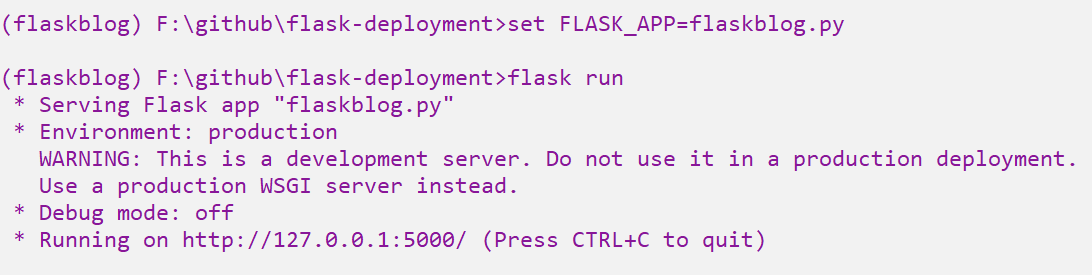
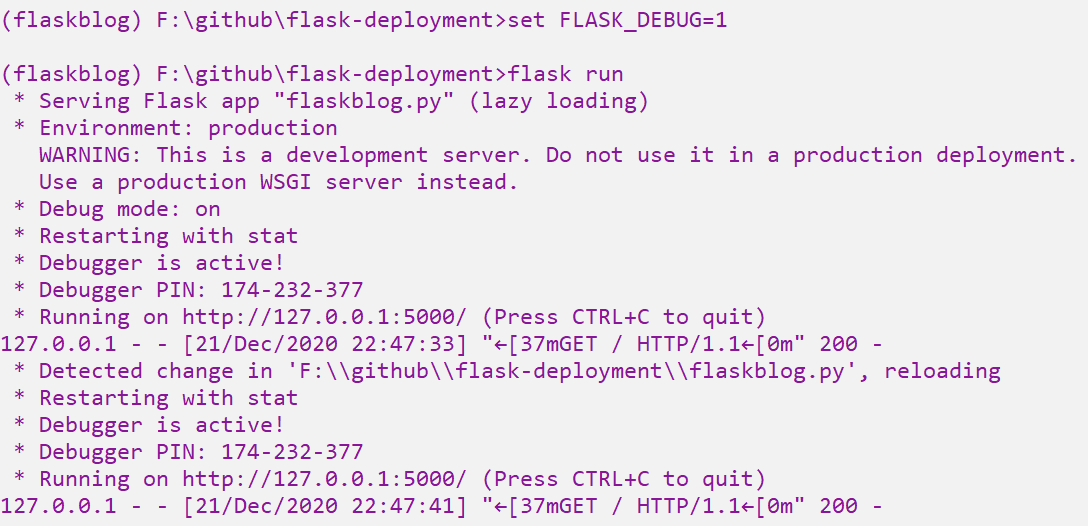
**\_\_name\_\_ 🡪** is the name if we run the script directly from python. But if we import this module to somewhere else then this name will be the name of the module.

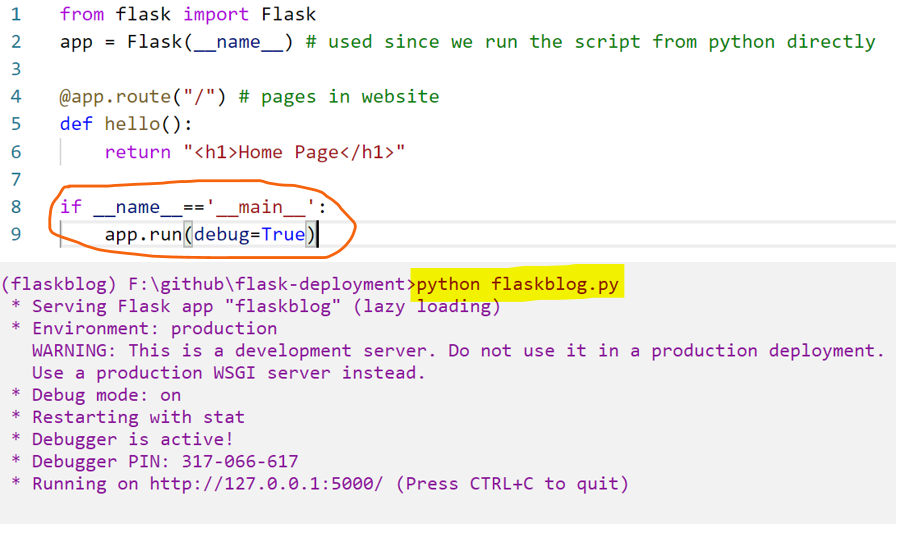
**1)Running the browser by creating environment variable**

  
For each and every smaller changes, its hard to generate URL’s again and again. This can be avoided by running the application in the debug mode.



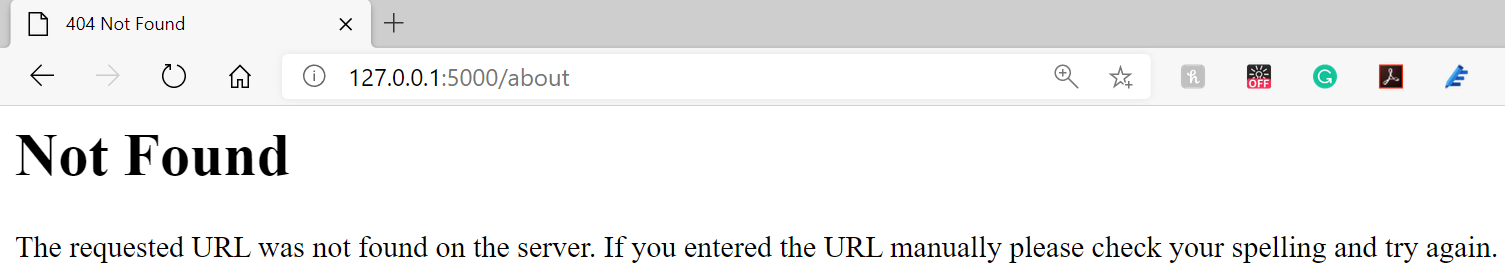
**2) Debug Mode**Now I don’t want to start the server again and again, by refreshing the browser, the contents will be loaded.

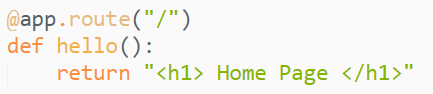


**3)Running directly from python**

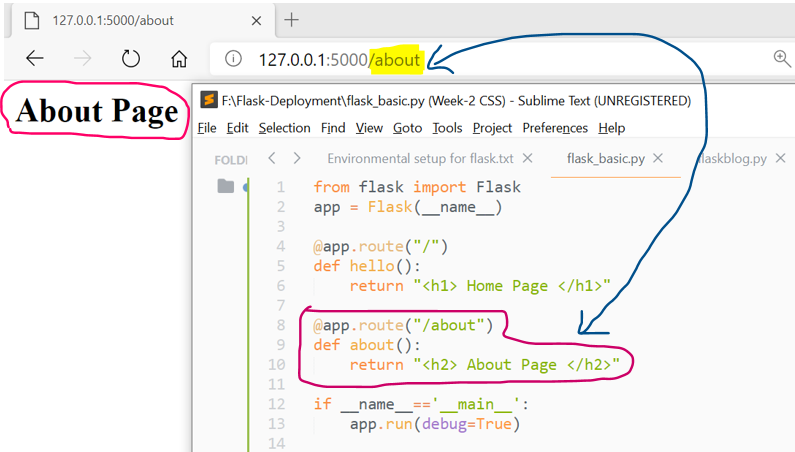
**Go to about page.**



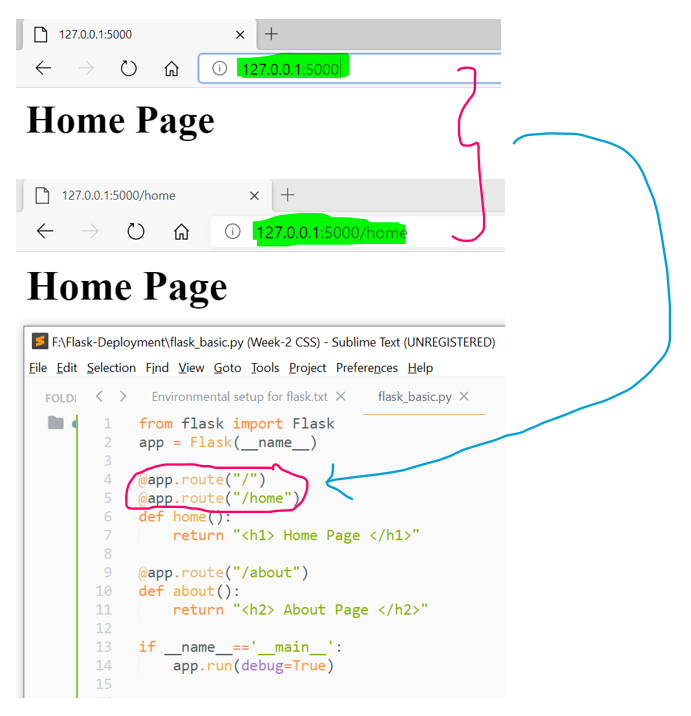
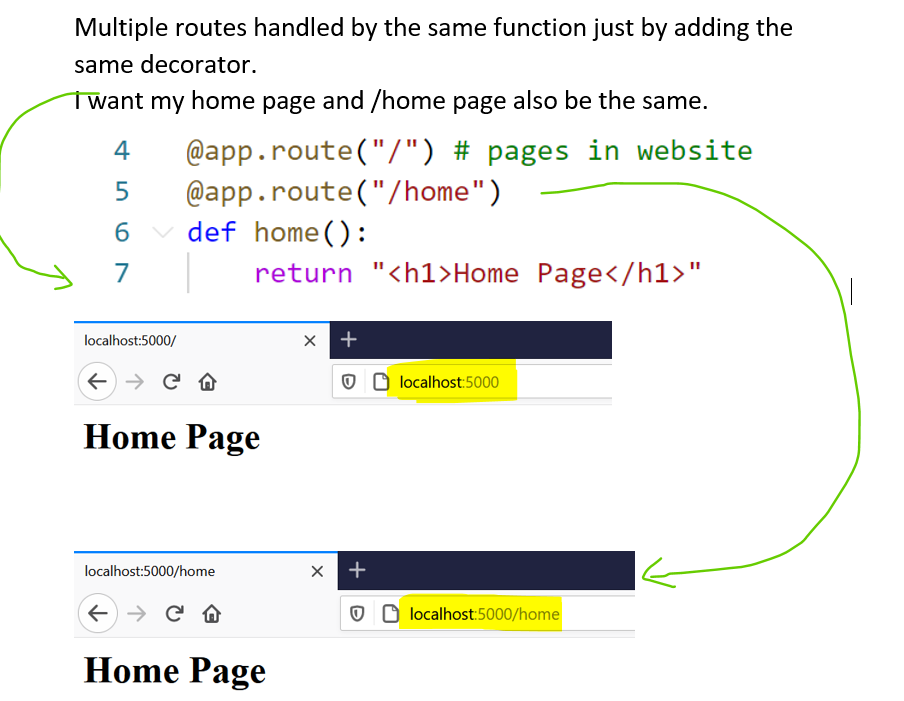


Actually the about doesn’t exist, only the home page exists.  


**Now Try to make a about page for the website**

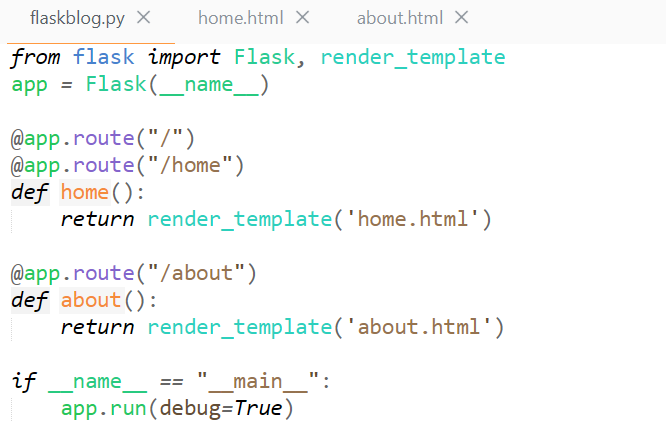


**Having 2 routes with the same function**

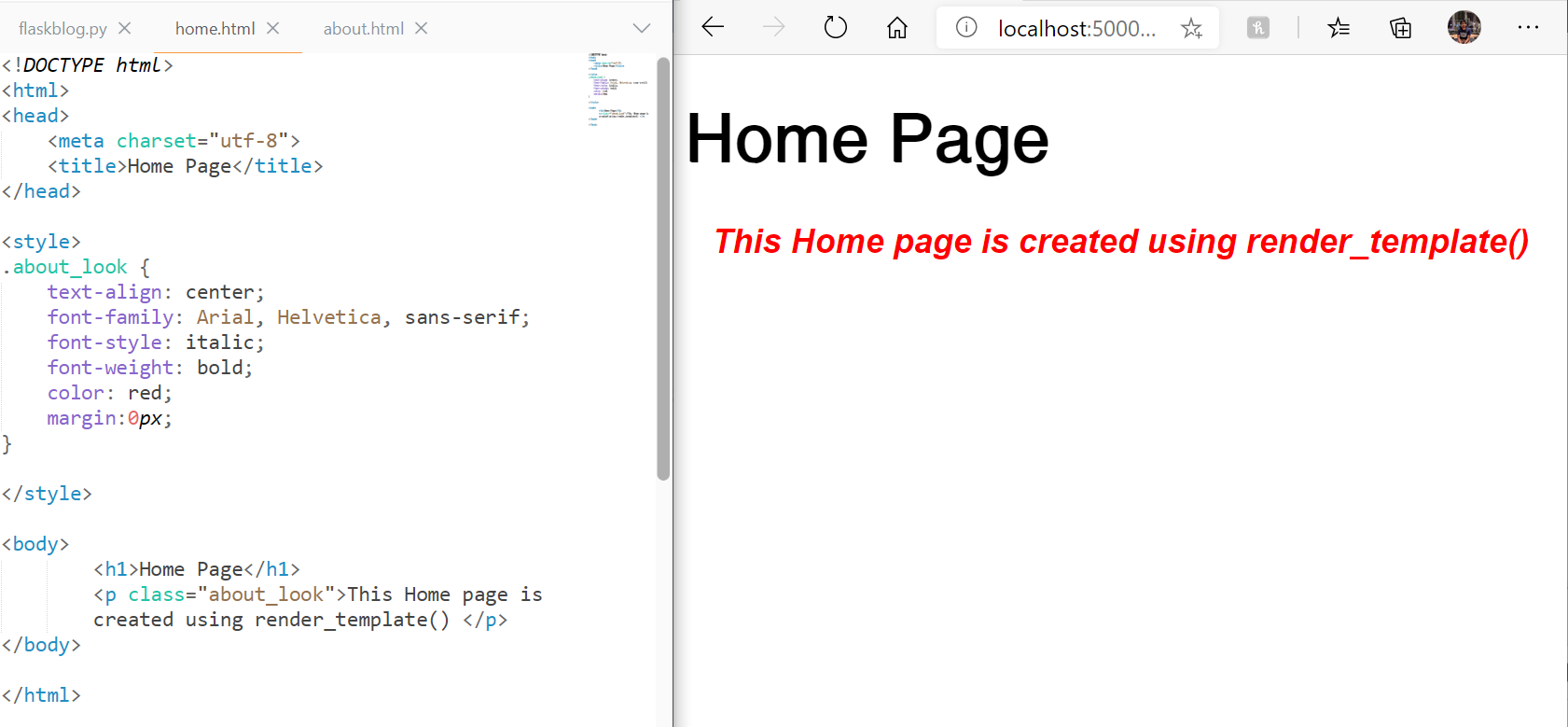
  


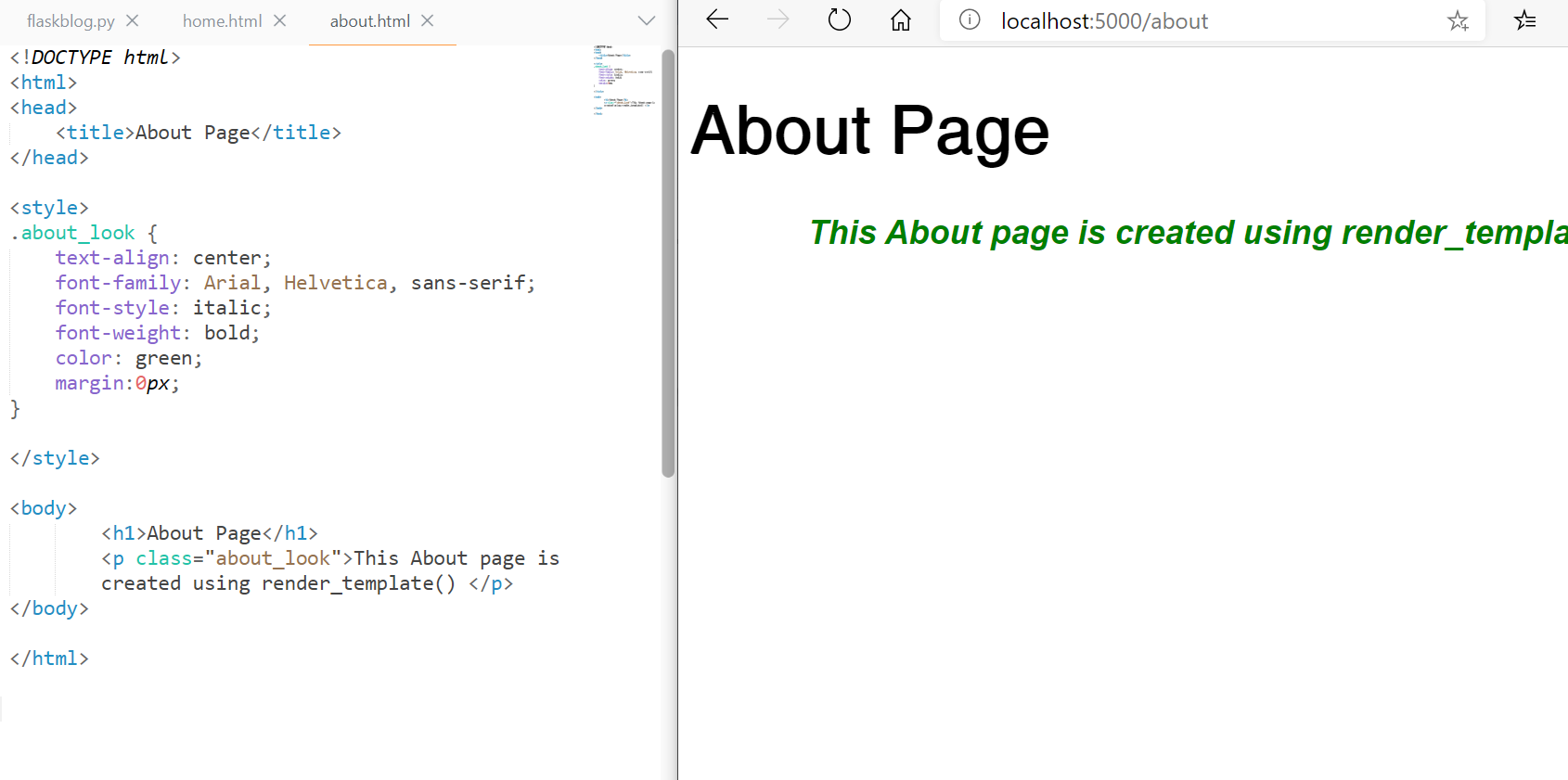
**Templates**

**render template**  
**render\_template** is used to generate output from a **template** file based on the Jinja2 engine that is found in the application's templates folder. Note that **render\_template** is typically imported directly from the **flask** package instead of from **flask**. templating .

html file location 🡪 templates/html file

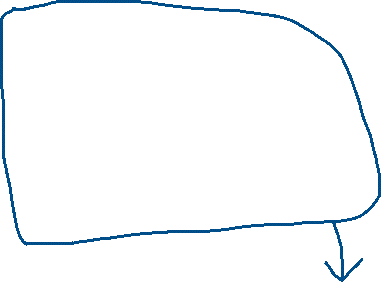
Storing both home.html and about.html file in the same project under template folder.





**Posts**



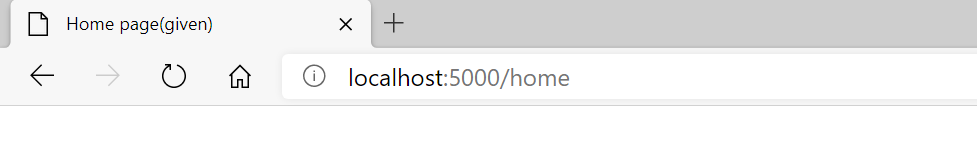


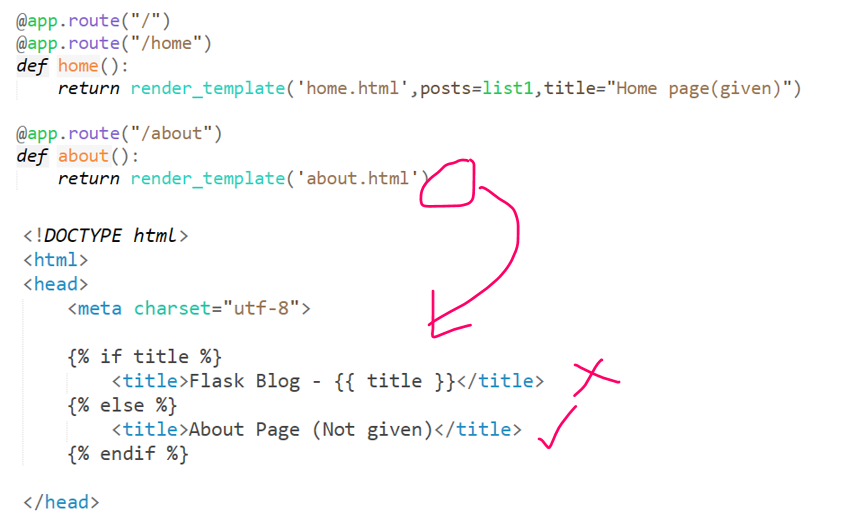
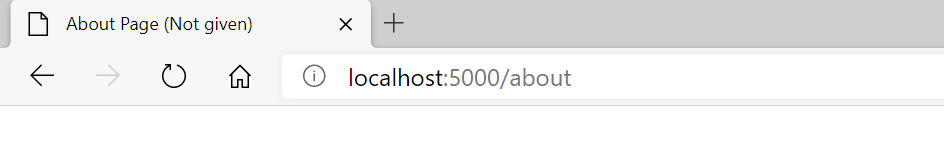




**title in render\_template()**





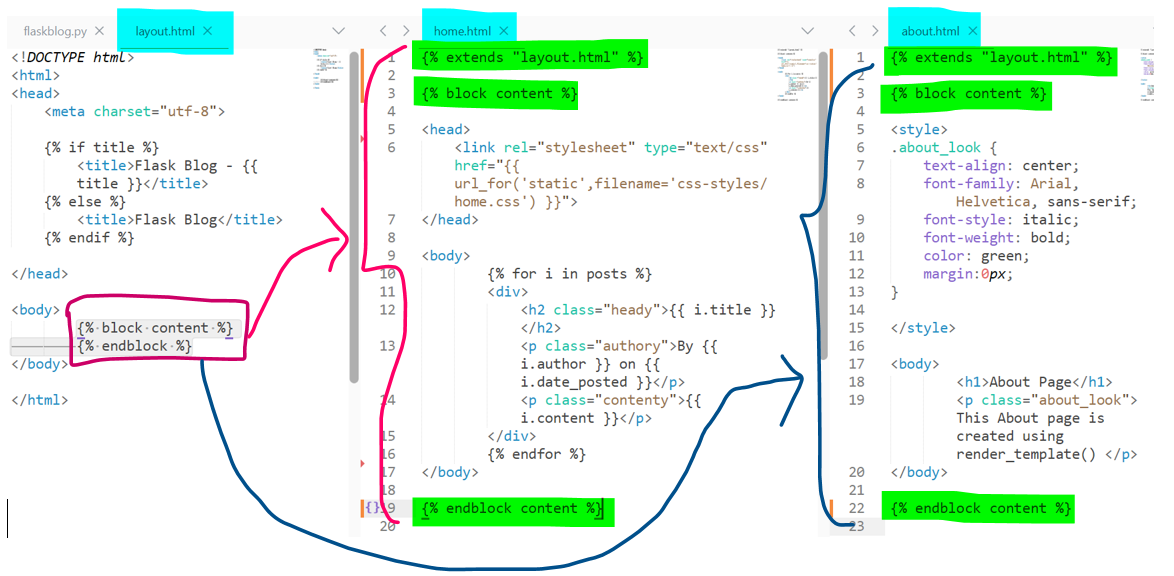
**Templates using Inheritance**

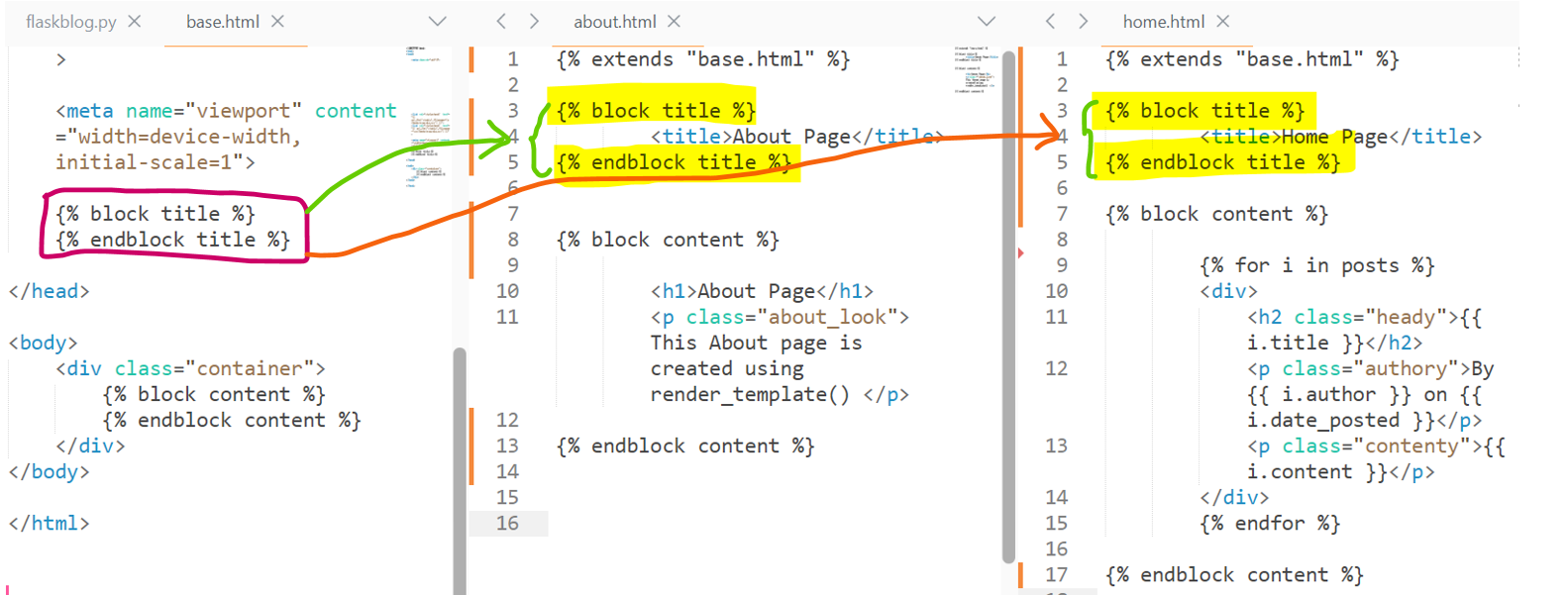
I am going to use layout.html, which contains the common characteristics of both home.html and about.html. Store the layout.html file in the templates folder.

layout.html file

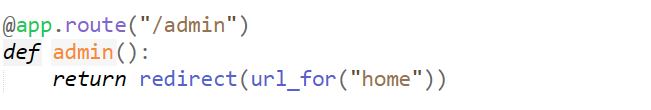
title part to common to both home.html and about.html, So I included it in the layout.html part.

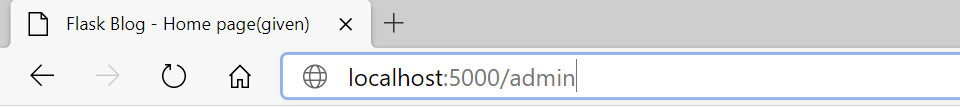
**How layout.html is it connected to home.html and about.html?**



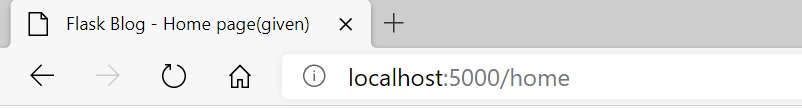


**redirect()**

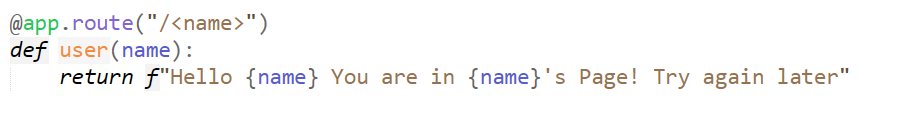
This will re-direct you to a specific page   




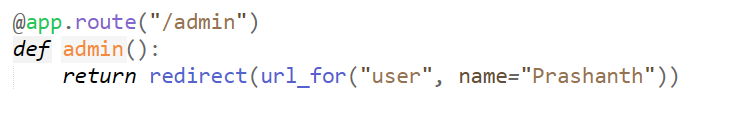
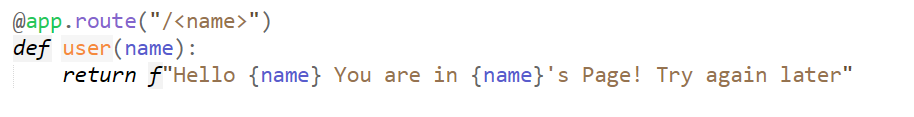
Once I give the route as admin, it will immediately re-direct me to the “home” page.



**/<name>**

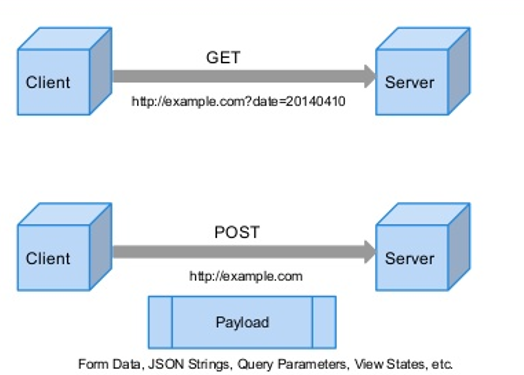


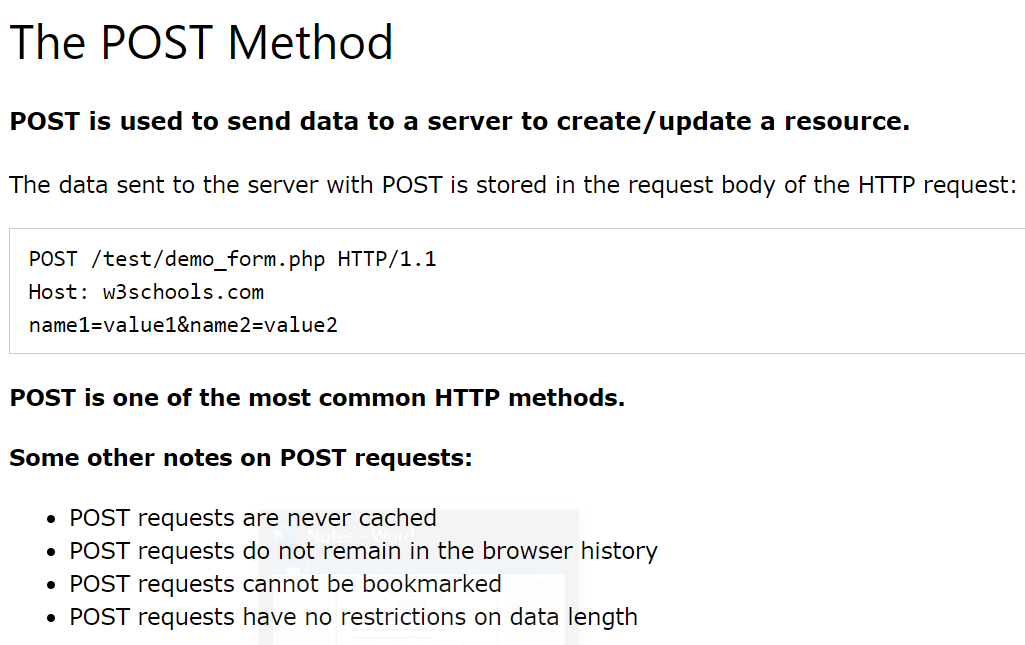
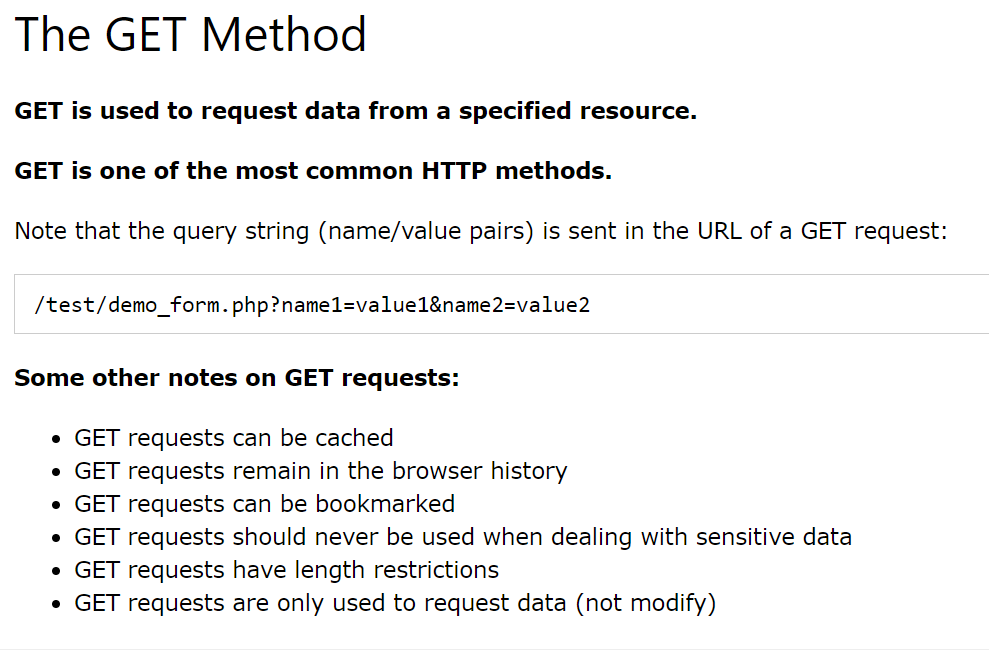
**(or)**



**Blocks**

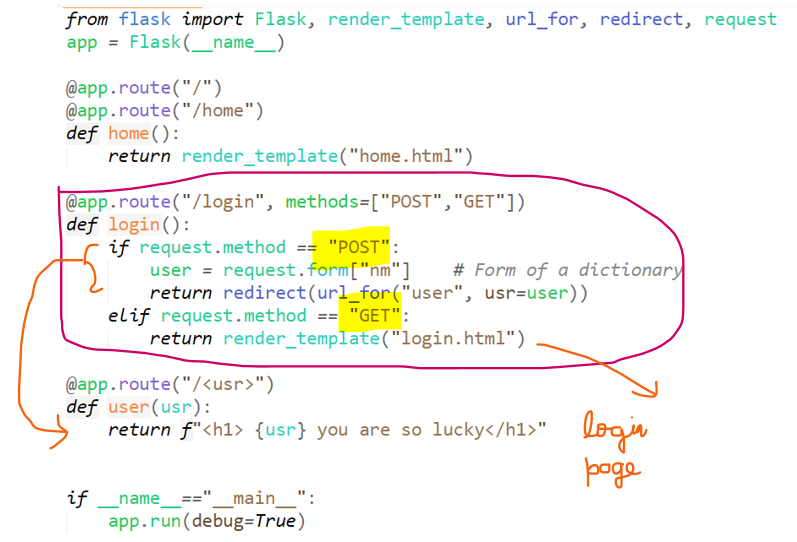
**GET and POST**

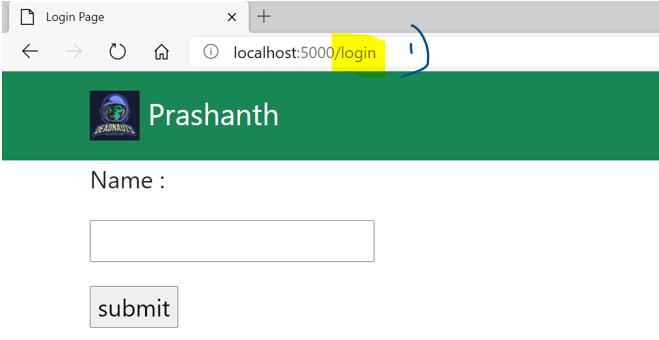


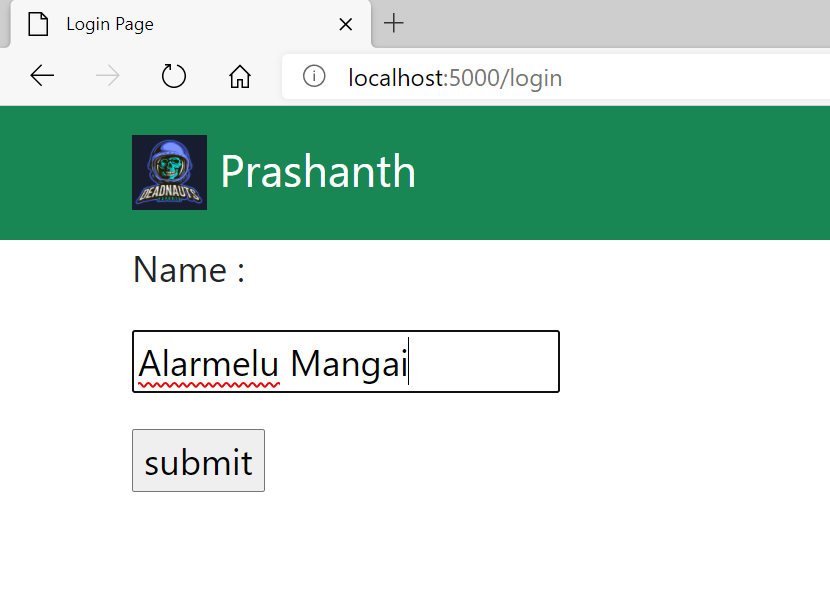
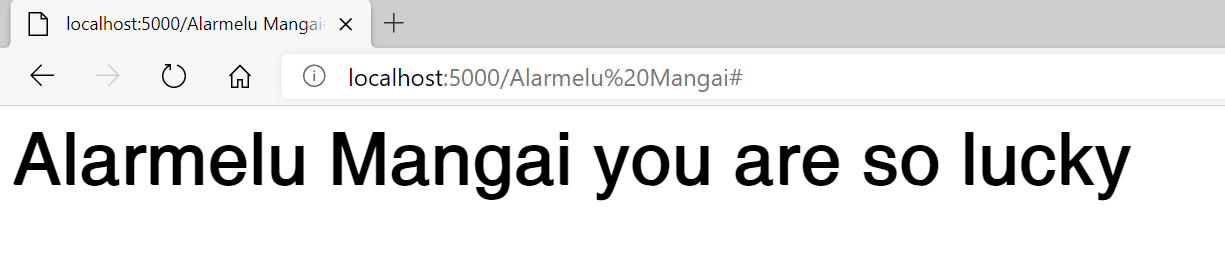


get 🡪 data sent through the URL [ visible in the url ]   
post 🡪 data sent through the BODY (user id and password)   
[ not visible in the url ]

application file

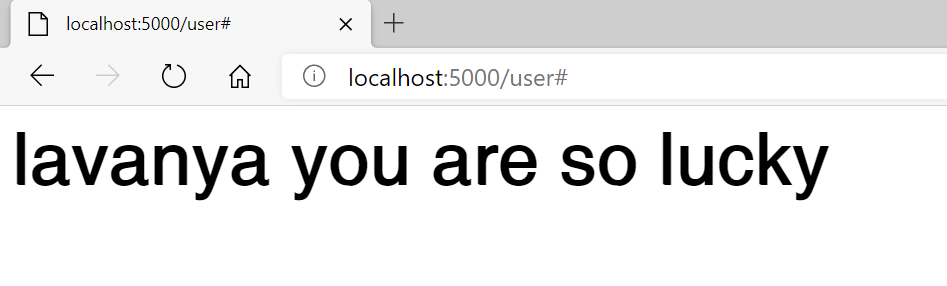
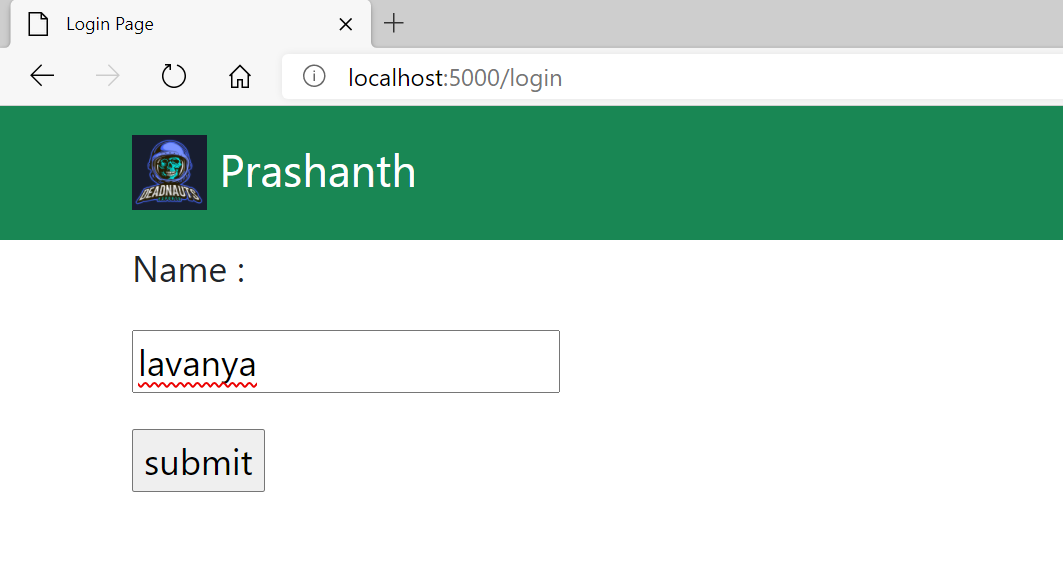


1)First give “login” in the url (i.e get method)  


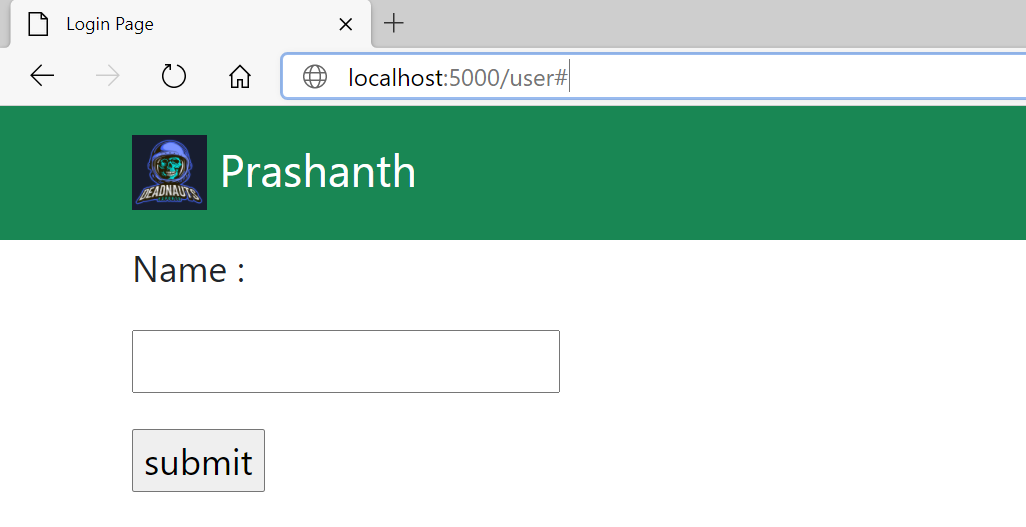
2)Then give the name and press submit (i.e post method)  
  
3)And it is switched to “(/<usr>)”  




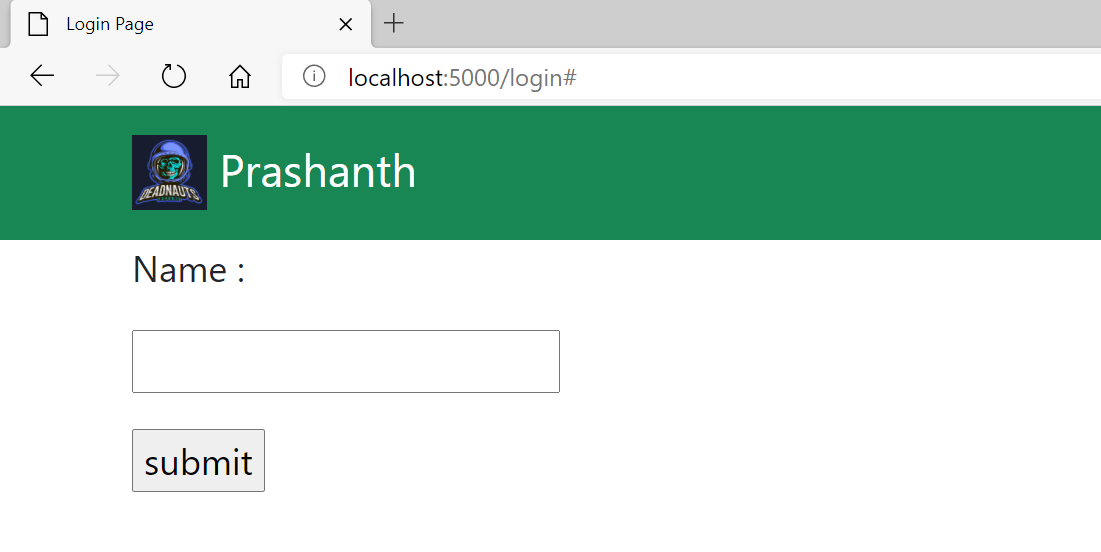
**Sessions**



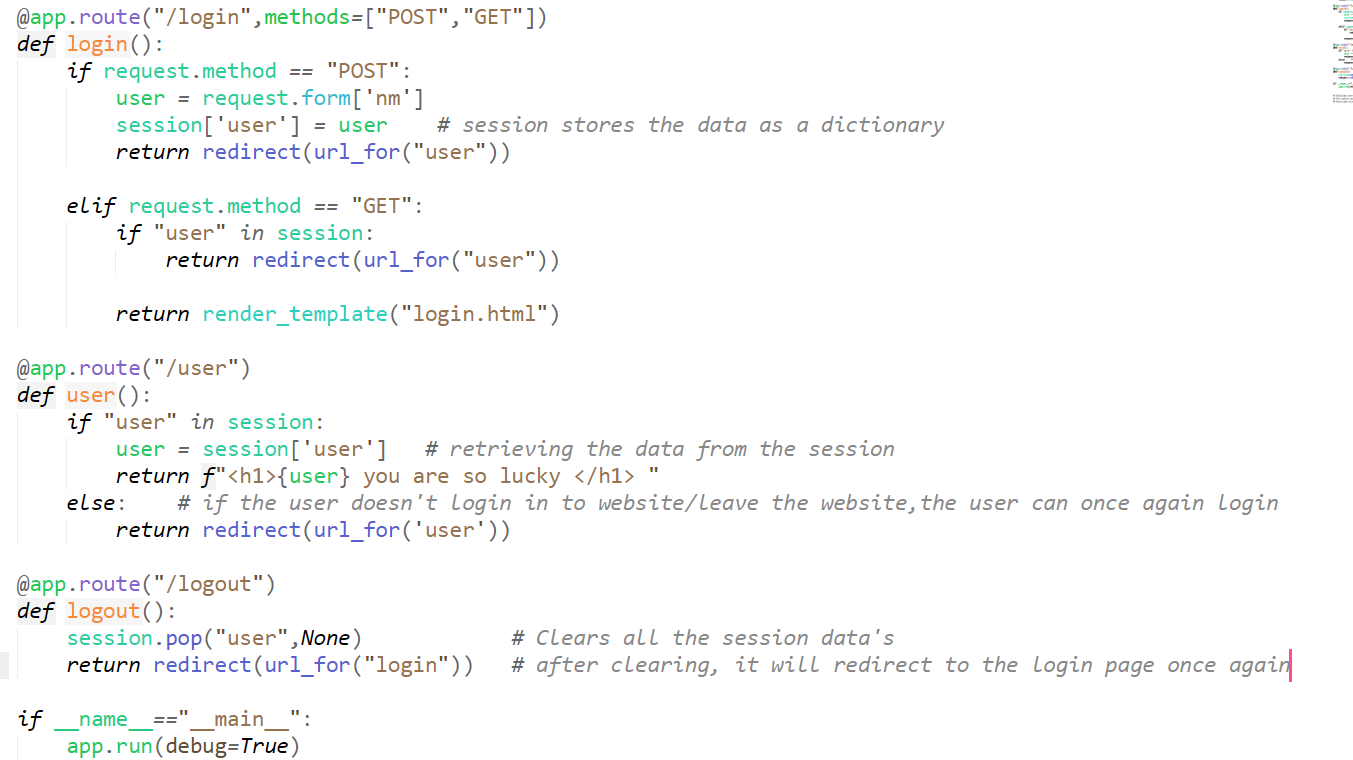
Now close the browser and type **user#** once again and check



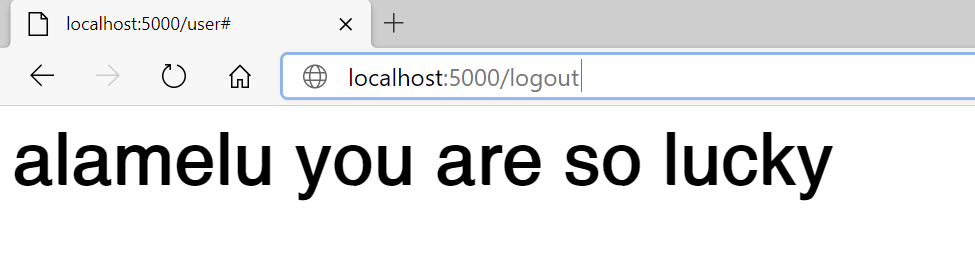
Type **user#** in the url and click enter, then automatically the website is directed to login page.



This is because when we close the browser and reopen it all the sessions data are erased, so we can’t able to retrieve back the user name who are previously logged in







Click enter, then the session clears all the data and redirects to the login page for another user.

flask file directory

