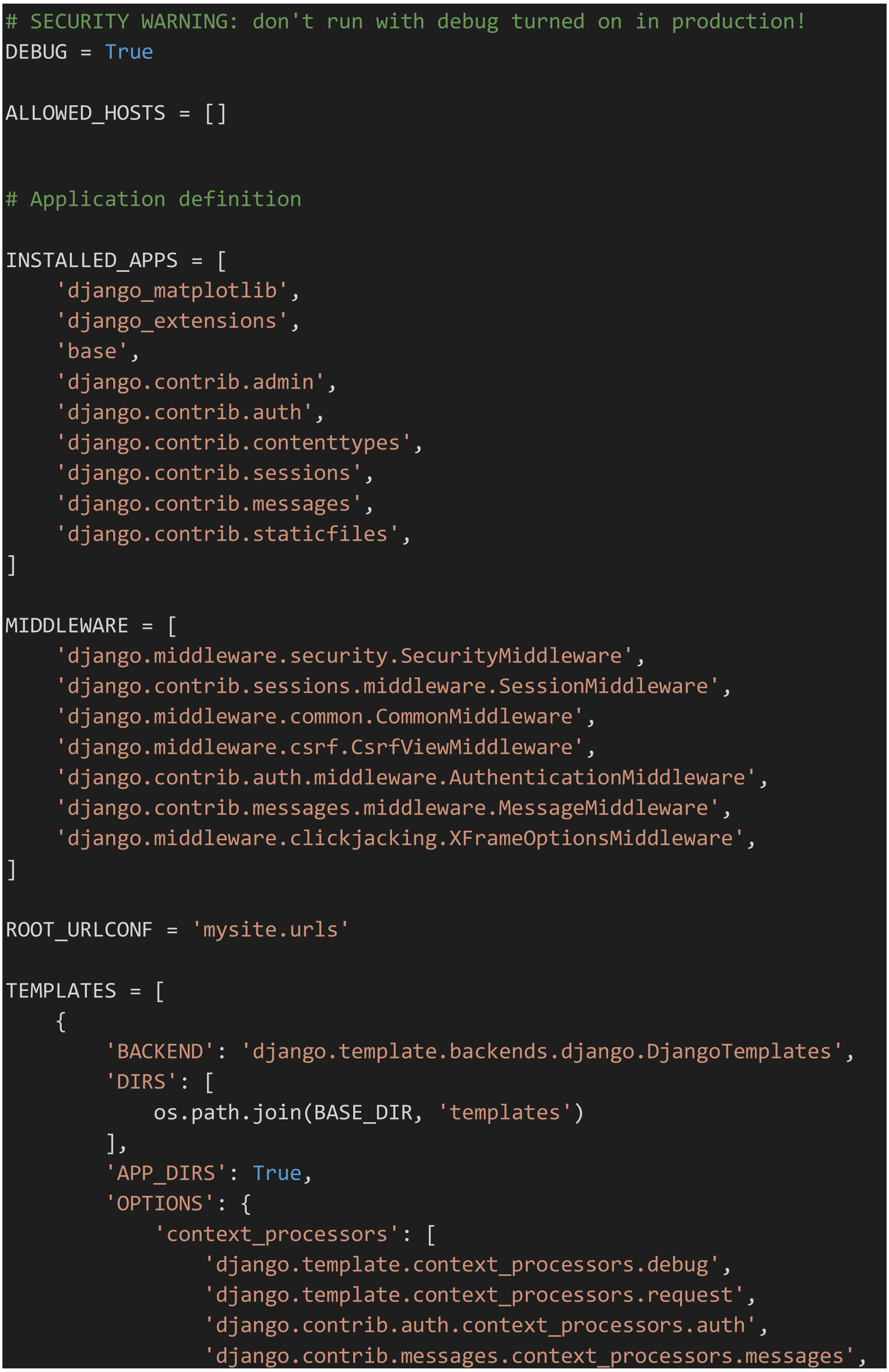
```
          left:30px;
          width:10;
          height:10;
          border-
color: transparent transparent  rgb(216, 204, 204) transparent; /*borde
r color should be same as div div background color*/
          border-style: solid;
          border-width: 10px;

          }

          .downarrowdiv:after{ /*arrow added to downarrowdiv DIV*/
          content:'';
          display:block;
          position:absolute;
          top:100%; /*should be set to 100% */
          left:30px;
          width:0;
          height:0;
          border-
color:  rgb(216, 204, 204) transparent transparent transparent; /*borde
r color should be same as div div background color*/
          border-style: solid;
          border-width: 10px;
          }

          .leftarrowdiv:after{ /*arrow added to leftarrowdiv DIV*/
          content:'';
          display:block;
          position:absolute;
          top:10px;
          left:-60px; /*should be set to -border-width x 2 */
          width:0;
          height:0;
          border-
color: transparent rgb(216, 204, 204) transparent transparent; /*border
 color should be same as div div background color*/
          border-style: solid;
          border-width: 30px;
          }

          .rightarrowdiv:after{ /*arrow added to rightarrowdiv DIV*/
          content:'';
          display:block;
          position:absolute;
          top:10px;
          left:100%; /*should be set to 100% */
          width:0;
          height:0;
          border-
color: transparent transparent transparent rgb(216, 204, 204); /*border
 color should be same as div div background color*/
          border-style: solid;
          border-width: 30px;
          }
    </style>
  </body>
</html>
```

#---#

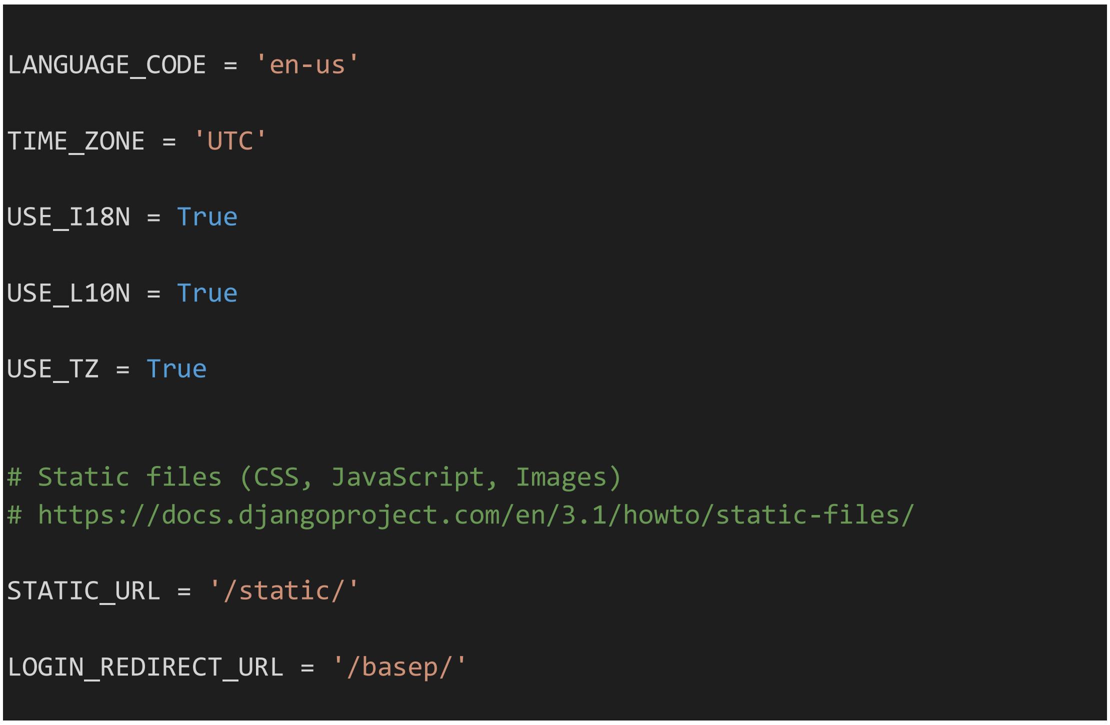
settings.py

#---#

```
"""
Django settings for mysite project.

Generated by 'django-admin startproject' using Django 3.1.2.

For more information on this file, see
https://docs.djangoproject.com/en/3.1/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/3.1/ref/settings/
"""

from pathlib import Path
import os

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent


# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/3.1/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = '1b4+ptv_pl-9l$vn!zuhnw_(#jh3ai4o*ih)a5emm%@e)xlwf9'
```



```
# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []


# Application definition

INSTALLED_APPS = [
    'django_matplotlib',
    'django_extensions',
    'base',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'mysite.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [
            os.path.join(BASE_DIR, 'templates')
        ],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'mysite.wsgi.application'


# Database
# https://docs.djangoproject.com/en/3.1/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}


# Password validation
# https://docs.djangoproject.com/en/3.1/ref/settings/#auth-password-
validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeS
imilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthV
alidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPassword
Validator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswor
dValidator',
    },
]


# Internationalization
# https://docs.djangoproject.com/en/3.1/topics/i18n/
```

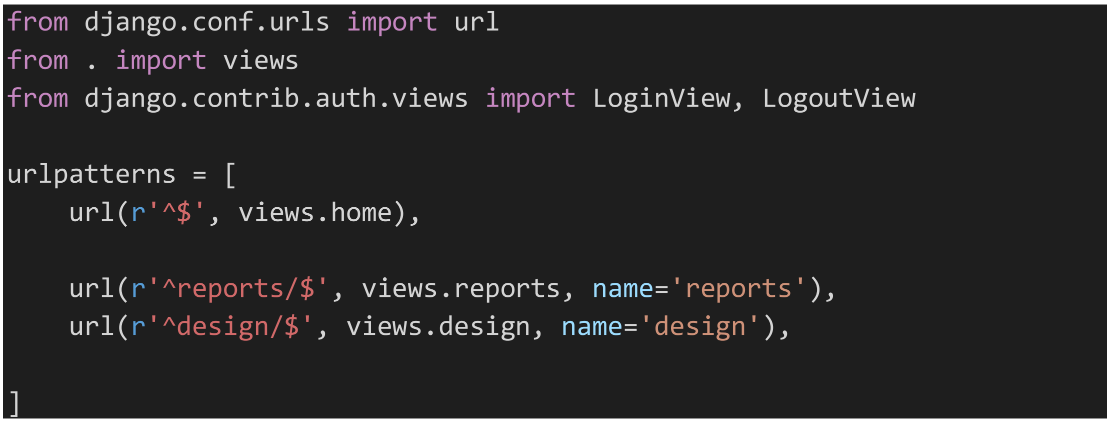


```
LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/3.1/howto/static-files/

STATIC_URL = '/static/'

LOGIN_REDIRECT_URL = '/basep/'
```

#--#

urls.py

#--#



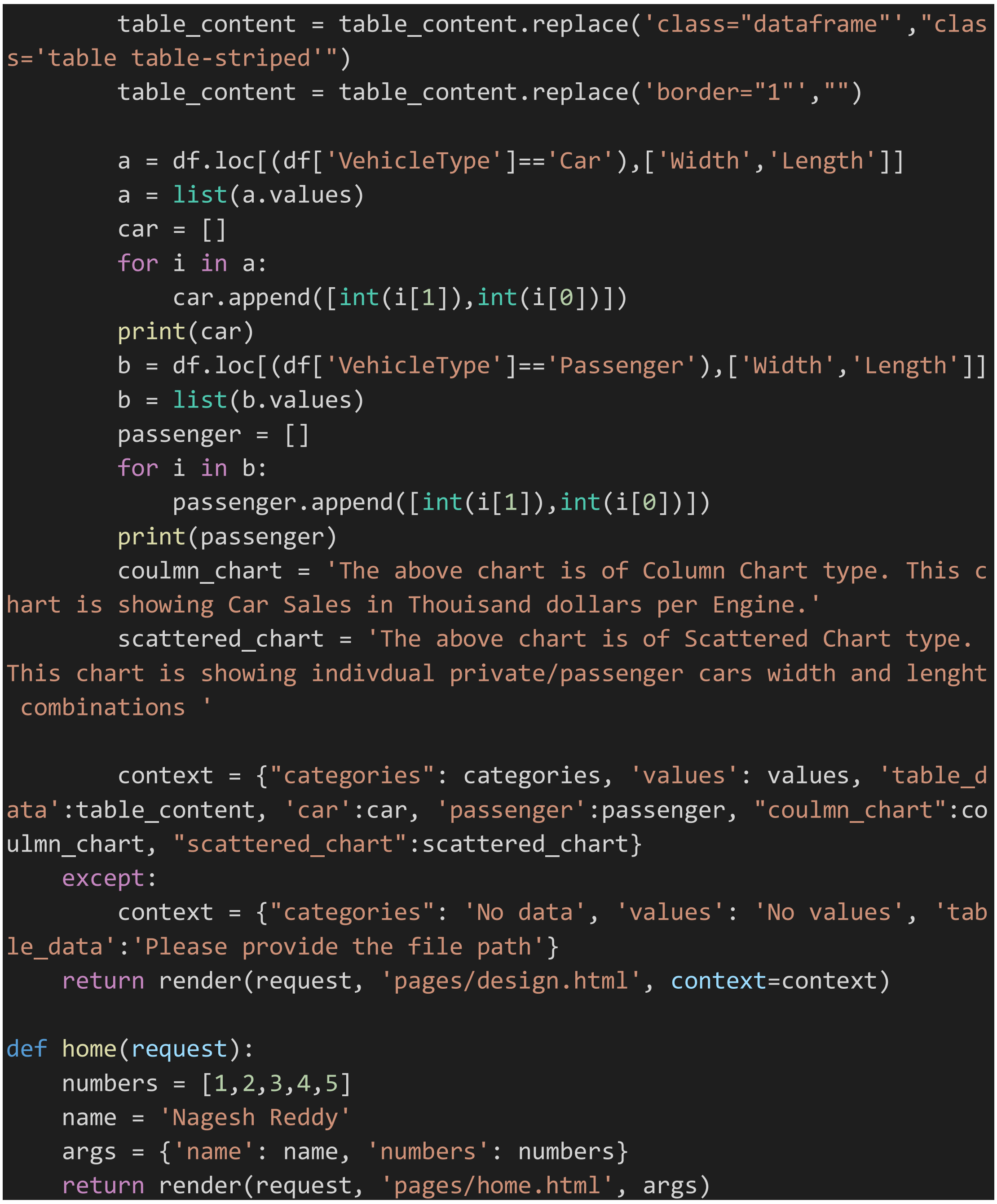
```
from django.conf.urls import url
from . import views
from django.contrib.auth.views import LoginView, LogoutView

urlpatterns = [
    url(r'^$', views.home),

    url(r'^reports/$', views.reports, name='reports'),
    url(r'^design/$', views.design, name='design'),

]
```

#--#

views.py

#--#

```python
from django.shortcuts import render, redirect
from django.contrib.auth.forms import UserCreationForm
from bokeh.plotting import figure, output_file, show
from bokeh.embed import components
from pytrends.request import TrendReq
import pandas as pd
import matplotlib
from matplotlib import pyplot as plt
from django.http import HttpResponse

def reports(request):
    try:
        df = pd.read_csv(r"C:\mysite\input_files\developer_survey_2019\survey_results_public.csv")
        #df = pd.read_csv(request.GET['Upload_File'])#"C:\Users\N2517\Desktop\developer_survey_2019\survey_results_public.csv")#(file_name) #
        rs = df.groupby("Age1stCode")["Age1stCode"].value_counts()
        pd.set_option('display.max_columns', 85)
        pd.set_option('display.max_rows', 85)
        categories = list(rs.index)
        values = list(rs.values)
        table_content = df.to_html(index=None)
        table_content = table_content.replace("","")
        table_content = table_content.replace('class="dataframe"',"class='table table-striped'")
        table_content = table_content.replace('border="1"',"")
        context = {"categories": categories, 'values': values, 'table_data':table_content}
    except:
        context = {"categories": 'No data', 'values': 'No values', 'table_data':'Please provide the file path'}
    return render(request, 'pages/reports.html', context=context)

def design(request):
    #file_name = request.GET #['Upload_File']
    try:
        df = pd.read_csv(request.GET['Upload_File'])    #"C:\mysite\input_files\car_sales.csv"
        rs = df.groupby("Engine size")["Sales in thousands"].agg("sum")
        categories = list(rs.index)
        values = list(rs.values)
        table_content = df.to_html(index=None)
        table_content = table_content.replace("","")
```

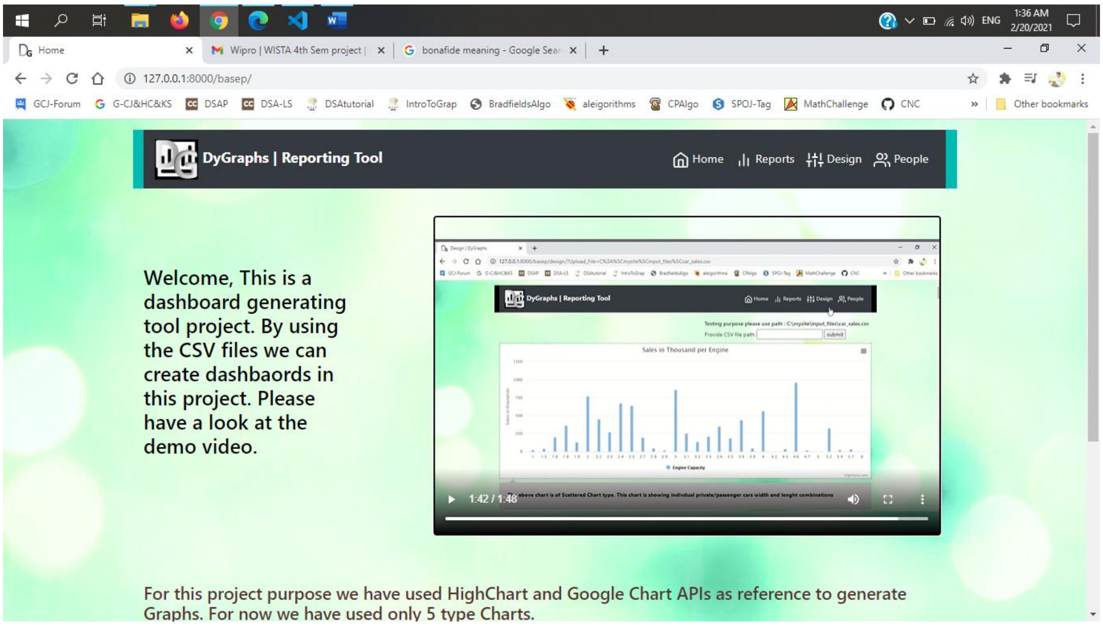


```
        table_content = table_content.replace('class="dataframe"',"clas
s='table table-striped'")
        table_content = table_content.replace('border="1"',"")

        a = df.loc[(df['VehicleType']=='Car'),['Width','Length']]
        a = list(a.values)
        car = []
        for i in a:
            car.append([int(i[1]),int(i[0])])
        print(car)
        b = df.loc[(df['VehicleType']=='Passenger'),['Width','Length']]
        b = list(b.values)
        passenger = []
        for i in b:
            passenger.append([int(i[1]),int(i[0])])
        print(passenger)
        coulmn_chart = 'The above chart is of Column Chart type. This c
hart is showing Car Sales in Thouisand dollars per Engine.'
        scattered_chart = 'The above chart is of Scattered Chart type.
This chart is showing indivdual private/passenger cars width and lenght
 combinations '

        context = {"categories": categories, 'values': values, 'table_d
ata':table_content, 'car':car, 'passenger':passenger, "coulmn_chart":co
ulmn_chart, "scattered_chart":scattered_chart}
    except:
        context = {"categories": 'No data', 'values': 'No values', 'tab
le_data':'Please provide the file path'}
    return render(request, 'pages/design.html', context=context)

def home(request):
    numbers = [1,2,3,4,5]
    name = 'Nagesh Reddy'
    args = {'name': name, 'numbers': numbers}
    return render(request, 'pages/home.html', args)
```

CHAPTER 6

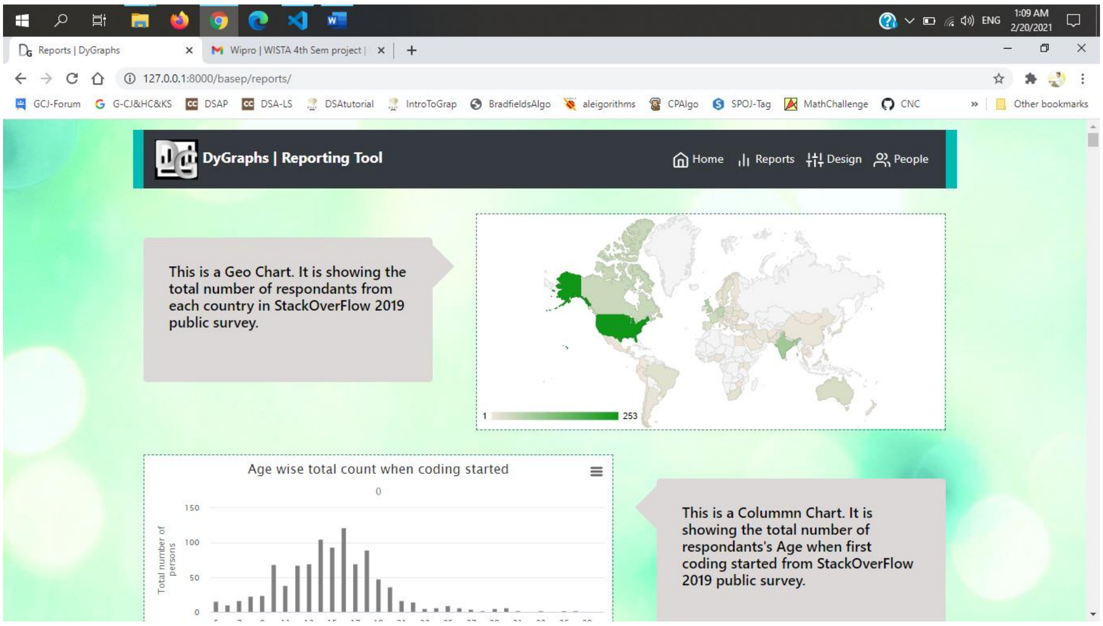
RESULTS

Screenshots of Application



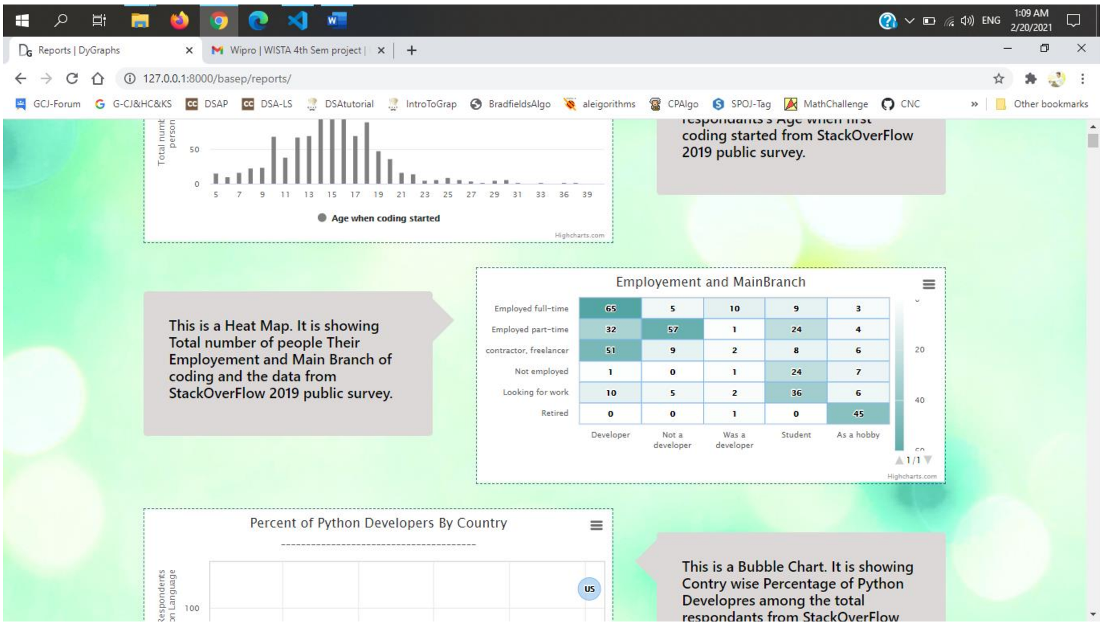
Welcome, This is a dashboard generating tool project. By using the CSV files we can create dashbaords in this project. Please have a look at the demo video.

For this project purpose we have used HighChart and Google Chart APIs as reference to generate Graphs. For now we have used only 5 type Charts.



This is a Geo Chart. It is showing the total number of respondants from each country in StackOverFlow 2019 public survey.

This is a Colummn Chart. It is showing the total number of respondants's Age when first coding started from StackOverFlow 2019 public survey.



respondants's Age when first coding started from StackOverFlow 2019 public survey.

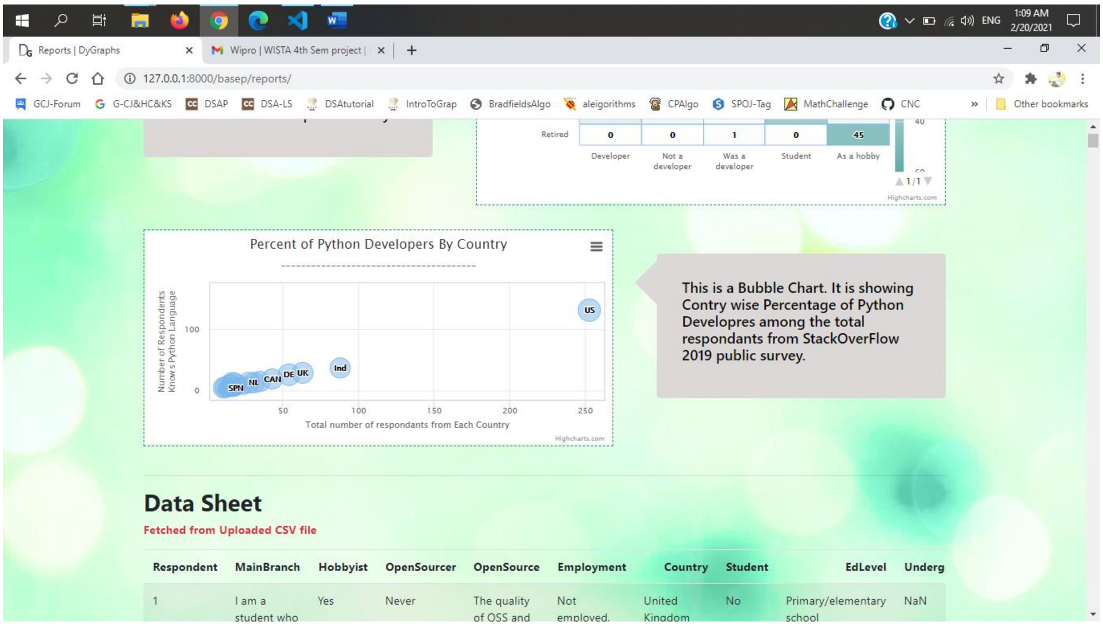
This is a Heat Map. It is showing Total number of people Their Employement and Main Branch of coding and the data from StackOverFlow 2019 public survey.

Employement and MainBranch

	Developer	Not a developer	Was a developer	Student	As a hobby
Employed full-time	65	5	10	9	3
Employed part-time	32	57	1	24	4
contractor, freelancer	51	9	2	8	6
Not employed	1	0	1	24	7
Looking for work	10	5	2	36	6
Retired	0	0	1	0	45

Percent of Python Developers By Country

This is a Bubble Chart. It is showing Contry wise Percentage of Python Developres among the total respondants from StackOverFlow


Reports | DyGraphs
Wipro | WISTA 4th Sem project |
127.0.0.1:8000/basep/reports/
ENG
1:09 AM
2/20/2021
40
1/1
Highcharts.com
Number of Respondents Knows Python Language
100
0
US
Ind
DE
UK
CAN
NL
SPN
50
100
150
200
250
Total number of respondants from Each Country
Highcharts.com

Retired	0	0	1	0	45
	Developer	Not a developer	Was a developer	Student	As a hobby

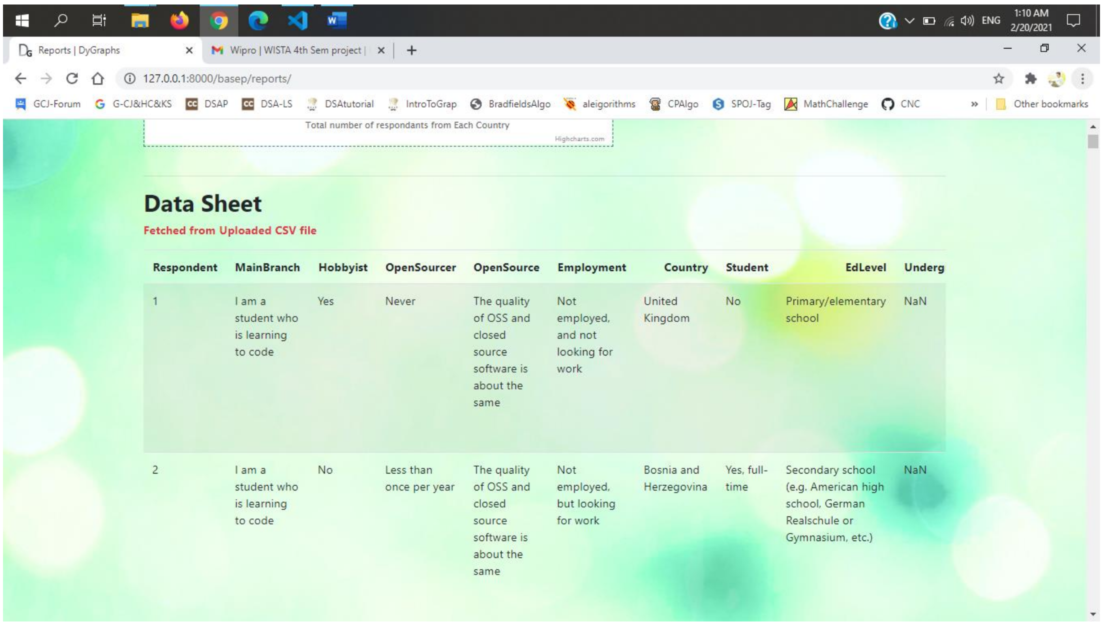
Percent of Python Developers By Country

This is a Bubble Chart. It is showing Contry wise Percentage of Python Developres among the total respondants from StackOverFlow 2019 public survey.

Data Sheet

Fetched from Uploaded CSV file

Respondent	MainBranch	Hobbyist	OpenSourcer	OpenSource	Employment	Country	Student	EdLevel	Underg
1	I am a student who	Yes	Never	The quality of OSS and	Not employed,	United Kingdom	No	Primary/elementary school	NaN



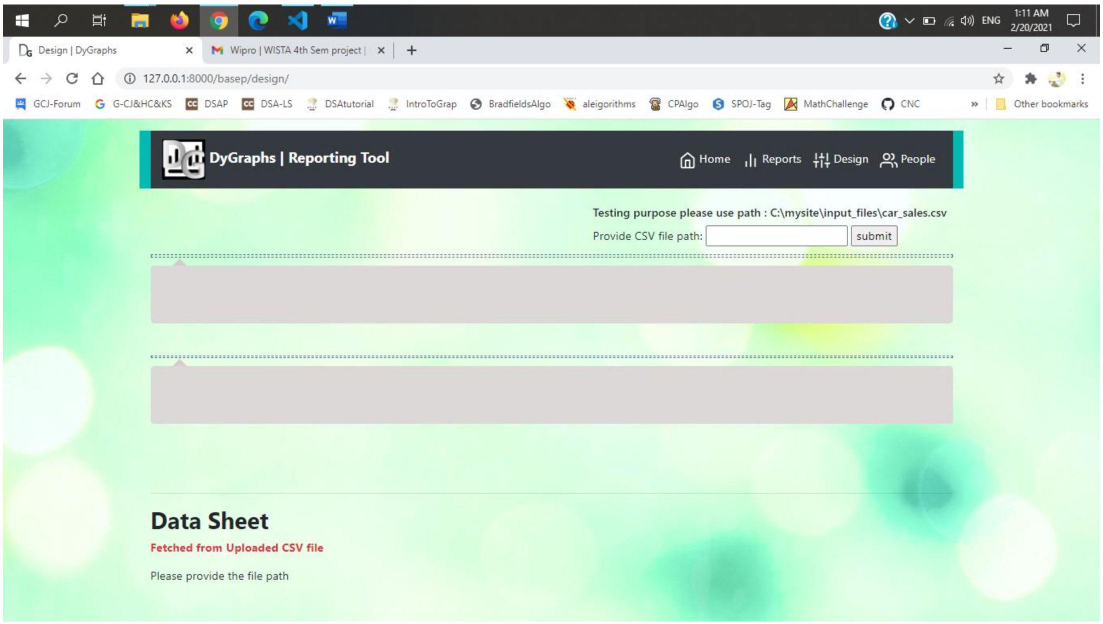
Total number of respondants from Each Country

Highcharts.com

Data Sheet

Fetched from Uploaded CSV file

Respondent	MainBranch	Hobbyist	OpenSourcer	OpenSource	Employment	Country	Student	EdLevel	Underg
1	I am a student who is learning to code	Yes	Never	The quality of OSS and closed source software is about the same	Not employed, and not looking for work	United Kingdom	No	Primary/elementary school	NaN
2	I am a student who is learning to code	No	Less than once per year	The quality of OSS and closed source software is about the same	Not employed, but looking for work	Bosnia and Herzegovina	Yes, full-time	Secondary school (e.g. American high school, German Realschule or Gymnasium, etc.)	NaN



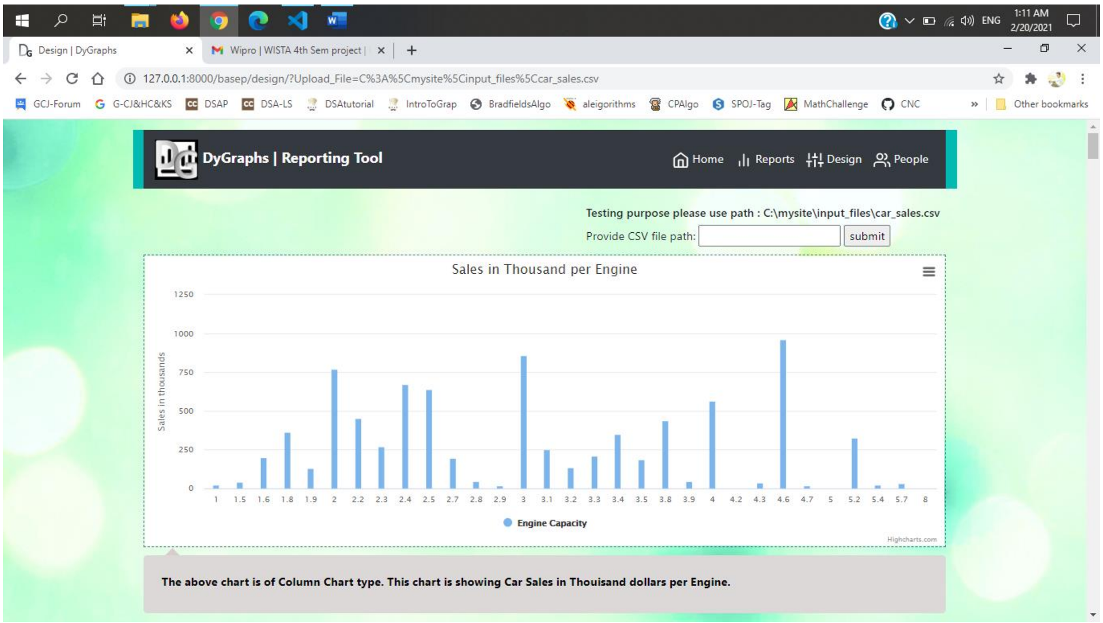
Testing purpose please use path : C:\mysite\input_files\car_sales.csv

Provide CSV file path:

Data Sheet

Fetched from Uploaded CSV file

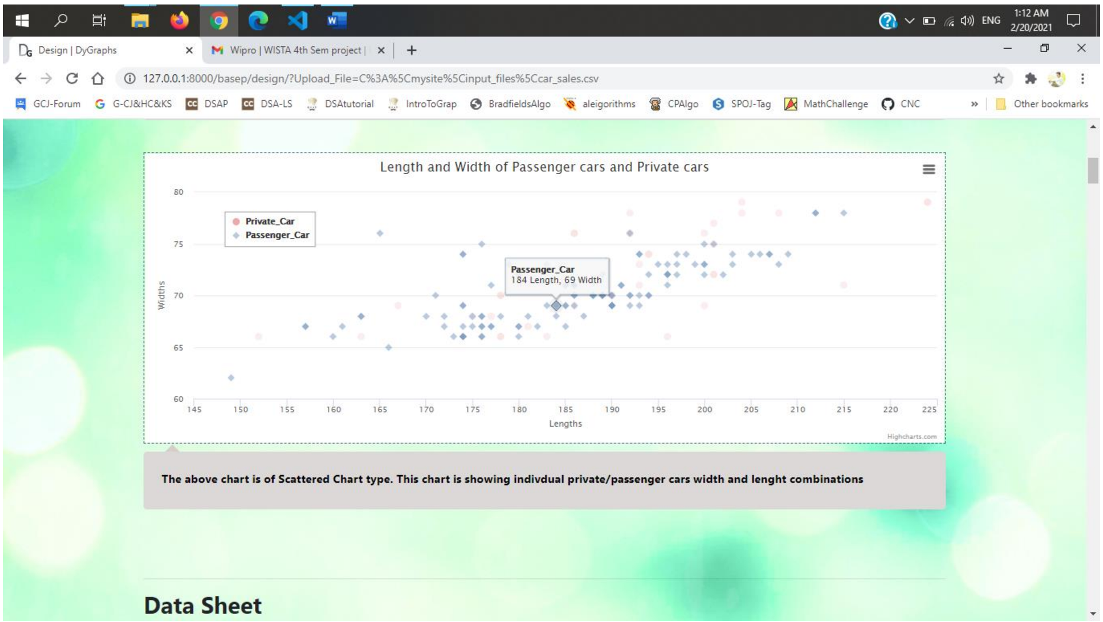
Please provide the file path


Design | DyGraphs
Wipro | WISTA 4th Sem project |
127.0.0.1:8000/basep/design/?Upload_File=C%3A%5Cmysite%5Cinput_files%5Ccar_sales.csv
GCJ-Forum G-CJ&HC&KS DSAP DSA-LS DSAtutorial IntroToGrap BradfieldsAlgo aleigorithms CPAlgo SPOJ-Tag MathChallenge CNC Other bookmarks
DyGraphs | Reporting Tool
Home Reports Design People
Sales in Thousand per Engine
Sales in thousands
1250
1000
750
500
250
0
1 1.5 1.6 1.8 1.9 2 2.2 2.3 2.4 2.5 2.7 2.8 2.9 3 3.1 3.2 3.3 3.4 3.5 3.8 3.9 4 4.2 4.3 4.6 4.7 5 5.2 5.4 5.7 8
● Engine Capacity
Highcharts.com
1:11 AM
2/20/2021

Testing purpose please use path : C:\mysite\input_files\car_sales.csv

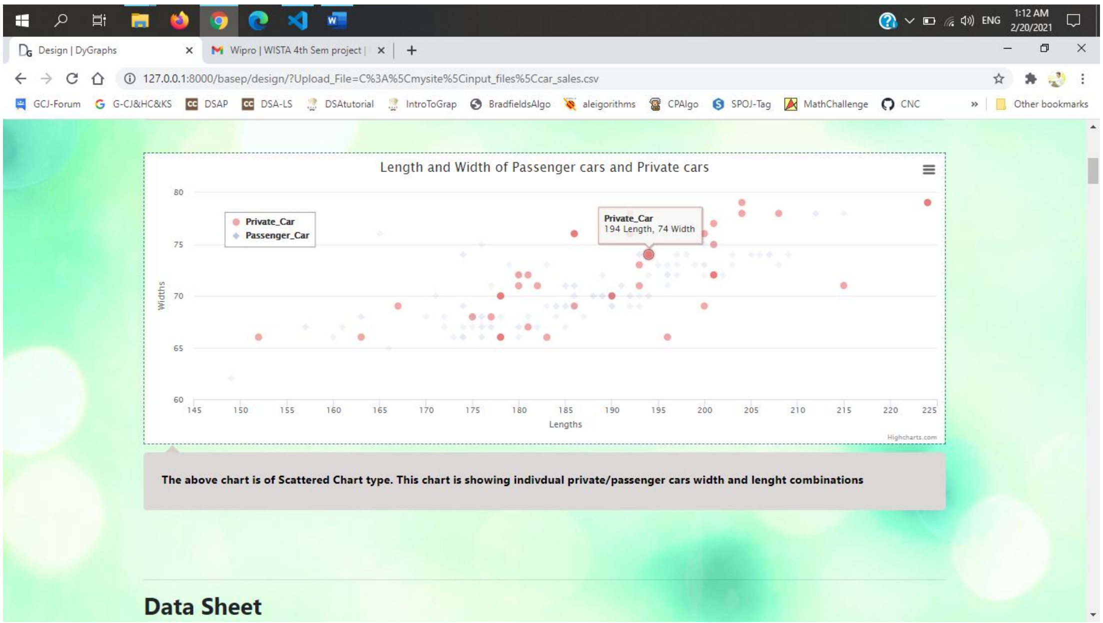
Provide CSV file path: submit

The above chart is of Column Chart type. This chart is showing Car Sales in Thouisand dollars per Engine.


Design | DyGraphs
Wipro | WISTA 4th Sem project |
127.0.0.1:8000/basep/design/?Upload_File=C%3A%5Cmysite%5Cinput_files%5Ccar_sales.csv
GCJ-Forum G-CJ&HC&KS DSAP DSA-LS DSAtutorial IntroToGrap BradfieldsAlgo aleigorithms CPAlgo SPOJ-Tag MathChallenge CNC Other bookmarks
Length and Width of Passenger cars and Private cars
● Private_Car
◆ Passenger_Car
Passenger_Car
184 Length, 69 Width
Widths
80
75
70
65
60
145 150 155 160 165 170 175 180 185 190 195 200 205 210 215 220 225
Lengths
Highcharts.com
1:12 AM
2/20/2021

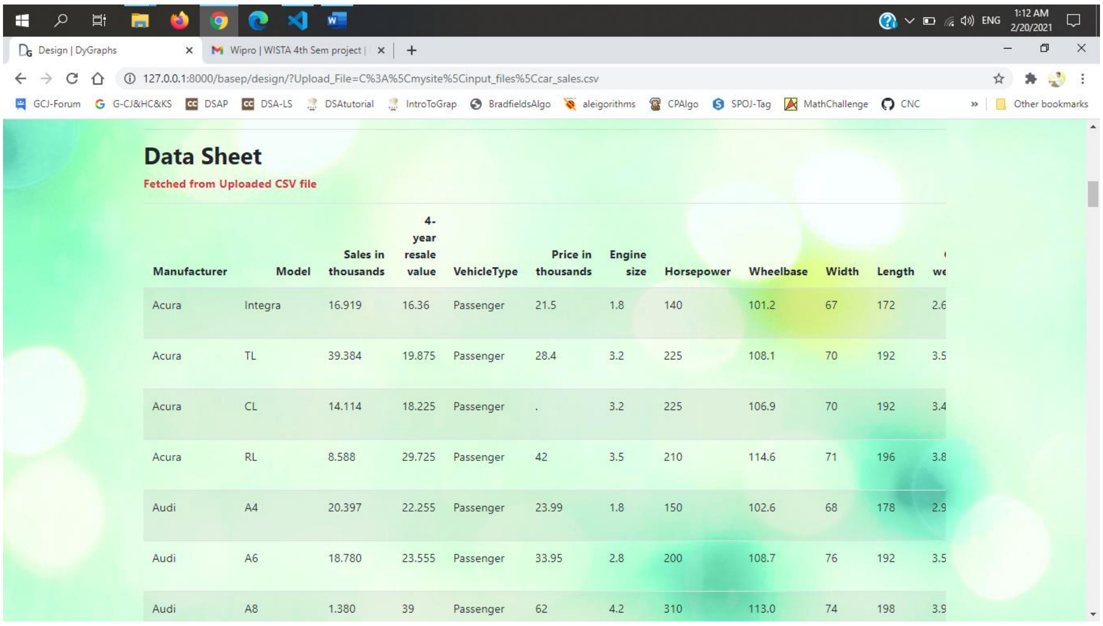
The above chart is of Scattered Chart type. This chart is showing indivdual private/passenger cars width and lenght combinations

Data Sheet


Design | DyGraphs
Wipro | WISTA 4th Sem project |
127.0.0.1:8000/basep/design/?Upload_File=C%3A%5Cmysite%5Cinput_files%5Ccar_sales.csv
Length and Width of Passenger cars and Private cars
Private_Car
Passenger_Car
Private_Car
194 Length, 74 Width
Widths
80
75
70
65
60
145
150
155
160
165
170
175
180
185
190
195
200
205
210
215
220
225
Lengths
Highcharts.com

The above chart is of Scattered Chart type. This chart is showing indivdual private/passenger cars width and lenght combinations

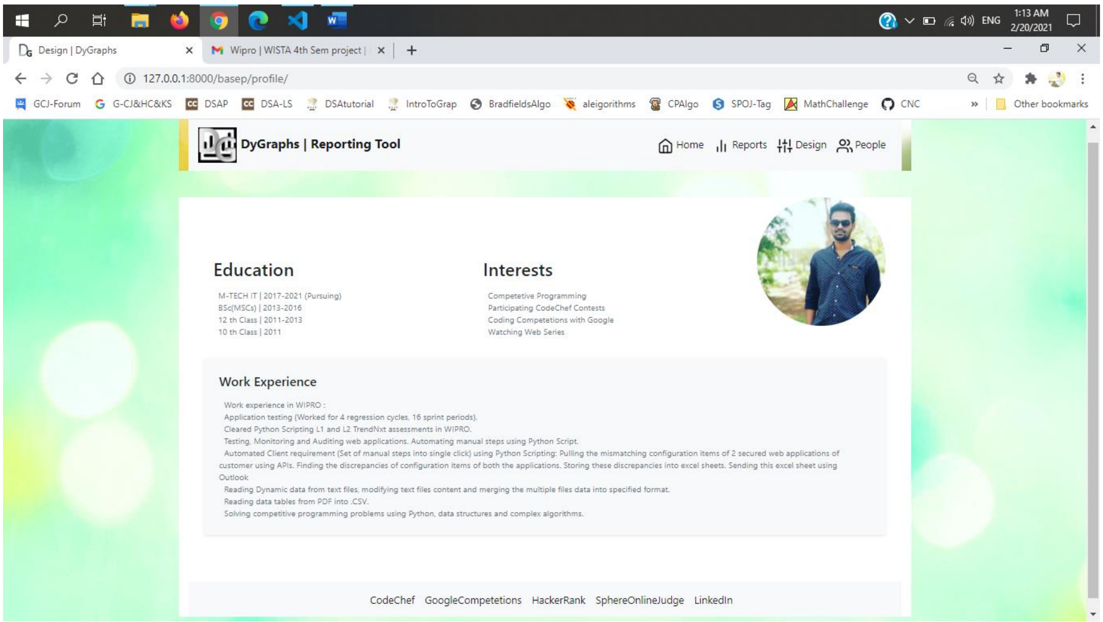
Data Sheet


Design | DyGraphs
Wipro | WISTA 4th Sem project |
127.0.0.1:8000/basep/design/?Upload_File=C%3A%5Cmysite%5Cinput_files%5Ccar_sales.csv

Data Sheet

Fetched from Uploaded CSV file

Manufacturer	Model	Sales in thousands	4-year resale value	VehicleType	Price in thousands	Engine size	Horsepower	Wheelbase	Width	Length	we
Acura	Integra	16.919	16.36	Passenger	21.5	1.8	140	101.2	67	172	2.6
Acura	TL	39.384	19.875	Passenger	28.4	3.2	225	108.1	70	192	3.5
Acura	CL	14.114	18.225	Passenger	.	3.2	225	106.9	70	192	3.4
Acura	RL	8.588	29.725	Passenger	42	3.5	210	114.6	71	196	3.8
Audi	A4	20.397	22.255	Passenger	23.99	1.8	150	102.6	68	178	2.9
Audi	A6	18.780	23.555	Passenger	33.95	2.8	200	108.7	76	192	3.5
Audi	A8	1.380	39	Passenger	62	4.2	310	113.0	74	198	3.9



DyGraphs | Reporting Tool

Education

M-TECH IT | 2017-2021 (Pursuing)
BSc(MSCs) | 2013-2016
12 th Class | 2011-2013
10 th Class | 2011

Interests

Competetive Programming
Participating CodeChef Contests
Coding Competetions with Google
Watching Web Series

Work Experience

Work experience in WIPRO :
Application testing (Worked for 4 regression cycles, 16 sprint periods).
Cleared Python Scripting L1 and L2 TrendNxt assessments in WIPRO.
Testing, Monitoring and Auditing web applications. Automating manual steps using Python Script.
Automated Client requirement (Set of manual steps into single click) using Python Scripting: Pulling the mismatching configuration items of 2 secured web applications of customer using APIs. Finding the discrepancies of configuration items of both the applications. Storing these discrepancies into excel sheets. Sending this excel sheet using Outlook
Reading Dynamic data from text files, modifying text files content and merging the multiple files data into specified format.
Reading data tables from PDF into .CSV.
Solving competitive programming problems using Python, data structures and complex algorithms.

CodeChef GoogleCompetetions HackerRank SphereOnlineJudge LinkedIn

CHAPTER 7

CONCLUSION

Proposed system can create dashboard for specified csv files for now. In our application total we are showing 6 different types of graphs for 2 different data sets.

Following are the Graphs Which we are shown in the application:

i. Geo Chart
ii. Heat Map
iii. Scattered Chart
iv. Column Chart
v. Bubble Chart

We have taken two different kinds of data sets:

i. Care Sales data from Kaggle
ii. StackOverflow 2019 public survey data.

CHAPTER 8

REFERENCES

- Django framework documentation from Internet: https://docs.djangoproject.com/en/3.1/

- IEEE Recommended Practice for Software Requirements Specification-IEEE Std 830-1993.

- Python Programming language documentation from Internet: https://www.python.org/doc/

- Pandas Python library documentation from Internet: https://pandas.pydata.org/docs/

- Numpy Python library documentation from Internet: https://numpy.org/doc/

- Stack Overflow Survey from Stack OverFlow site from Internet: https://insights.stackoverflow.com/survey

- Datapane Python library documentation from Internet: https://datapane.com/

- Dash Python library documentation from Internet: https://dash.plotly.com/

- Plotly Python library documentation from Internet: https://plotly.com/python/

- Matplotlib Python library documentation from Internet: https://matplotlib.org/

- Bokeh Python library documentation from Internet: https://docs.bokeh.org/en/latest/index.html

- Car Sales data from Kaggle from Internet: https://www.kaggle.com/hsinha53/car-sales

- Bootstarp documentation from Internet: https://getbootstrap.com/

- HighCharts for creating charts: https://www.highcharts.com/?credits

- Google Chart for creating chart: https://developers.google.com/chart/interactive/docs/quick_start