Changing Host IP Address in Flask

Last Updated : 05 Feb, 2023

In this article, we will cover how we can change the host IP address in Flask using Python. The host IP address is the network address that identifies a device on a network. In the context of a Flask application, the host IP address is the address that the application listens to for incoming requests. By default, Flask applications listen on the localhost address **127.0.0.1:5000**, which means they can only be accessed from the same machine that the application is running on.

However, it is often useful to be able to access the application from other devices on the same network, or even from the internet. In these cases, the host IP address can be changed to allow the application to be accessed from other devices. This is done by specifying the host IP address in the app.run() function of the Flask application.

Changing the IP address in a Flask application using the "host" parameter

Here are the steps for changing the host IP address in a Flask application using the "host" parameter in the app.run() function. Open the Flask application in a text editor. Locate app.run() function in the main script file. Add the "host" parameter to the app.run() function, followed by the desired IP address, such as: app.run(host='192.168.0.105') this is the local address of your system. This will only allow the application to be accessed from the specified IP address.

Python3

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
```
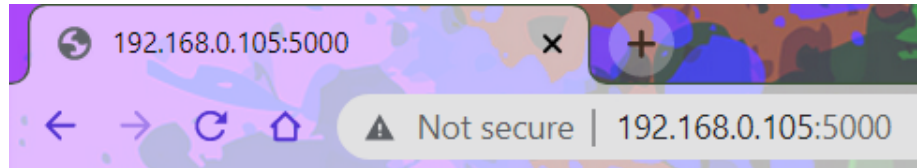
```python
def hello():
    return 'Hello, World! this application runing on 192.168.0.105'

if __name__ == '__main__':
    app.run(host='192.168.0.105')
```

Output:

Changing IP from the command line while deploying the Flask app

Here, the app.run() function does not specify an IP address or a port, so it will use the defaults of localhost (127.0.0.1) and port 5000. You can run this application by setting the FLASK_APP environment variable to the name of your application file and then using the flask run command and then you can change the IP address and port number while running the command

```
set FLASK_APP=app.py
flask run
```

```
flask run --host=192.168.0.105 --port=5000
```

Python3

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def hello():
    return 'Hello, World!'

if __name__ == '__main__':
    app.run()
```
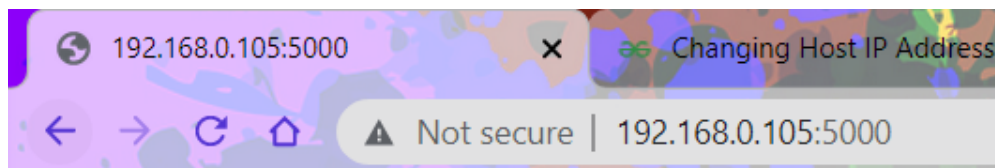
Output:

This will run the server on IP address 192.168.0.105 and port 5000.

```
C:\Users\Lenovo\Desktop\Flask>set FLASK_APP=app.py

C:\Users\Lenovo\Desktop\Flask>flask run --host=192.168.0.105 --port=5000
 * Serving Flask app 'app.py'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://192.168.0.105:5000
Press CTRL+C to quit
```

192.168.0.105:5000   ×   Changing Host IP Address

← → C ⌂   ⚠ Not secure | 192.168.0.105:5000

Hello, World!

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!

deve…   4

Previous Article   **Next Article**

How to Change Port in Flask app   Flask Rendering Templates

Similar Reads

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

Build a GUI Application to ping the host using Python

Prerequisite: Python GUI – Tkinter In this article, we are going to see how to ping the host with a URL or IP using the python ping module in Python. This module...

2 min read

10 Best GitHub Alternatives to Host Python Projects in 2024

Believe it or not but when it comes to hosting any project in anyone's mind whether it's a student or a working professional, GitHub strikes the mind for...

7 min read

Python Flask - ImmutableMultiDict

MultiDict is a sub-class of Dictionary that can contain multiple values for the same key, unlike normal Dictionaries. It is used because some form elements ha...

2 min read

Handling 404 Error in Flask

Prerequisite: Creating simple application in Flask A 404 Error is showed whenever a page is not found. Maybe the owner changed its URL and forgot to...

4 min read

Python | Using for loop in Flask

Prerequisite: HTML Basics, Python Basics, Flask It is not possible to write front-end course every time user make changes in his/her profile. We use a template...

3 min read

Article Tags :   Python   Technical Scripter   Python Flask   Technical Scripter 2022

Practice Tags :   python

Company   **Languages**

About Us

Python

Legal

Java

In Media

C++

Contact Us

PHP

Advertise with us

GoLang

GFG Corporate Solution

SQL

Placement Training Program

R Language

GeeksforGeeks Community

Android Tutorial

Tutorials Archive

DSA

Data Science & ML

Data Structures

Data Science With Python

Algorithms

Data Science For Beginner

DSA for Beginners

Machine Learning

Basic DSA Problems

ML Maths

DSA Roadmap

Data Visualisation

Top 100 DSA Interview Problems

Pandas

DSA Roadmap by Sandeep Jain

NumPy

All Cheat Sheets

NLP

Deep Learning

Web Technologies

Python Tutorial

HTML

Python Programming Examples

CSS

Python Projects

JavaScript

Python Tkinter

TypeScript

Web Scraping

ReactJS

OpenCV Tutorial

NextJS

Python Interview Question

Bootstrap

Django

Web Design

Computer Science

DevOps

Operating Systems

Git

Computer Network

Linux

Database Management System

AWS

Software Engineering

Docker

Digital Logic Design

Kubernetes

Engineering Maths

Azure

Software Development

GCP

Software Testing

DevOps Roadmap

System Design

Inteview Preparation

High Level Design

Competitive Programming

Low Level Design

Top DS or Algo for CP

UML Diagrams

Company-Wise Recruitment Process

Interview Guide

Company-Wise Preparation

Design Patterns

Aptitude Preparation

OOAD

Puzzles

System Design Bootcamp

Interview Questions

School Subjects

Mathematics

Physics

Chemistry

Biology

Social Science

English Grammar

Commerce

World GK

GeeksforGeeks Videos

DSA

Python

Java

C++

Web Development

Data Science

CS Subjects