BRIGHTCOM GROUP

BCG EMPLOYEE DETAILS FORM APPLICATION

Contents:

Description

Abstract

Software's Required

Html(Hyper Text Mockup Language)

Css(Casscading Style Sheet)

BootStrap

MySql

Python/

Django

JavaScript

Abstract:

The Employee Details Form is a document that is used to gather information about a new employee, including their personal and contact information, as well as their employment history, qualifications, and job-related preferences. The purpose of the form is to create a comprehensive profile of the employee that can be used to manage their employment and help them integrate into their new role. The form typically includes fields for name, address, phone number, email, education, work experience, job title, start date, and emergency contact information, among others. This form is important for HR and payroll purposes and is usually filled out during the onboarding process.

Teammembers:

- Naveen
- Vignan
- Ravi teja
- Sai srikar
- Anitha
- Lakshmi koteswari
- Madhava sharma
- Mamatha

Task:

- We have to create the forms page in Html and we have to give the employee details (Empid,Emp name,Date of Join,Experted Salary,previous experience&Designation)
- We have to create the project in Django
- We have to create the Database by MySql
- We have to link the Html page & Django application
- We have to give login access to HR and Manager
- Employee details present in Database we have to do sorting in accending order and desending order
- We have to do pagination for Employee Details present in database

Software's Required:

Softwares Required	Versions
Html(Hyper Text Mockup Language)	Html 5
Css(Casscading Style Sheet)	Css3
BootStrap	Bootstrap 4
MySql	8.0.32
Python	Python 3.11
Django	4.1, 4.1.6
JavaScript	Es6
Operating System	Windows 10/11

Html&Css:

HTML5 is the fifth revision of the HTML standard and brings a range of new features and capabilities to web development. Some of the most notable features include improved support for multimedia content, such as video and audio, new semantic elements to describe different parts of a web page, and improved support for offline web applications.
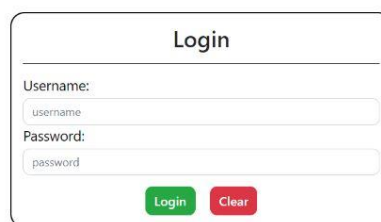
CSS3, on the other hand, is the latest version of the style sheet language used to describe the look and formatting of a web page. It provides a range of new selectors, properties, and techniques for creating modern, dynamic, and responsive web pages. Some of the most notable features in

CSS3 include support for responsive design, new layout and positioning techniques, and advanced effects and animations.

Together, HTML5 and CSS3 provide a powerful platform for creating modern, dynamic, and engaging websites and web applications.

By using Html5& Css3 we have developed following web pages:

- Empform.html
- Homepage.html
- Login.html
- empdetails.html

(Login Page)

(Home Page)

(Employee_Details)

Bootstrap:

Bootstrap is a popular open-source front-end framework used for developing responsive and mobile-first websites and web applications. It was developed by Twitter and was originally released in 2011.

Bootstrap is based on HTML, CSS, and JavaScript and provides a set of pre-designed components, such as navigation bars, buttons, forms, and modals, that can be easily added to a project to create a user-friendly interface. It also provides a grid-based layout system that makes it easy to create responsive, multi-column designs.

Bootstrap also includes a number of CSS classes that can be used to style elements on a page and to implement various design elements, such as typography, colors, and spacing. The framework also supports JavaScript plugins for adding interactivity to a page, such as dropdown menus and carousels.

Bootstrap is widely used by developers and designers for its simplicity and versatility. It allows for quick and easy prototyping, and can be customized to fit the specific needs of a project.

Note: in this project we have done pagination by using bootstrap the code as follows:

```html
<nav aria-label="...">
    {% if empdat.has_other_pages %}
      <ul class="pagination">
        {% if empdat.has_previous %}
          <li class="page-item"><a
class="page-link" href="?page={{
empdata.previous_page_number
}}">Previous</a></li>
        {% else %}
```

```html
            <li class="page-item
disabled"><span class="page-
link">Previous</span></li>
            {% endif %}

            {% for i in
empdat.paginator.page_range %}
            {% if empdat.number == i %}
            <li class="page-item active"
aria-current="page"><a class="page-link"
href="#">{{ i }}</a></li>
            {% else %}
            <li class="page-item"><a
class="page-link" href="?page={{ i }}">{{ i
}}</a></li>
            {% endif %}
            {% endfor %}

            {% if empdat.has_next %}
            <li class="page-item"><a
class="page-link" href="?page={{
empdata.next_page_number }}">Next</a></li>
            {% else %}
            <li class="page-item"><span
class="page-link">Next</span></li>
            {% endif %}
        </ul>
      {% endif %}
      </nav>
```

(pagination)

Python & Django:

DJANGO:

Django is a framework which is used to create/design web-application.django is 3rd party module

1.Django Features:

- We can create any application with in less time.
- Django is versatile framework because we can create any kind of domine (sales,health........) websites.
- Django is more securable because lot of pre-defined code with more security.
- Django supports all databases like mysql,sql,.........
- Django is portable framework.

2.Django Application:

- Django project is a collection of applications along with their configurations.
- A project must required atleast one aplication and almost n number of application.
- Django-admin.py file will be installed automatically where we install django.
- We use django-admin.py file to create django project.

syntax:

- django-admin startproject <project name>
- An application is a specific task of the project.

manage.py:

- manage.py file will be create automatically whenever we create project.
- We use manage.py file to create django application.
- Also use startapp command along with manage.py file to create django application.

syntax:

- python manage.py startapp <appname>

3 .Django Archticuture:

- Whenever user hits the request from browser.
- The request goes to urls.py file and matches the corresponding pattern.

- If the pattern is matching then it will go to the corresponding view and execute the view
- From views,it will go to the models.py file and execute the model in models.py file.
- It may also go to templates and execute the presentation code in the templates.
- Finally, after executing all files the httpRespones will return to browser.

4.Generall points:

- Django follows MVT model
- M----->model-------------->to write databases related code
- V------->views------------> to write business related code
- T------>templates-------->to write presentation related code

Start project :

step1: settings.py:

1.installapp:

'appname';

2.databases:

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.mysql',

```
    'NAME': 'mydb',

    'USER': 'srikar',

    'PASSWORD': 'Srikar@2001',

    'HOST':'192.168.30.72',

    'PORT':'3306',

  }

}
```

3.import os

4.TEMPLATES_DIR=os.path.join(BASE_DIR,'templates')

5.temlates:

dir=[TEMPLATES_DIR]

step2: MODELS.py:

➢ A models is python class which is used to create database table.
➢ We can use class keyword to create django class.
➢ Models.Model is base class for every user defined class.
➢ Model is a collection of python class keyword ,classname,field name and field type.
➢ Use makemigrations and migrate commaands to convert django model into database table.

Makemigrations:

syntax:

> ➢ python manage.py makemigrations.

1. We run the makemigrations command in terminal then django will go to models.py file and check for latest modifications.
2. If any migration are in orm language then it will be converted into sql language.

Migrate:

syntax:

> ➢ python manage.py migrate

1. It will create a new python file in migrations folder and save the sql code.
2. If any pythonfile available in models.py then it will take sql code from that file and execute the database,so it will create table as per django model.

step 3: Views.py:

1. A views is python function which takes httprequest and executes the views body and returns httpResponse.
2. We use a def keyword to create views
3. Every view ends with any one of the following function:
   - render:
     > ➢ return httpRequest from the templates.

- HttpResponse:
    - ➤ returns the httpRequest to browser as httpResponse.
4. We use conditional statements to receive the type of request like GET or POST.
- GET:------------->GET requests are only used to request data.
- POST:------------>POST is stored in the request body of the HTTP request.
- SORT :-----------> sort is used to descending and ascending by specific column.

step4: Templates:

- templates means .html file
- {{}}------>To write variable name.
- { % %}-------->To add link one page to another page.

step5:urls.py:

It contains all the names and path of views which help to redirect to views.

step6: runserver

syntax:

- ➤ python manage.py runserver.

```
PROBLEMS  1     OUTPUT    DEBUG CONSOLE    TERMINAL

PS C:\Users\USER\Desktop\finalapp> python manage.py createsuperuser
Email address:
Password:
Password (again):
This password is too common.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
PS C:\Users\USER\Desktop\finalapp> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
February 13, 2023 - 09:27:10
Django version 4.1.6, using settings 'bg.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

JavaScript:

JavaScript is a high-level, interpreted programming language that is widely used for front-end web development. It is used to add interactivity, dynamic behaviour, and other advanced features to websites. JavaScript runs in the user's web browser, allowing for a dynamic and interactive user experience.

JavaScript can be used for a variety of tasks, including:

Form validation: JavaScript can be used to check if a user has entered valid information into a form before it is submitted.

Dynamic content updates: JavaScript can be used to update content on a web page without having to reload the entire page.

Animations and effects: JavaScript can be used to create animations, such as sliding images or fading text, and other visual effects.

User interactivity: JavaScript can be used to respond to user actions, such as clicking on buttons or moving the mouse, and take appropriate actions.

JavaScript is a flexible and versatile language that can be used in a variety of web development projects. It can be combined with HTML and CSS to create dynamic and interactive web pages, and can also be used with back-end technologies, such as Node.js, to create full-stack web applications.

Note : Java script is used for sorting in our application the following code as follows:

```
<script>
    function sortTable(n) {
      var table, rows, switching, i, x, y,
shouldSwitch, dir, switchcount = 0;
      table =
document.getElementById("myTable");
      switching = true;
      //Set the sorting direction to
ascending:
      dir = "asc";
      /*Make a loop that will continue until
      no switching has been done:*/
      while (switching) {
        //start by saying: no switching is
done:
          switching = false;
          rows = table.rows;
```

```
        /*Loop through all table rows (except
the
        first, which contains table
headers):*/
        for (i = 1; i < (rows.length - 1);
i++) {
            //start by saying there should be
no switching:
            shouldSwitch = false;
            /*Get the two elements you want to
compare,
            one from current row and one from
the next:*/
            x =
rows[i].getElementsByTagName("TD")[n];
            y = rows[i +
1].getElementsByTagName("TD")[n];
            /*check if the two rows should
switch place,
            based on the direction, asc or
desc:*/
            if (dir == "asc") {
                if (x.innerHTML.toLowerCase() >
y.innerHTML.toLowerCase()) {
                    //if so, mark as a switch and
break the loop:
                    shouldSwitch= true;
                    break;
                }
            } else if (dir == "desc") {
                if (x.innerHTML.toLowerCase() <
y.innerHTML.toLowerCase()) {
```

```
                //if so, mark as a switch and
break the loop:
                shouldSwitch = true;
                break;
            }
          }
        }
        if (shouldSwitch) {
          /*If a switch has been marked, make
the switch
          and mark that a switch has been
done:*/
          rows[i].parentNode.insertBefore(row
s[i + 1], rows[i]);
          switching = true;
          //Each time a switch is done,
increase this count by 1:
          switchcount ++;
        } else {
          /*If no switching has been done AND
the direction is "asc",
          set the direction to "desc" and run
the while loop again.*/
          if (switchcount == 0 && dir ==
"asc") {
            dir = "desc";
            switching = true;
          }
        }
      }
    }
  </script>
```

MySQL:

MySQL is a widely-used open-source relational database management system (RDBMS). It is used for storing, organizing, and retrieving data in a structured way. MySQL is a popular choice for web-based applications, as it is fast, reliable, and easy to use.

MySQL is based on the Structured Query Language (SQL), which is used to create, manipulate, and query the data stored in a database. In MySQL, data is stored in tables, which consist of rows and columns. Each row represents a single record, and each column represents a specific field of data.

MySQL provides a number of features that make it a powerful tool for managing data, including:

- Data security: MySQL provides a number of security features to ensure the protection of data stored in a database, including user authentication and access control.
- Scalability: MySQL can handle large amounts of data and can easily be scaled up to support growing amounts of data.
- High performance: MySQL is optimized for fast performance and can handle large amounts of data and concurrent users.

- Cross-platform support: MySQL runs on a variety of platforms, including Windows, macOS, and Linux, and can be used with a wide range of programming languages, including PHP, Java, and Python.

Overall, MySQL is a widely-used and reliable database management system that is well suited for web-based applications and other data-intensive projects.

Note: we have to create Database by following Quary

To connect database remotely we have to run following ***commands in our mysql database:***

- create user 'root'@'%' identified by 'admin';
- grant all privileges on *.* to 'root'@'%' with grant option;
- flush privileges;
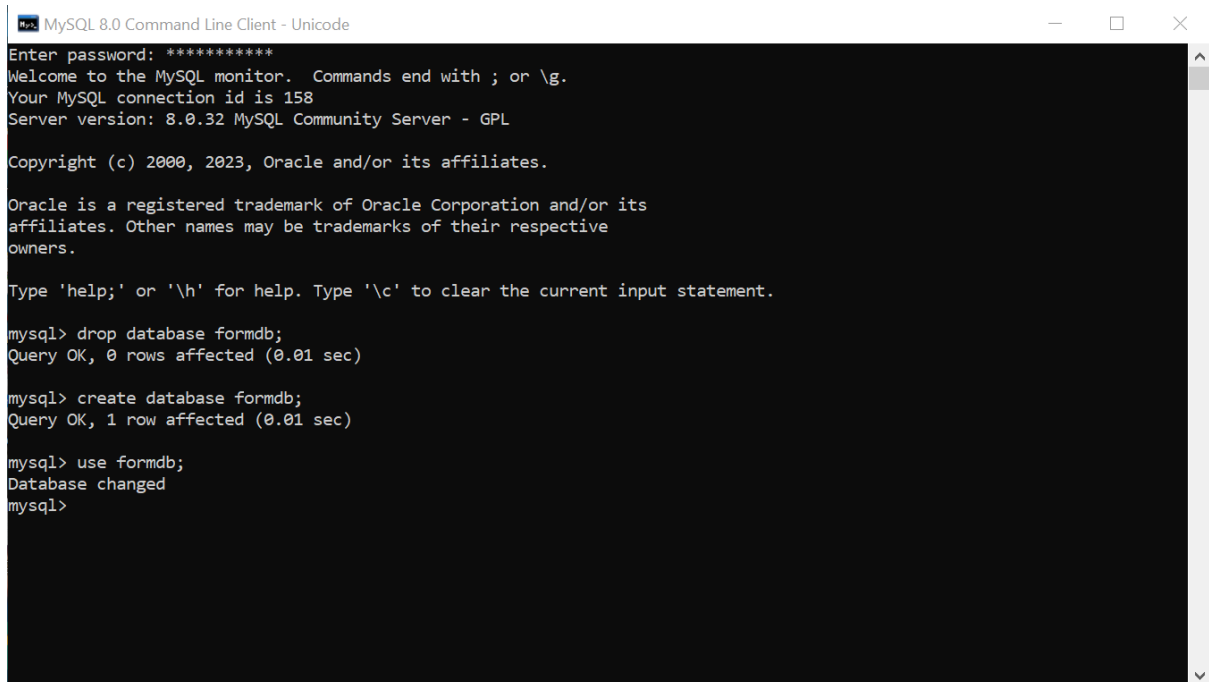
we have to configure database in settings.py

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': "formdb",
        'USER': "Madhav",
        'PASSWORD':"Madhav@1996",
        'HOST':"localhost",
        'PORT':3306
    }
}
```

We have to create table in modules.py by using following quary in mysql: ***create database formdb;***