Skip to content DSA Tutorials ML & Data Science Web Development Practice Sign In Sign In Home Saved Videos Courses For Working Professionals For Students For School Students Data Structures & Algorithms Data Structures Linked List Tree Algorithms Analysis of Algorithms Searching Algorithms Sorting Algorithms Company Wise SDE Sheets DSA Cheat Sheets Top Interview Questions Puzzles System Design System Design Tutorial Software Design Patterns Interview Corner Languages Web Development CSS Frameworks JavaScript Frameworks JavaScript Libraries Web Dev Cheat Sheets ML & Data Science Machine Learning Data Science Packages Data Visualization Data Visualization with Python Data Visualization with R Data Visualization Tools Data Analysis Deep Learning Interview Questions ML Frameworks CS Subjects DevOps School Learning Mathematics CBSE Syllabus Maths Notes (Class 8-12) Maths Formulas (Class 8 -11) NCERT Solutions RD Sharma Solutions Science Notes Physics Notes (Class 8-12) Chemistry Notes (Class 8-12) Biology Notes Social Science Syllabus Social Science Notes SS Notes (Class 7-12) CBSE History Notes (Class 7-10) CBSE Geography Notes (Class 7-10) CBSE Civics Notes (Class 7-10) CBSE Previous Year Papers Commerce Business Studies Accountancy Microeconomics Statistics for Economics Macroeconomics Indian Economic Development CBSE Class 12 Previous Years Papers GATE Other CS Exams ISRO UGC NET Banking Exams Study Material SBI Clerk SBI PO IBPS PO IBPS Clerk UPSC Study Material History Notes SSC CGL Study Material Student For Job Seekers For Recruiters Curated DSA Lists Tutorials Jobs Practice Contests GBlog Puzzles What's New ? Change Language Trending Now Data Structures Algorithms Topic-wise Practice Python Machine Learning Data Science JavaScript Java Web Development Bootstrap C C++ ReactJS Competitive Programming Aptitude Puzzles Projects Related Articles Write an Interview Experience Share Your Campus Experience Flask Tutorial Flask Setup & Installation Flask Quick Start Flask â€“ (Creating first simple application) How to Run a Flask Application Flask App Routing Flask â€“ HTTP Method Flask â€“ Variable Rule Redirecting to URL in Flask Python Flask â€“ Redirect and Errors How to Change Port in Flask app Changing Host IP Address in Flask Serve Templates and Static Files in Flask User Registration, Login, and Logout in Flask Define and Access the Database in Flask Flask Deployment and Error Handling Flask Projects Write an Interview Experience Share Your Campus Experience Flask Tutorial Flask Setup & Installation Flask Quick Start Flask â€“ (Creating first simple application) How to Run a Flask Application Flask App Routing Flask â€“ HTTP Method Flask â€“ Variable Rule Redirecting to URL in Flask Python Flask â€“ Redirect and Errors How to Change Port in Flask app Changing Host IP Address in Flask Serve Templates and Static Files in Flask User Registration, Login, and Logout in Flask Define and Access the Database in Flask Flask Deployment and Error Handling Flask Projects Flask â€“ (Creating first simple application) Read Discuss Courses Practice Building a webpage using python. There are many modules or frameworks which allow building your webpage using python like a bottle, Django, Flask, etc. But the real popular ones are Flask and Django. Django is easy to use as compared to Flask but Flask provides you with the versatility to program with. To understand what Flask is you have to understand a few general terms.Â WSGI Web Server Gateway Interface (WSGI) has been adopted as a standard for Python web application development. WSGI is a specification for a universal interface between the web server and the web applications.Â Werkzeug It is a WSGI toolkit, which implements requests, response objects, and other utility functions. This enables building a web framework on top of it. The Flask framework uses Werkzeug as one of its bases. jinja2 jinja2 is a popular templating engine for Python. A web templating system combines a template with a certain data source to render dynamic web pages.Â Flask is a web application framework written in Python. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects. Installation: We will require two packages to set up your environment. virtualenv for a user to create multiple Python environments side-by-side. Thereby, it can avoid compatibility issues between the different versions of the libraries and the next will be Flask itself.Â virtualenvÂ pip install virtualenv Create Python virtual environment virtualenv venv Activate virtual environment windows > venv\Scripts\activate linux > source ./venv/bin/activate For windows, if this is your first time running the script, you might get an error like below: venv\Scripts\activate : File C:\flask_project\venv\Scripts\Activate.ps1 cannot be loaded because running scripts is disabled on this system. For Â more information, see about_Execution_Policies at https:/go.microsoft.com/fwlink/?LinkID=135170. At line:1 char:1 + venv\Scripts\activate Â Â + FullyQualifiedErrorId : UnauthorizedAccess This means that you donâ€™t have access to execute the scripts. To solve this error, run the powershell as admin, when you right click on powershell icon, choose the option â€˜Run as adminstratorâ€™. Now, the powershell will open in the admin mode. Type the following command in Shell set-executionpolicy remotesigned Now, you will be prompted to change the execution policy. Please type A. This means Yes to all. FlaskÂ pip install Flask After completing the installation of the package, letâ€™s get our hands on the code. Python3 # Importing flask module in the project is mandatory # An object of Flask class is our WSGI application. from flask import Flask Â # Flask constructor takes the name of # current module (__name__) as argument. app = Flask(__name__) Â # The route() function of the Flask class is a decorator, # which tells the application which URL should call # the associated function. @app.route('/') # â€˜/â€™ URL is bound with hello_world() function. def hello_world(): Â Â Â Â return 'Hello World' Â # main driver function if __name__ == '__main__': Â Â Â Â Â # run() method of Flask class runs the application Â Â Â Â # on the local development server. Â Â Â Â app.run() Save it in a file and then run the script we will be getting an output like this. Then go to the URL given there you will see your first webpage displaying hello world there on your local server. Digging further into the context, the route() decorator in Flask is used to bind a URL to a function. Now to extend this functionality our small web app is also equipped with another method add_url_rule() which is a function of an application object that is also available to bind a URL with a function as in the above example, route() is used.Â Example:Â def gfg(): return â€˜geeksforgeeksâ€™ app.add_url_rule(â€˜/â€™, â€˜g2gâ€™, gfg) Output:Â geeksforgeeks You can also add variables in your web app, well you might be thinking about how itâ€™ll help you, itâ€™ll help you to build a URL dynamically. So letâ€™s figure it out with an example. Python3 from flask import Flask app = Flask(__name__) Â @app.route('/hello/') def hello_name(name): Â Â Â return 'Hello %s!' % name Â if __name__ == '__main__': Â Â Â app.run() And go to the URL http://127.0.0.1:5000/hello/geeksforgeeks itâ€™ll give you the following output. We can also use HTTP methods in Flask letâ€™s see how to do that The HTTP protocol is the foundation of data communication on the world wide web. Different methods of data retrieval from specified URL are defined in this protocol. The methods are described down below. GET : Sends data in simple or unencrypted form to the server. HEAD : Sends data in simple or unencrypted form to the server without body.Â HEAD : Sends form data to the server. Data is not cached. PUT : Replaces target resource with the updated content. DELETE : Deletes target resource provided as URL. By default, the Flask route responds to the GET requests. However, this preference can be altered by providing methods argument to route() decorator. In order to demonstrate the use of the POST method in URL routing, first, let us create an HTML form and use the POST method to send form data to a URL. Now letâ€™s create an HTML login page. Below is the source code of the file:Â HTML Â Â Â Â Â Â Â Â
Â Â Â Â Â Â
Â Â Â Â Â Â Â Â Â Â Â Â

Enter Name:

Â Â ÂÂÂÂÂÂÂÂÂÂ Â

Â Â ÂÂÂÂÂÂÂÂÂÂ Â

submit

Â Â ÂÂÂÂÂÂ

Â Â Â Â Â Â Â Â Now save this file HTML and try this python script to create the server. Python3 from flask import Flask, redirect, url_for, request app = Flask(__name__) Â Â @app.route('/success/') def success(name): Â Â Â Â return 'welcome %s' % name Â Â @app.route('/login', methods=['POST', 'GET']) def login(): Â Â Â Â if request.method == 'POST': Â Â Â Â Â Â Â Â user = request.form['nm'] Â Â Â Â Â Â Â Â return redirect(url_for('success', name=user)) Â Â Â Â else: Â Â Â Â Â Â Â Â user = request.args.get('nm') Â Â Â Â Â Â Â Â return redirect(url_for('success', name=user)) Â Â if __name__ == '__main__': Â Â Â Â app.run(debug=True) After the development server starts running, open login.html in the browser, enter your name in the text field and click submit button. The output would be the following. The result will be something like this And thereâ€™s much more to Flask than this. If you are interested in this web framework of Python you can dig into the links provided down below for further information. This article is contributed by Subhajit Saha. If you like GeeksforGeeks and would like to contribute, you can also write an article using write.geeksforgeeks.org or mail your article to review-team@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks. Please write comments if you find anything incorrect, or if you want to share more information about the topic discussed above. Last Updated : 15 Apr, 2023 50 Similar Reads Documenting Flask Endpoint using Flask-Autodoc How to use Flask-Session in Python Flask ? Minify HTML in Flask using Flask-Minify Connect Flask to a Database with Flask-SQLAlchemy How to Integrate Flask-Admin and Flask-Login Flask URL Helper Function - Flask url_for() How to Build a Simple Android App with Flask Backend? How to write a simple Flask API for hello world? Creating Your First Application in Python Creating first web application using Bottle Framework - Python Related Tutorials Pandas AI: The Generative AI Python Library OpenAI Python API - Complete Guide Python for Kids - Fun Tutorial to Learn Python Programming Data Analysis Tutorial Flask Tutorial Previous Handling 404 Error in Flask Next Todo list app using Flask | Python Article Contributed By : GeeksforGeeks Vote for difficulty Current difficulty : Medium Easy Normal Medium Hard Expert Improved By : eshaangupta101 miss_anonymous sainathp Article Tags : Python Web Technologies Practice Tags : python Improve Article Report Issue Learn more Join our Community Courses 147k+ interested Geeks Full Stack Development with React & Node JS - Live Explore 160k+ interested Geeks Data Structures & Algorithms in Python - Self Paced Explore 138k+ interested Geeks Complete Machine Learning & Data Science Program Explore Learn more Learn more A-143, 9th Floor, Sovereign Corporate Tower, Sector-136, Noida, Uttar Pradesh - 201305 feedback@geeksforgeeks.org Company About Us Legal Careers In Media Contact Us Advertise with us Explore Job-A-Thon For Freshers Job-A-Thon For Experienced GfG Weekly Contest Offline Classes (Delhi/NCR) DSA in JAVA/C++ Master System Design Master CP Languages Python Java C++ PHP GoLang SQL R Language Android Tutorial Data Structures Array String Linked List Stack Queue Tree Graph Algorithms Sorting Searching Greedy Dynamic Programming Pattern Searching Recursion Backtracking Web Development HTML CSS JavaScript Bootstrap ReactJS AngularJS NodeJS Computer Science GATE CS Notes Operating Systems Computer Network Database Management System Software Engineering Digital Logic Design Engineering Maths Python Python Programming Examples Django Tutorial Python Projects Python Tkinter OpenCV Python Tutorial Python Interview Question Data Science & ML Data Science With Python Data Science For Beginner Machine Learning Tutorial Maths For Machine Learning Pandas Tutorial NumPy Tutorial NLP Tutorial Deep Learning Tutorial DevOps Git AWS Docker Kubernetes Azure GCP Competitive Programming Top DSA for CP Top 50 Tree Problems Top 50 Graph Problems Top 50 Array Problems Top 50 String Problems Top 50 DP Problems Top 15 Websites for CP System Design What is System Design Monolithic and Distributed SD Scalability in SD Databases in SD High Level Design or HLD Low Level Design or LLD Top SD Interview Questions Interview Corner Company Wise Preparation Preparation for SDE Experienced Interviews Internship Interviews Competitive Programming Aptitude Preparation GfG School CBSE Notes for Class 8 CBSE Notes for Class 9 CBSE Notes for Class 10 CBSE Notes for Class 11 CBSE Notes for Class 12 English Grammar Commerce Accountancy Business Studies Economics Management Income Tax Finance UPSC Polity Notes Geography Notes History Notes Science and Technology Notes Economics Notes Important Topics in Ethics UPSC Previous Year Papers SSC/ BANKING SSC CGL Syllabus SBI PO Syllabus SBI Clerk Syllabus IBPS PO Syllabus IBPS Clerk Syllabus Aptitude Questions SSC CGL Practice Papers Write & Earn Write an Article Improve an Article Pick Topics to Write Write Interview Experience Internships Video Internship @geeksforgeeks , Some rights reserved We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our Cookie Policy & Privacy Policy Got It !