Login

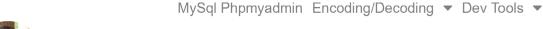


Dev Forum ▼

Search article

Q

Register





Sheetal Kumar Expert in PHP, MySql, Android, Python, Javascript, React February 1, 2019

Create Python Flask App in MVC Format

Flask is a microframework written in Python. It helps to create a small app to large scale web application.

Advantages of using flask framework are as follows:

- 1. Easy installation
- 2. Small size
- 3. Large community support
- 4. Fast in execution

In this article, we will create a simple blog to understand the structure and code pattern of flask framework.

Create a flask project







Login

Register

MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼

https://www.python.org/downloads/

Install PIP that is a package installer in python.

```
curl "https://bootstrap.pypa.io/get-pip.py" -o "get-
pip.py"

python get-pip.py
```

Install environment for flask app.

```
pip install Flask
pip install virtualenv
```

Now create your application directory on your desired location. In our example this is flaskblog.

Clone Flask MVC Structure from Github:

https://github.com/sheetalkumar105/flaskmvc.git

```
git clone https://github.com/sheetalkumar105
/flaskmvc.git
```



Dev Forum ▼ Q Login Register

MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼

Login

Register

Q

MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼

Dev Forum ▼

DavStudio The Developers Choice Lapp
L controller
Linitpy
L HomeController.py
L helpers
Linitpy
L Utility.py
L models
Linitpy
L AuthModel.py
L AuthModel.py L routes
L routes
L routes Linitpy

L views



Dev Forum ▼



MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼

Login

Register

L app.py

L config.py

L setup.py

Now start Flask App:

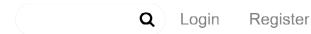
```
sudo pip install -e .
export FLASK_APP=app/__init__.py
export FLASK_DEBUG=1
flask run
```

You will get the following message:

- * Serving Flask app "app"
- * Forcing debug mode on
- * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
- * Restarting with stat







* Debugger PIN: 114-170-438 Phpmyadmin Encoding/Decoding Dev Tools

Now open browser and open link: http://127.0.0.1:5000/

To change the database connection open app/config.py and configure

```
config['MONGO_DBNAME'] = 'flaskblog'
config['MONGO_URI'] = 'mongodb://localhost:27017
/flaskblog'
```

Now create a post model to retrieve posts. Create a file PostModel.py in models directory.

Write the code inside that:

Register

```
The Developers Choice
                                                               Q Login
                         Dev Forum ▼
from flask_pymongo import ObjectId
                         MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼
from app.helpers.Utility import toDictionaryArray
from datetime import datetime
class PostModel():
  def __init__(self):
      pass
 def getPost(self,_id):
      users =
mongo.db.posts.find({'_id':ObjectId(_id)})
      post=toDictionaryArray(users)
      return post[0]
  def getAllPosts(self):
      users = mongo.db.posts.find()
      return toDictionaryArray(users)
 def addPost(self, title, content,image):
      d = datetime.now()
      insertdata = {'title': title, 'content':
content, 'image': image, 'status': "1", 'created_at':
d, 'updated_at': d}
      res = mongo.db.posts.insert(insertdata)
```

Register

```
MySql Phpmyadmin Encoding/Decoding  Dev Tools  

def updatePost(self, _id,title, content,image):
    d = datetime.now()
    updatedata = {'title': title, 'content':
    content, 'image': image, 'status': "1", 'updated_at':
    d}
    mongo.db.posts.update({'_id': ObjectId(_id)},
    {'$set': updatedata}, False, True)
    return _id

def deletePost(self, _id):
    mongo.db.posts.delete_many({'_id': ObjectId(_id)})
```

Add in helpers/Utility.py

postmodel=PostModel()

```
def allowed_file(filename, ALLOWED_EXTENSIONS):
    return '.' in filename and \
        filename.rsplit('.', 1)[1].lower() in
ALLOWED_EXTENSIONS
```



Dev Forum ▼ Q Login Register

MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼

Q Login Register

return render_template('registration.html',

return render_template('createpost.html',

title='Registration')

def new(self):

title='New Post')

```
The Developers Choice
                                                                   Login
                                                                           Register
                         Dev Forum ▼
                         MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼
      _firstname = request.form.get('firstname', '')
      _lastname = request.form.get('lastname', '')
      _email = request.form.get('email', '')
      _password = request.form.get('password', '')
      return
sendResponse(authmodel.registerUser(_firstname,_lastname
 def writePost(self):
      title=request.form.get("title","")
      content=request.form.get("content","")
      image = request.files['image']
      filename=''
      if image and allowed_file(image.filename,
set(['png', 'jpg', 'jpeg', 'gif'])):
          ilename, file_extension =
os.path.splitext(image.filename)
          millis = int(round(time.time() * 1000))
          filename = str(millis) + file_extension
          image.save(os.path.join("app/static/images",
filename))
      _id=postmodel.addPost(title,content,filename)
      return redirect("/")
```





MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼

Add in Front Router:

```
@front.route('/registration', methods=['POST'])

def registeruser():
    return homecontroller.registeruser()

@front.route('/new', methods=['GET'])

def newpost():
    return homecontroller.new()

@front.route('/new', methods=['POST'])

def writePost():
    return homecontroller.writePost()
```

Now create views inside views directory that uses word Search Game Terms & Conditions Privacy Policy jinja Template Engine.

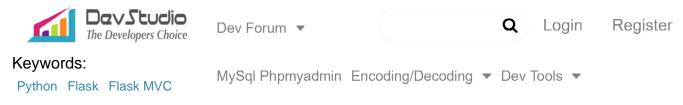
createpost.html

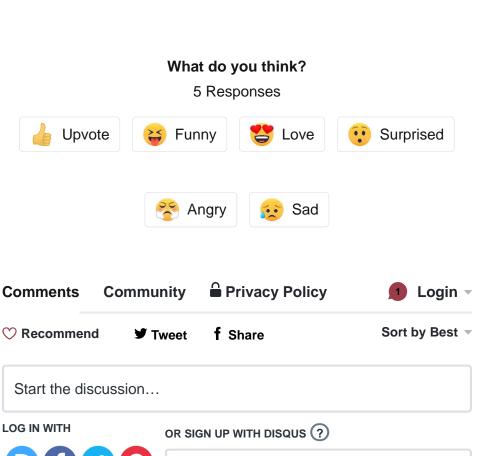
```
DavStudio
                                                               Q
                                                                    Login
                                                                             Register
                         Dev Forum ▼
        The Developers Choice
  <title>{{ title }} - New Post</title>
                         MySql Phpmyadmin Encoding/Decoding ▼ Dev Tools ▼
</head>
<body>
    <a href="/">Home</a>
                            <form id="registration"
method="post" enctype="multipart/form-data">
    Title: <input type="text" name="title" /><br>
    Content: <input type="text" name="content" /><br>
    Image: <input type="file" name="image" /><br>
    <button type="submit" id="btnSubmit">Submit</button>
  </form>
</body>
</html>
```

index.html

All setup is done. Open link http://127.0.0.1:5000/new to create a new post.

All the post will view on http://127.0.0.1:5000/





Be the first to comment.

Name

List and List operations, iteration, traversal in Python
Implode (join) and explode (split) in python
List of Keywords in Python and their uses Functions and their use in Python
Useful program in python Create JSON and XML in python
Create Python Flask App in MVC Format
Deploy python flask app on Linux Server Merge two arrays in python
Get length of String or get count of List in Python
Check key exist or not in dictionary in python Basic use of Machine learning
Mean Median Mode in Machine Learning Percentiles in Machine learning
Use of Standard Deviation in machine learning
Use of Data Distribution in machine learning