

FEROZE GANDHI POLYTECHNIC

RAEBARELI

MAJOR PROJECT

“FLASK BLOG”

SESSION 2021-22

SUBMITTED TO

DEEP SONI

FEROZE GANDHI POLYTECHNIC

SUBMITTED BY

AKSHAT SRIVASTAVA

AQSA FATIMA

BHASKAR GUPTA

PRAGATI SINGH

Information Technology

3rd Year 6th Sem

ACKNOWLEDGEMENT

Apart from my effort, the success of the project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who express have been instrumental in the successful completion of this project.

We would like to express our deep and sincere gratitude to our supervisor Mr. Deep Soni Sir (Project Teacher) who gave me his full support and encouraged us to work in an innovative and challenging project in the educational field. His wide knowledge and logical thinking gave us right direction all the time.

We are deeply grateful to our project coordinator for his help and support provided at every step of the project.

Last but not the least, we thank to all our classmates for their support and co-operation.

**AKSHAT SRIVASTAVA
AQSA FATIMA
BHASKAR GUPTA
PRAGATI SINGH**

DECLARATION

This is to certify that the project report entitled “Flask Blog” is done by us is an authentic work carried out for the partial fulfilment of the requirements for the award of the Diploma in “(Information Technology)” under the guidance of Mr. Deep Soni. The matter embodied in this project work has not been submitted earlier for awarding any degree or diploma to the best of our knowledge and belief.

**AKSHAT SRIVASTAVA
AQSA FATIMA
BHASKAR GUPTA
PRAGATI SINGH**

Project Objective

Build a fullfledged flask-based website and connect it to a backend database-based MySQL to hold on blogs, and contact us requests.

Modules Used

1. Python's Flask

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. Wikipedia

Developer(s): Armin Ronacher

License: BSD

Initial release: April 1, 2010; 12 years ago

Stable release: 2.1.2 / 28 April 2022; 45 days ago

Written in: Python

2. Python's inbuilt modules

Built-in modules are **written in C and integrated with the Python shell**. Each built-in module contains resources for certain system-specific functionalities such as OS management, disk IO, etc. The standard library also contains many Python scripts (with the .py extension) containing useful utilities.

3. Flask SQLALchemy

Using raw SQL in Flask web applications to perform CRUD operations on databases can be tedious. Instead, SQLAlchemy, a Python toolkit is a powerful OR Mapper that gives application developers the full power and flexibility of SQL. Flask-SQLAlchemy is the Flask extension that adds support for SQLAlchemy to your Flask application.

4. JSON

JSON (JavaScript Object Notation), specified by **RFC 7159** (which obsoletes **RFC 4627**) and by ECMA-404, is a lightweight data-interchange format inspired by JavaScript object literal syntax (although it is not a strict subset of JavaScript 1).

json exposes an API familiar to users of the standard library marshal and pickle modules

5. Flask Mail

Web-based applications typically require the ability to send mail to the user/client. Flask doesn't have an out-of-the-box solution to send mail.

Instead, the Flask-Mail extension makes it easy to establish a simple interface with any email server.

Email generally uses two protocols, one for sending mails (smtp) and one for receiving mail (pop3). This article is about sending mails.

6. Secure_filename from werkzeug.utils

`werkzeug.secure_filename(filename)`[source] **Pass it a filename and it will return a secure version of it.** This filename can then safely be stored on a regular file system and passed to os.

7. Math

This module provides access to the mathematical functions defined by the C standard.

Tools used and things to know

1. SQL BACKEND INTERGRATION

The entire backend of the website is based on MYSQL. Here we have 2 tables.

- a. **Blogs** - Holding the blog's id, heading and content.
- b. **Contact Us** - Holding the contact id, email, name and content.

2. HOMEPAGE

On the homepage, there are recently uploaded blogs that can be accessed by clicking on them.

3. PAGINATION

The pagination logic is made so that we can click on the “next” or “previous” button to access older and newer blogs.

Pagination Logic:

1. First page
prev=#
next=page+1
2. Middle page
prev=page-1
next=page+1
3. Last page
prev=page-1
next=#

4. CONTACT US PAGE

The contact us page can be used to connect with the admin of the website. It is set in such a way that when you fill out the contact us form, an email will be sent to the admin with your details and message in it.

5. BOOTSTRAP

The entire front end of the website is based on the Bootstrap CSS utility.

6. CONFIG.JSON

The config.json file consists of all the elements that are required to be security taken. Mainly, email id and password for sending emails, etc.

7. Authentication

We've done basic authentication. The username and password are stored in the config.json file. The user can log in by going to localhost/dashboard.

Code.

Main.py

```
from flask import Flask, render_template,  
request, session, redirect  
from flask_sqlalchemy import SQLAlchemy  
import json  
from flask_mail import Mail # for sending emails  
import os  
from werkzeug.utils import secure_filename #  
update, this is a new import library for  
uploading files  
import math #for pagination purposes  
  
with open('config.json', 'r') as c:  
    params = json.load(c)["params"]  
  
local_server = True  
app = Flask(__name__)  
app.secret_key = 'SECRET KEY'  
app.config['UPLOAD_FOLDER'] =  
params['upload_location']  
  
'''  
sending mails  
'''  
app.config.update(  
    MAIL_SERVER='smtp.gmail.com',  
    MAIL_PORT='465',
```



```

        MAIL_USE_SSL=True,
        MAIL_USERNAME=params['gmail_user'],
        MAIL_PASSWORD=params['gmail_password']
    )
mail=Mail(app)
if(local_server):
    app.config['SQLALCHEMY_DATABASE_URI'] =
params['local_uri']
else:
    app.config['SQLALCHEMY_DATABASE_URI'] =
params['prod_uri']
db = SQLAlchemy(app)

class Contacts(db.Model):
    sno = db.Column(db.Integer,
primary_key=True)
    name = db.Column(db.String(80),
nullable=False)
    email = db.Column(db.String(120),
nullable=False)
    phone_num = db.Column(db.String(120),
nullable=False)
    mes = db.Column(db.String(120),
nullable=False)

class Posts(db.Model):
    sno = db.Column(db.Integer,
primary_key=True)
    title = db.Column(db.String(80),
nullable=False)

```

```

        content = db.Column(db.String(120),
nullable=False)
        slug = db.Column(db.String(120),
nullable=False)

@app.route("/")
def home():
    posts = Posts.query.filter_by().all()
    # [0: 5]
    last = math.ceil(len(posts)/5) #5 as in
number of posts per page
    page=request.args.get('page')
    if (not str(page).isnumeric()):
        page = 1
    page=int(page)
    #slicing the posts
    posts=posts[int(page-1)*5:int(page)*5+5]
#params['no_of_posts'] hona chahiye tha but i
didn't used it.
'''
    Pagination logic:
    1. First page
    prev=#
    next=page+1
    2. Middle page
    prev=page-1
    next=page+1
    3. Last page
    prev=page-1
    next=#

```

```

'''
#First
if (page==1):
    prev="#"
    next = "/?page=" + str(int(page) + 1)
#last
elif(page==last):
    prev= "/?page=" + str(int(page) - 1)
    next = "#"
#Middle
else:
    prev= "/?page=" + str(int(page) - 1)
    next = "/?page=" + str(int(page) + 1)

    return render_template("index.html",
params=params, posts=posts, prev=prev,
next=next)
@app.route("/dashboard", methods=['GET',
'POST'])
def dashboard():
    if ('user' in session and session['user'] ==
params ['admin_name']):
        posts = Posts.query.all()
        return render_template
('dashboard.html', params=params, posts=posts)

if request.method=='POST':
    # REDIRECT TO ADMIN
    username=request.form.get('uname')
    password=request.form.get('pass')

```

```
        if (username== params['admin_name'] and
password==params['admin_password']):
            #set the session variable
            session['user']= username
            posts = Posts.query.all()
            return
render_template("dashboard.html", params=params,
posts=posts)
    else:
        return render_template("login.html",
params=params)
```

```
@app.route("/post/<string:post_slug>",
methods=['GET']) # string:post_slug is a
variable, also important keep in note
def post_route(post_slug):
    post =
Posts.query.filter_by(slug=post_slug).first()
    return render_template('post.html',
params=params, post=post)
```

```
@app.route("/edit/<string:sno>", methods=['GET',
'POST'])
def edit(sno):
    if ('user' in session and session['user'] ==
params ['admin_name']):
        if request.method=='POST':
            box_title=request.form.get('title')
            slug=request.form.get('slug')
            content=request.form.get('content')
```

```

        if sno=='0':
            post=Posts(title=box_title,
slug=slug, content=content)
            db.session.add(post)
            db.session.commit()
        else:

post=Posts.query.filter_by(sno=sno).first()
        post.title=box_title
        post.slug=slug
        post.content=content
        db.session.commit()
        return redirect('/edit/'+sno)


post=Posts.query.filter_by(sno=sno).first()
        return render_template('edit.html',
params=params, post=post, sno=sno)


@app.route("/delete/<string:sno>",
methods=['GET', 'POST'])
def delete(sno):
    if ('user' in session and session['user']
== params ['admin_name']):

post=Posts.query.filter_by(sno=sno).first()
        db.session.delete(post)
        db.session.commit()
        return redirect("/dashboard")

```

```

@app.route("/uploader", methods=['GET', 'POST'])
def uploader():
    if ('user' in session and session
['user']==params['admin_name']):
        if (request.method=='POST'):
            f=request.files['file1']

f.save(os.path.join(app.config['UPLOAD_FOLDER'],
secure_filename(f.filename)))
            return "upload successfully"

@app.route("/logout")
def logout():
    session.pop('user') #killing usersession
    return redirect ('/dashboard')

@app.route("/about")
def about():
    return render_template('about.html',
params=params)

@app.route("/contact", methods=['GET', 'POST'])
def contact ():
    if (request.method=='POST'):
        name=request.form.get('name')
        email=request.form.get('email')
        phone_num=request.form.get('phone_num')
        mes=request.form.get('mes')
        entry= Contacts(name=name,
phone_num=phone_num, email=email, mes=mes) #isme
hamne sno nahi dala hai, so we are assuming ki

```

vo apne aap lelega, and for that, ham ek initial entry database me de denge

```
        db.session.add(entry)
        db.session.commit()
        # mail.send_message('New message from '
+ name, sender=email,
recipients=[params['_gmail_user']], body= mes +
"/n" + phone_num)

# '''
# Issue arised: no module found named "mysql
db"
# solution:
# pip install mysqlclient
# '''
return render_template('contact.html')

app.run(debug=True, host="0.0.0.0", port=5000)
#host="0.0.0.0" will make the page accessable by
going to http://[ip]:5000/ on any computer in
the network
```

Layout.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8" />
    <meta name="viewport"
content="width=device-width, initial-scale=1,
shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>Flask Blog</title>
    <link rel="icon" type="image/x-icon"
href="assets/favicon.ico" />
    <!-- Font Awesome icons (free version)-->
    <script
src="https://use.fontawesome.com/releases/v5.15.
4/js/all.js" crossorigin="anonymous"></script>
    <!-- Google fonts-->
    <link
href="https://fonts.googleapis.com/css?family=Lo
ra:400,700,400italic,700italic" rel="stylesheet"
    type="text/css" />
    <link
href="https://fonts.googleapis.com/css?family=Op
en+Sans:300italic,400italic,600italic,700italic,
800italic,400,300,600,700,800"
    rel="stylesheet" type="text/css" />
    <!-- Core theme CSS (includes Bootstrap)-->
```



```

    <link href=" {{url_for ('static',
filename='css/styles.css')}} " rel="stylesheet"
/>
</head>

<body>
    <!-- Navigation-->
    <nav class="navbar navbar-expand-lg
navbar-light" id="mainNav">
        <div class="container px-4 px-lg-5">
            <a class="navbar-brand"
href="index.html">Start Bootstrap</a>
            <button class="navbar-toggler"
type="button" data-bs-toggle="collapse"
data-bs-target="#navbarResponsive"
            aria-controls="navbarResponsive"
            aria-expanded="false" aria-label="Toggle
navigation">

                Menu

                <i class="fas fa-bars"></i>
            </button>
            <div class="collapse
navbar-collapse" id="navbarResponsive">
                <ul class="navbar-nav ms-auto
py-4 py-lg-0">

                    <li class="nav-item"><a
class="nav-link px-lg-3 py-3 py-lg-4"
href="/">Home</a></li>

                    <li class="nav-item"><a
class="nav-link px-lg-3 py-3 py-lg-4"
href="/about">About</a></li>

```

```

        <li class="nav-item"><a
class="nav-link px-lg-3 py-3 py-lg-4"
href="post.html">Sample Post</a></li>
        <li class="nav-item"><a
class="nav-link px-lg-3 py-3 py-lg-4"
href="/contact">Contact</a></li>
    </ul>
</div>
</div>
</nav>

<!-- Body content here -->
{% block body %} {% endblock %}

<!-- Footer-->
<footer class="border-top">
    <div class="container px-4 px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
            <div class="col-md-10 col-lg-8
col-xl-7">
                <ul class="list-inline
text-center">
                    <li
class="list-inline-item">
                        <a href="#!">
                            <span
class="fa-stack fa-lg">
                                <i
class="fas fa-circle fa-stack-2x"></i>

```

```

<i
class="fab fa-twitter fa-stack-1x
fa-inverse"></i>

</span>
</a>
</li>
<li
class="list-inline-item">
<a href="#"!>
<span
class="fa-stack fa-lg">
<i
class="fas fa-circle fa-stack-2x"></i>
<i
class="fab fa-facebook-f fa-stack-1x
fa-inverse"></i>

</span>
</a>
</li>
<li
class="list-inline-item">
<a href="#"!>
<span
class="fa-stack fa-lg">
<i
class="fas fa-circle fa-stack-2x"></i>
<i
class="fab fa-github fa-stack-1x
fa-inverse"></i>

</span>
</a>

```

```

                </li>
            </ul>
            <div class="small
text-center text-muted fst-italic">Copyright
&copy; Your Website 2021</div>
        </div>
    </div>
</div>
</footer>
<!-- Bootstrap core JS-->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.
3/dist/js/bootstrap.bundle.min.js"></script>
    <!-- Core theme JS-->
    <script src="js/scripts.js"></script>
</body>

</html>

```

```

HOME
{% extends "layout.html" %}
{% block body %}
<!-- Page Header-->
<header class="masthead"
style="background-image: url('{{ url_for
('static',
filename='assets/img/about-bg.jpg')}}')">
    <div class=" container position-relative
px-4 px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">

```

```

        <div class="col-md-10 col-lg-8
col-xl-7">
            <div class="page-heading">
                <h1>About Me</h1>
                <span
class="subheading">This is what I do.</span>
            </div>
        </div>
    </div>
</div>
</header>
<!-- Main Content-->
<main class="mb-4">
    <div class="container px-4 px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
            <div class="col-md-10 col-lg-8
col-xl-7">
                <p>{{params['about_text']}}</p>
            </div>
        </div>
    </div>
</main>
{% endblock %}

```

Contact us

```
{% extends "layout.html" %}
{% block body %}
<!-- Page Header-->
<header class="masthead"
style="background-image: url('{{ url_for
('static',
filename='assets/img/contact-bg.jpg')}}')">
    <div class="container position-relative px-4
px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
            <div class="col-md-10 col-lg-8
col-xl-7">
                <div class="page-heading">
                    <h1>Contact Me</h1>
                    <span
class="subheading">Have questions? I have
answers.</span>
                </div>
            </div>
        </div>
    </div>
</header>
<!-- Main Content-->
<main class="mb-4">
    <div class="container px-4 px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
```

```

        <div class="col-md-10 col-lg-8
col-xl-7">
            <p>Want to get in touch? Fill
out the form below to send me a message and I
will get back to you as soon
                as possible!</p>
            <div class="my-5">
                <!-- * * * * * * * * * * * * * *
* * *-->
                    <!-- * * SB Forms Contact
Form * *-->
                        <!-- * * * * * * * * * * * * * *
* * *-->
                            <!-- This form is
pre-integrated with SB Forms.-->
                                <!-- To make this form
functional, sign up at-->
                                    <!--
https://startbootstrap.com/solution/contact-form
s-->
                                        <!-- to get an API token!-->
                                            <form id="contactForm"
data-sb-form-api-token="API_TOKEN"
action="/contact" method="post">
                                                <div
class="form-floating">
                                                    <input
class="form-control" id="name" name="name"
type="text"

```

```

placeholder="Enter your name..."
data-sb-validations="required" />
        <label
for="name">Name</label>
        <div
class="invalid-feedback"
data-sb-feedback="name:required">A name is
required.</div>
        </div>
        <div
class="form-floating">
                <input
class="form-control" id="email" name="email"
type="email"
placeholder="Enter your email..."
data-sb-validations="required,email" />
                <label
for="email">Email address</label>
                <div
class="invalid-feedback"
data-sb-feedback="email:required">An email is
required.</div>
                <div
class="invalid-feedback"
data-sb-feedback="email:email">Email is not
valid.</div>
                </div>
        </div>
        <div
class="form-floating">

```



```
        <input
class="form-control" id="phone" name="phone_num"
type="tel"
```

```
placeholder="Enter your phone number..."
data-sb-validations="required" />
```

```
        <label
for="phone">Phone Number</label>
```

```
        <div
class="invalid-feedback"
data-sb-feedback="phone:required">A phone number
is required.
```

```
    </div>
```

```
</div>
```

```
<div
```

```
class="form-floating">
```

```
        <textarea
```

```
class="form-control" id="message" name="mes"
```

```
placeholder="Enter your message here..."
style="height: 12rem"
```

```
data-sb-validations="required"></textarea>
```

```
        <label
```

```
for="message">Message</label>
```

```
        <div
```

```
class="invalid-feedback"
data-sb-feedback="message:required">A message is
required.
```

```
    </div>
```

```
</div>
```

```

                                <br />
                                <!-- Submit success
message-->

                                <!-->
                                <!-- This is what your
users will see when the form-->
                                <!-- has successfully
submitted-->

                                <div class="d-none"
id="submitSuccessMessage">
                                <div
class="text-center mb-3">
                                <div
class="fw-bolder">Form submission
successful!</div>

                                To activate this
form, sign up at

                                <br />
                                <a

href="https://startbootstrap.com/solution/contac
t-forms">https://startbootstrap.com/solution/con
tact-forms</a>

                                </div>
                                </div>
                                <!-- Submit error
message-->

                                <!-->
                                <!-- This is what your
users will see when there is-->

```

```

                                <!-- an error submitting
the form-->

                                <div class="d-none"
id="submitErrorMessage">

                                    <div
class="text-center text-danger mb-3">Error
sending message!</div>

                                </div>
                                <!-- Submit Button-->
                                <button class="btn
btn-primary text-uppercase" id="submitButton"
type="submit">Send</button>
                                    </form>
                                </div>
                            </div>
                        </div>
                    </div>
                </main>
            {% endblock%}

```

Dashboard

```
{% extends "layout.html" %}
{% block body %}
<!-- Page Header-->
<header class="masthead"
style="background-image: url(' {{ url_for
('static', filename='assets/img/home-bg.jpg') }}
') ">
    <div class="container position-relative px-4
px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
            <div class="col-md-10 col-lg-8
col-xl-7">
                <div class="site-heading">
                    <h1>Logged in</h1>
                    <span class="subheading">A
Blog Theme by Start Bootstrap</span>
                </div>
            </div>
        </div>
    </div>
</header>

<div style="margin: 10rem;">
    <h1>Basic actions </h1>
    <a href="/edit/0"><button>add a new
post</button></a>
```

```
    <a
href="/logout"><button>Logout</button></a>
</div>
```

```
<div class="file" style="margin: 10rem;">
    <h2>Upload a File</h2>
    <form action="/uploader" method="post"
enctype="multipart/form-data">
        <input type="file" name="file1" id="">
        <button type="submit">Submit</button>
    </form>
</div>
```

```
<!-- Main Content-->
<table class="table" style="margin: 10rem;
margin-right: 10rem;">
    <thead>
        <tr>
            <th scope="col">sno</th>
            <th scope="col">title</th>
            <th scope="col">slug</th>
            <th scope="col">edit</th>
            <th scope="col">delete</th>

        </tr>
    </thead>
    <tbody>
        <!-- Jenga for loop -->
        {% for post in posts %}
```

```

        <tr>
            <th scope="row">{{post.sno}}</th>
            <td>{{post.title}}</td>
            <td>{{post.slug}}</td>
            <td> <a href="/edit/{{post.sno}}">
<button>edit</button> </a> </td>
            <td><a
href="/delete/{{post.sno}}"><button>delete</butt
on></a></td>
        </tr>

        {% endfor %}

    </tbody>
</table>

{% endblock %}

```

Edit.html

```
{% extends "layout.html" %}
{% block body %}
<!-- Page Header-->
<header class="masthead"
style="background-image: url('{{ url_for
('static',
filename='assets/img/contact-bg.jpg') }}')">
    <div class="container position-relative px-4
px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
            <div class="col-md-10 col-lg-8
col-xl-7">
                <div class="page-heading">
                    <h1>Logged In</h1>
                    <!-- <span
class="subheading"></span> -->
                </div>
            </div>
        </div>
    </div>
</header>
<div style="margin: 10rem;">
    <h1>Basic actions </h1>
    <a href="/dashboard"><button>Go to
dashboard</button></a>
    <a
href="/logout"><button>Logout</button></a>
</div>
<hr>
```

```

<!-- Main Content-->
<main class="mb-4">
    <div class="container px-4 px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
            <div class="col-md-10 col-lg-8
col-xl-7">
                <p>Want to get in touch? Fill
out the form below to send me a message and I
will get back to you as soon
                    as possible!</p>
                <div class="my-5">
                    <form id="contactForm"
data-sb-form-api-token="API_TOKEN"
action="/edit/{{post.sno}}">
                        method="post">
                            <div
class="form-floating">
                                <input
class="form-control" id="title" name="title"
type="text"

placeholder="Enter your name
data-sb-validations=" required
value="{{post.title}}" />
                                    <label
for="name">Title</label>
                                        <div
class="invalid-feedback"
data-sb-feedback="name:required">A name is
required.</div>

```



```

        </div>
        <div
class="form-floating">
            <input
class="form-control" id="slug" name="slug"
type="text"

placeholder="Enter your email..."
data-sb-validations="required,email"

value="{{post.slug}}" />
            <label
for="slug">slug</label>
            <div
class="invalid-feedback"
data-sb-feedback="email:required">An email is
required.</div>
            <div
class="invalid-feedback"
data-sb-feedback="email:email">Email is not
valid.</div>
        </div>
        <div
class="form-floating">
            <input
class="form-control" id="content" name="content"
type="text"

placeholder="Enter your phone number..."
data-sb-validations="required"

```

```

value="{{post.content}}" />
                                <label
for="content">Content</label>
                                <div
class="invalid-feedback"
data-sb-feedback="phone:required">A phone number
is required.
                                </div>
                                </div>
                                <br />
                                <!-- Submit success
message-->
                                <!------>
                                <!-- This is what your
users will see when the form-->
                                <!-- has successfully
submitted-->
                                <div class="d-none"
id="submitSuccessMessage">
                                <div
class="text-center mb-3">
                                <div
class="fw-bolder">Form submission
successful!</div>
                                To activate this
form, sign up at
                                <br />
                                <a
href="https://startbootstrap.com/solution/contac

```

```

t-forms">https://startbootstrap.com/solution/con
tact-forms</a>

        </div>
    </div>
    <!-- Submit error
message-->

        <!------>
        <!-- This is what your
users will see when there is-->
        <!-- an error submitting
the form-->

        <div class="d-none"
id="submitErrorMessage">
            <div
class="text-center text-danger mb-3">Error
sending message!</div>
            </div>
            <!-- Submit Button-->
            <button class="btn
btn-primary text-uppercase" id="submitButton"
type="submit">Send</button>
        </form>
    </div>
</div>
</div>
</main>
{% endblock%}

```

INDEX.HTML

```
{% extends "layout.html" %}
{% block body %}
<!-- Page Header-->
<header class="masthead"
style="background-image: url(' {{ url_for
('static', filename='assets/img/home-bg.jpg') }}
')">
    <div class="container position-relative px-4
px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
            <div class="col-md-10 col-lg-8
col-xl-7">
                <div class="site-heading">
                    <h1>Clean Blog</h1>
                    <span class="subheading">A
Blog Theme by Start Bootstrap</span>
                </div>
            </div>
        </div>
    </div>
</header>
<!-- Main Content-->
<div class="container px-4 px-lg-5">
    <div class="row gx-4 gx-lg-5
justify-content-center">
        <div class="col-md-10 col-lg-8
col-xl-7">
            <!-- Post preview-->
```

```

        {% for post in posts %}
        <div class="post-preview">
            <a href="/post/{{post.slug}}">
                <h2 class="post-title">{{
post.title }}</h2>
                <h3 class="post-subtitle">{{
post.content[0:13]}}.</h3>
            </a>
            <p class="post-meta">
                Posted by
                <a href="#">Start
Bootstrap</a>
                post.date
            </p>
        </div>
        <!-- Divider-->
        <hr class="my-4" />
        {% endfor %}
        <!-- Pager-->
        <a class="btn btn-primary
float-left" href="{{prev}}">prev Posts
&rarr;</a>

        <a class="btn btn-primary
float-right" href="{{next}}">Older Posts
&rarr;</a>

    </div>
</div>

```

```
{% endblock %}
```

Login.html

```
<!doctype html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta name="viewport"
content="width=device-width, initial-scale=1,
shrink-to-fit=no">
  <meta name="description" content="">
  <meta name="author" content="">
  <link rel="icon"
href="../../../../favicon.ico">

  <title>Signin Template for Bootstrap</title>

  <!-- Bootstrap core CSS -->
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstr
ap/4.1.3/css/bootstrap.min.css"

integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27
NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
crossorigin="anonymous">

  <!-- Custom styles for this template -->
  <link href="signin.css" rel="stylesheet">
</head>

<body class="text-center">
```

```
<link href="{{url_for('static',
filename='css/signin.css')}}" rel="stylesheet">
```

```
<form class="form-signin" action="/dashboard"
method="post" action="/dashboard">
```

```
<h1 class="h3 mb-3 font-weight-normal">Admin
Login</h1>
```

```
<label for="uname" class="sr-only"
name="uname">Email address</label>
```

```
<input type="text" id="uname"
class="form-control" name="uname"
placeholder="username" required autofocus>
```

```
<label for="pass"
class="sr-only">Password</label>
```

```
<input type="password" id="pass" name="pass"
class="form-control" placeholder="Password"
required>
```

```
<div class="checkbox mb-3">
<label>
<input type="checkbox"
value="remember-me"> Remember me
</label>
```

```
</div>
<button class="btn btn-lg btn-primary
btn-block" type="submit">Sign in</button>
```

```
<p class="mt-5 mb-3 text-muted">©
{{params['blog_name']} } 2017-2018</p>
</form>
```


</body>

</html>

POST.html

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

```
{% block body %}
```

```
<!-- Page Header-->
```

```
<header class="masthead"
```

```
style="background-image:
```

```
url('assets/img/post-bg.jpg') ">
```

```
    <div class="container position-relative px-4  
px-lg-5">
```

```
        <div class="row gx-4 gx-lg-5  
justify-content-center">
```

```
            <div class="col-md-10 col-lg-8  
col-xl-7">
```

```
                <div class="post-heading">
```

```
                    <h1>{{post.title}}</h1>
```

```
                    <h2
```

```
class="subheading">Problems look mighty small  
from 150 miles up</h2>
```

```
                <span class="meta">
```

```
                    Posted by
```

```
                    <a href="#!">Start
```

```
Bootstrap</a>
```

```
                    on August 24, 2021
```

```
                </span>
```

```
            </div>
```

```
        </div>
```

```
        </div>
    </div>
</header>
<!-- Post Content-->
<article class="mb-4">
    <div class="container px-4 px-lg-5">
        <div class="row gx-4 gx-lg-5
justify-content-center">
            <div class="col-md-10 col-lg-8
col-xl-7">
                <p>{{post.content}}</p>

            </div>
        </div>
    </div>
</article>
{% endblock %}
```

Config.json

```
{
  "params": {
    "local_server": "True",
    "local_uri":
"mysql://root:asw2842@localhost/codeblog",
    "prod_uri":
"mysql://root:asw2842@localhost/codeblog",
    "facebook_url":
"https://www.facebook.com/",
    "twitter_url": "https://twitter.com/",
    "github_url": "https://github.com",
    "gmail_user": "emaim@gmail.com",
    "gmail_password": "*****",
    "about_text": "This is the about
content, hey this is akshat. This page is loaded
from config.json file",
    "number_of_posts": "5",
    "admin_name": "admin",
    "admin_password": "admin",
    "upload_location":
"C:\\Users\\User\\Desktop\\MyProjects\\flask\\static\\assets\\img\\"
  }
}
```

Blog.sql

```
-- MySQL dump 10.13  Distrib 8.0.28, for Win64
(x86_64)
--
-- Host: localhost      Database: codeblog
--
-----
-----
-- Server version 8.0.28

/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT
*/;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESUL
TS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION
*/;
/*!50503 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
UNIQUE_CHECKS=0 */;
/*!40014 SET
@OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
```

```

/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES,
SQL_NOTES=0 */;

--
-- Table structure for table `contacts`
--

DROP TABLE IF EXISTS `contacts`;
/*!40101 SET @saved_cs_client      =
@@character_set_client */;
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `contacts` (
  `name` varchar(225) DEFAULT NULL,
  `email` varchar(225) DEFAULT NULL,
  `phone_num` varchar(225) DEFAULT NULL,
  `mes` text
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client =
@saved_cs_client */;

--
-- Dumping data for table `contacts`
--

LOCK TABLES `contacts` WRITE;
/*!40000 ALTER TABLE `contacts` DISABLE KEYS */;
INSERT INTO `contacts` VALUES
('test1','akshat@gmail.com','8247284738','test1'
),('hdajhda','jhads@jh.com','jha','jkhda');
/*!40000 ALTER TABLE `contacts` ENABLE KEYS */;

```

```
UNLOCK TABLES;
```

```
--
```

```
-- Table structure for table `posts`
```

```
--
```

```
DROP TABLE IF EXISTS `posts`;
```

```
/*!40101 SET @saved_cs_client      =  
@@character_set_client */;
```

```
/*!50503 SET character_set_client = utf8mb4 */;
```

```
CREATE TABLE `posts` (  
  `sno` int NOT NULL AUTO_INCREMENT,
```

```
  `title` varchar(45) DEFAULT NULL,
```

```
  `content` text,  
  `slug` varchar(45) DEFAULT NULL,
```

```
  PRIMARY KEY (`sno`),
```

```
  UNIQUE KEY `sno_UNIQUE` (`sno`)
```

```
) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT  
CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
/*!40101 SET character_set_client =
```

```
@saved_cs_client */;
```

```
--
```

```
-- Dumping data for table `posts`
```

```
--
```

```
LOCK TABLES `posts` WRITE;
```

```
/*!40000 ALTER TABLE `posts` DISABLE KEYS */;
```

```
INSERT INTO `posts` VALUES
```

```
(4, '4', '4', '4'), (5, '5', '5', '5'), (6, '6', '6', '6'),
```

```
(7,'7','7','7'),(15,'8','8','8'),(16,'9','9','9')
);
/*!40000 ALTER TABLE `posts` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET
FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS
*/;
/*!40101 SET
CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT
*/;
/*!40101 SET
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS
*/;
/*!40101 SET
COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION
*/;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2022-02-23 18:44:03
```