



Flask project – Create a Joke App with PyJokes

Last Updated : 26 May, 2022

Flask is a micro web framework written in Python. It is classified as a micro-framework because it does not require particular tools or libraries. Flask is a lightweight WSGI web application framework. It is designed to make getting

[Flask Templates](#) [Jinja2](#) [Flask-REST API](#) [Python SQLAlchemy](#) [Flask Bcrypt](#) [Flask Cookies](#) [Json](#) [Postman](#)

For creating joke app we need two libraries flask and pyjokes.

Installation –

First we install flask

```
pip install flask
```

Second library installation

```
pip install pyjokes
```

You get funny one-liner, mostly related to programming by using just importing a library known as pyjokes.

Some Methods Of pyjokes Library

There are two major methods in pyjokes- `get_joke()` and `get_jokes()`.

- **`get_joke()`**– It only returns one joke. We get random joke each time.
- **Parameters** – There are two parameters- language and category. You can choose from the language and category above.
- **Return Type** – It return string type (str).
- **`get_jokes()`** – Here, we get a list of jokes.
- **Parameter**– The parameters are same as above- language and category.
- **Return type**– list.

Languages Supported By pyjokes:

- English – 'en'

- German – ‘de’
- Spanish – ‘es’
- Galician – ‘gl’
- Basque – ‘eu’
- Italian – ‘it’

Categories Included In pyjokes:

- For geeky jokes - ‘neutral’ (It is chosen by default)
- For Chris Norris Jokes – ‘chuck’.
- If you want all type of jokes – ‘all’
- There is one more category known as ‘twister’ which only works for the German Language (‘de’). This mostly includes tongue twister.

Creating Flask Project –

I have created a file and name it as app.py

app.py

Python3

```
from flask import Flask
import pyjokes

app=Flask(__name__)

@app.route("/")
def home():
    joke=pyjokes.get_joke() #It only returns one joke. We get random joke each ti
    return f'<h2>{joke}</h2>'

@app.route("/MultipleJokes")
def jokes():
    jokes=pyjokes.get_jokes() #Here, we get a list of jokes.
    return f'<h2>{jokes}</h2>'

if __name__ == "__main__":
    app.run(debug=True)
```

To run flask app you need run this command in your command prompt or terminal:

For windows

```
python app.py
```

For ubuntu

```
python3 app.py
```

After running the app you will get output like this

Output on “http://127.0.0.1:5000”



Output on “http://127.0.0.1:5000/MultipleJokes”



Looking to dive into the world of programming or sharpen your Python skills?

Our [Master Python: Complete Beginner to Advanced Course](#) is your ultimate guide to becoming proficient in Python. This course covers everything you need to build a solid foundation from fundamental programming concepts to advanced techniques. With **hands-on projects**, real-world examples, and expert guidance, you'll gain the confidence to tackle complex **coding challenges**. Whether you're starting from scratch or aiming to enhance your

skills, this course is the perfect fit. Enroll now and master Python, the language of the future!

v vivek...



Previous Article

How to write a simple Flask API for hello world?

Next Article

Create a Weather app using Flask | Python

Similar Reads

Python Script to create random jokes using pyjokes

Python supports creation of random jokes using one of its libraries. Let us explore it a little more, Pyjokes is a python library that is used to create one-line jokes fo...

2 min read

Joke Application Project Using Django Framework

Django is a high-level Python-based Web Framework that allows rapid development and clean, pragmatic design. It is also called batteries included...

2 min read

Joke App using Bottle Framework - Python

There are many frameworks in python which allows you to create webpage like bottle, flask, django. In this article you will learn how to create simple app...

2 min read

Documenting Flask Endpoint using Flask-Autodoc

Documentation of endpoints is an essential task in web development and being able to apply it in different frameworks is always a utility. This article discusses...

4 min read

How to use Flask-Session in Python Flask ?

Flask Session - Flask-Session is an extension for Flask that supports Server-side Session to your application. The Session is the time between the client logs in to...

4 min read

How to Integrate Flask-Admin and Flask-Login

In order to merge the admin and login pages, we can utilize a short form or any other login method that only requires the username and password. This is know...

8 min read

Minify HTML in Flask using Flask-Minify

Flask offers HTML rendering as output, it is usually desired that the output HTML should be concise and it serves the purpose as well. In this article, we would...

12 min read

Flask URL Helper Function - Flask url_for()

In this article, we are going to learn about the flask url_for() function of the flask URL helper in Python. Flask is a straightforward, speedy, scalable library, used f...

11 min read

Create a Weather app using Flask | Python

Prerequisite : Flask installation Flask is a lightweight framework written in Python. It is lightweight because it does not require particular tools or libraries...

2 min read

Create a Real Time Currency Converter app using Flask | Python

Prerequisite: installation of Python What is Flask in Python ? Flask is a popular and lightweight Python web framework, meaning it is a third-party Python libra...

8 min read

Article Tags : [Python](#) [Flask Projects](#) [Python Flask](#)

Practice Tags : [python](#)

Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305



Company

About Us
Legal
In Media
Contact Us
Advertise with us
GFG Corporate Solution
Placement Training Program
GeeksforGeeks Community

Languages

Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial
Tutorials Archive

DSA

Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems
DSA Roadmap
Top 100 DSA Interview Problems
DSA Roadmap by Sandeep Jain
All Cheat Sheets

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning
ML Maths
Data Visualisation
Pandas
NumPy
NLP
Deep Learning

Web Technologies

HTML
CSS
JavaScript
TypeScript
ReactJS
NextJS
Bootstrap
Web Design

Python Tutorial

Python Programming Examples
Python Projects
Python Tkinter
Web Scraping
OpenCV Tutorial
Python Interview Question
Django

Computer Science

DevOps

Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths
Software Development
Software Testing

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

Git
Linux
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

Interview Preparation

Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

GeeksforGeeks Videos

DSA
Python
Java
C++
Web Development
Data Science
CS Subjects

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved