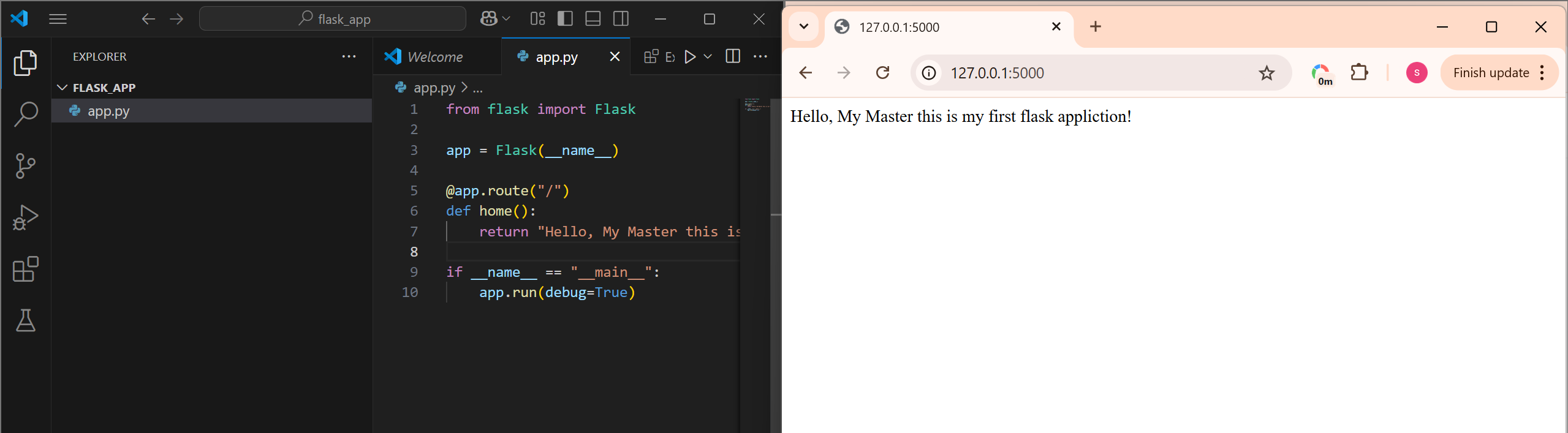
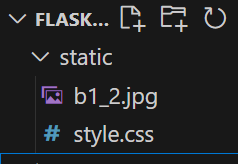
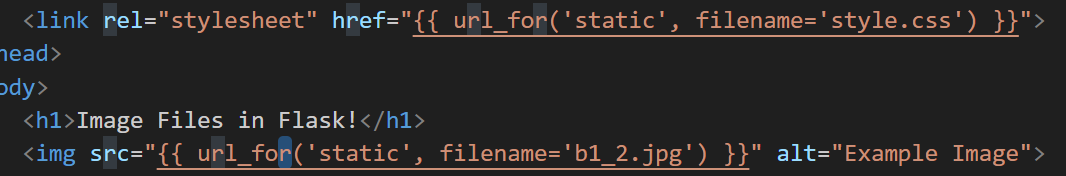
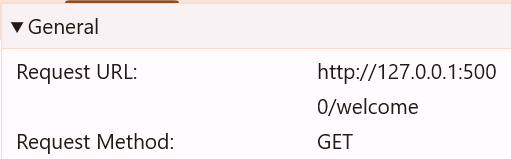
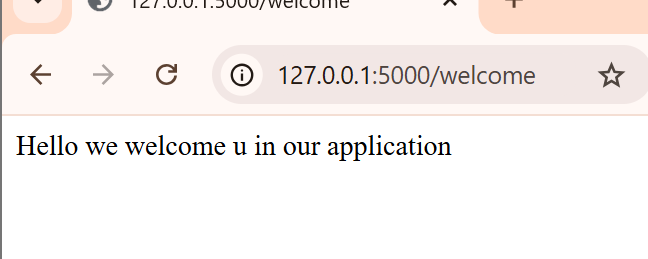
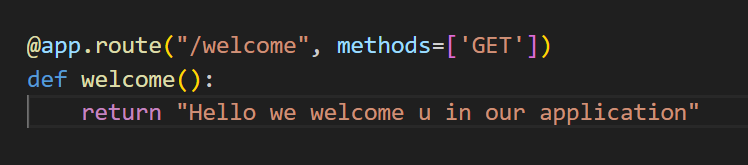
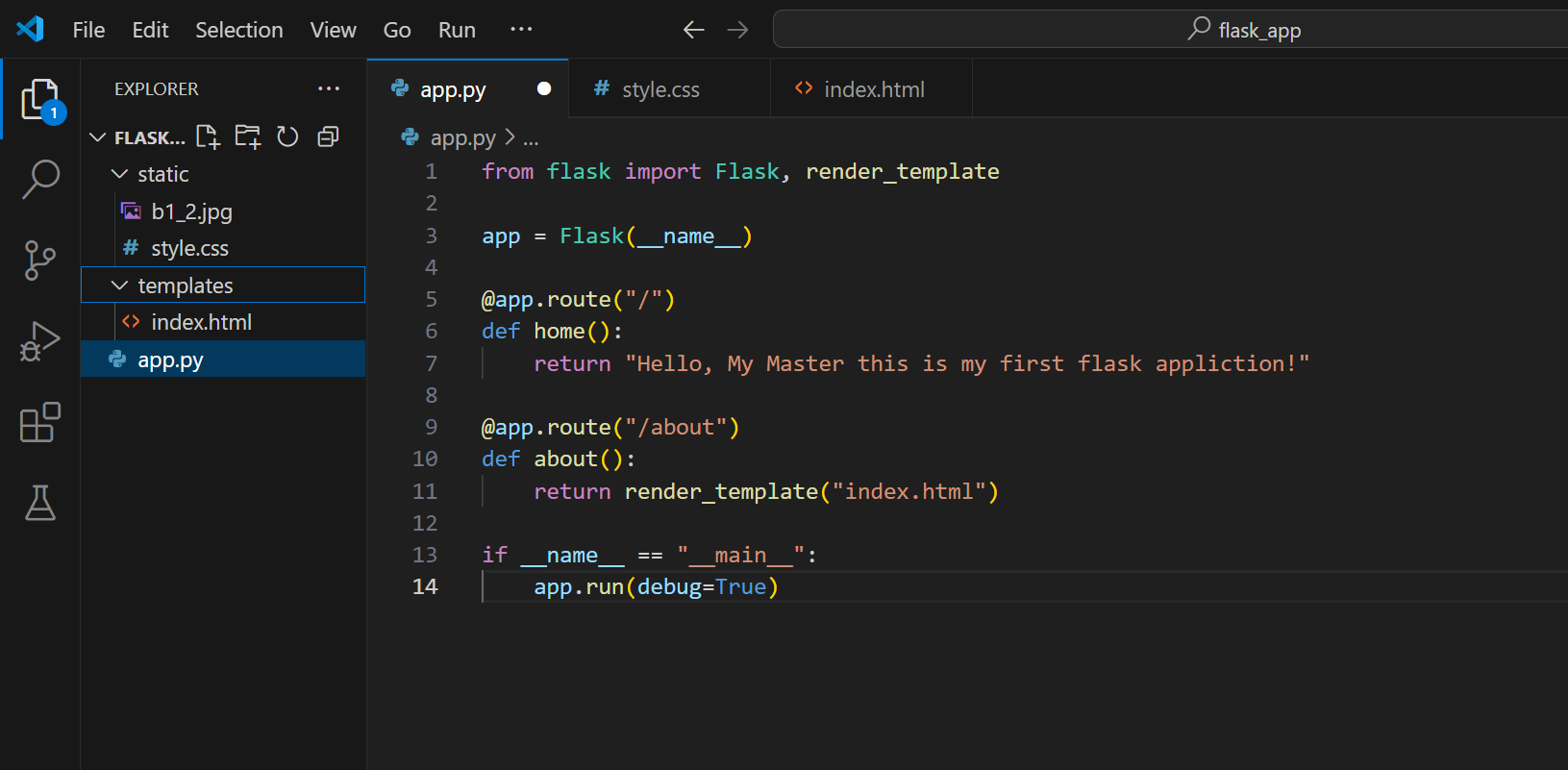
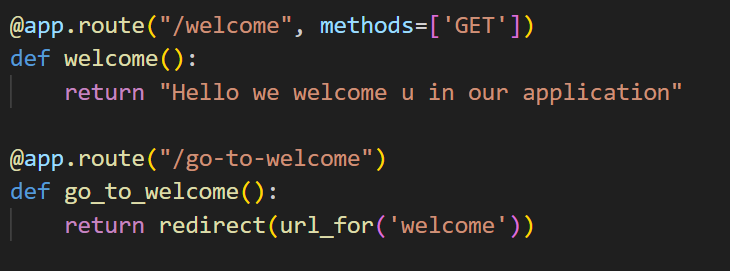
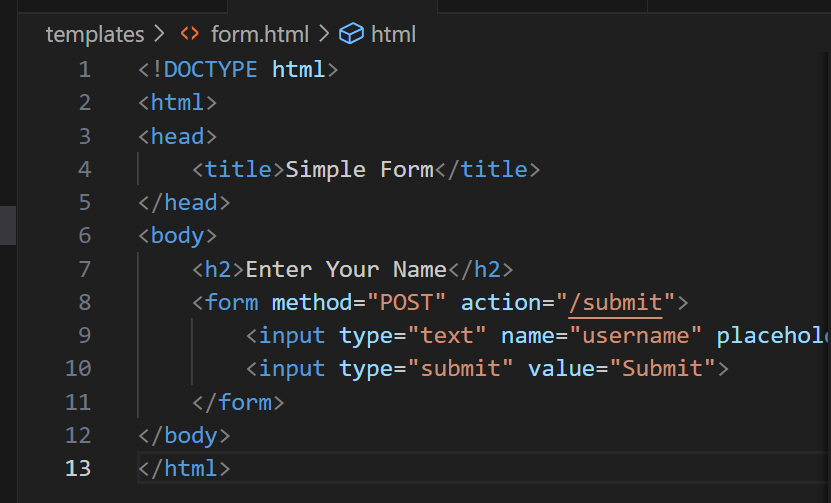
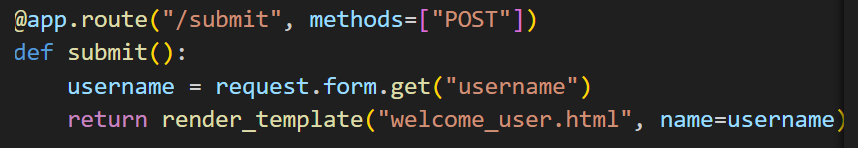
Question 1 : How do you create a basic Flask application?   
First Step: we install the flask using this command :  
  
   
Second Step:   
create “flask\_app” folder in your system and create a “app.py” file there  
write the code in app.py   
then run this command in cmd : python app.py  
   
  
  
  
  
  
Question 2 : How do you serve static files like images or CSS in Flask?   
  
First Step : create static folder in flask\_app folder and add Images or CSS files here  
  
  
second step: call those file in your application  
  
  
  
Question 3: How do you define different routes with different HTTP methods in Flask?  
In Flask, we can define routes that accept specific HTTP methods (like GET, POST, PUT, DELETE, PATCH, etc.) using the methods parameter in the @app.route() decorator.  
  
  
  
  
Question 4 : How do you render HTML templates in Flask?

For rendering the templates in flask we use “render\_template()” function.  
Flask looks for templates inside a folder called templates by default.

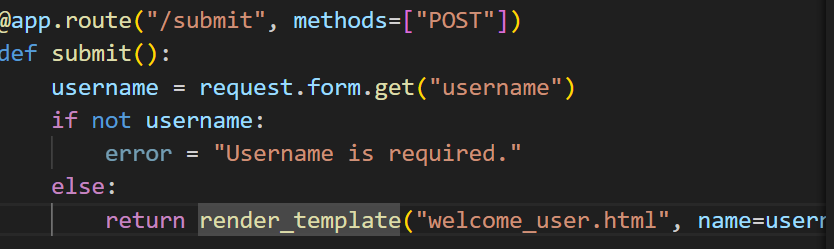
First step: create templates folder inside the flask\_app then we create a HTML file there.  
second step: we return the HTML file using render\_template() function.   
   
  
  
  
  
Question 5: How can you generate URLs for routes in Flask using url\_for?  
we use the url\_for() function to dynamically generate URLs for routes based on their function names, instead of hardcoding them.   
This makes your app more flexible and easier to maintain.  


Question 6: How do you handle forms in Flask?  
we can handle the forms in flask using request.form

First we create a HTML form in templates using accept post request.  
  
  
Next we accessing form data with request.form  


Question 7: How can you validate form data in Flask?

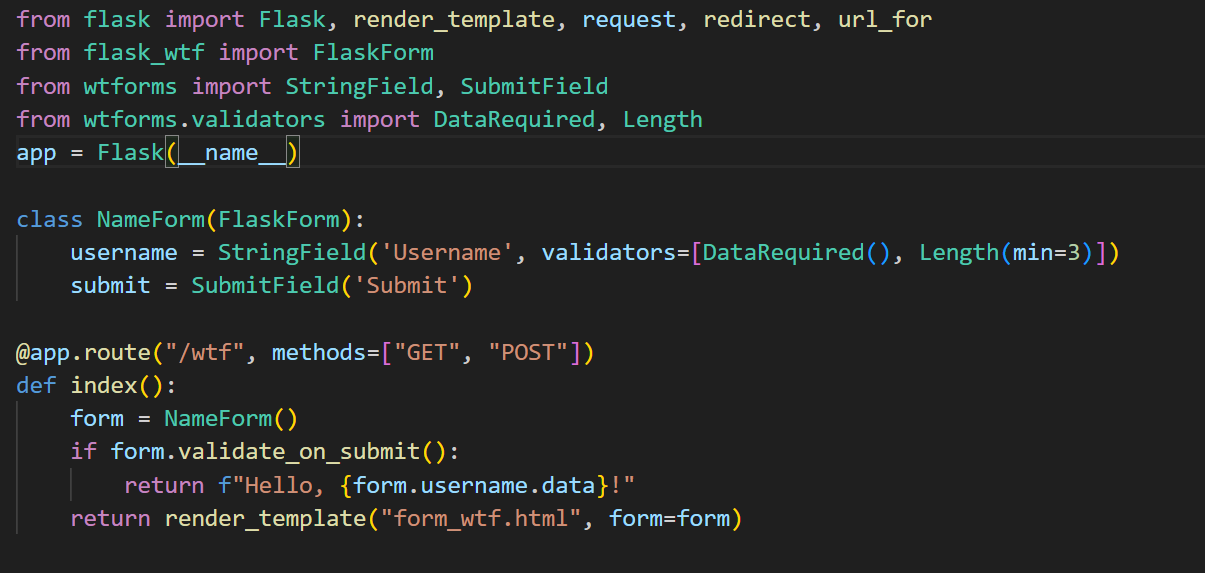
In Flask, we can validate form data in two main ways:

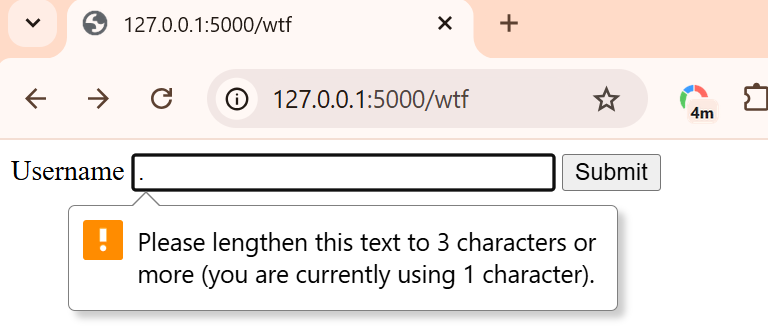
First way: we can check the form data manually using if else condition  


Second way: using Flask-WTF (Web Template Forms)

For this we have to install flask-wtf to run this command : pip install flask-wtf

And import these modules

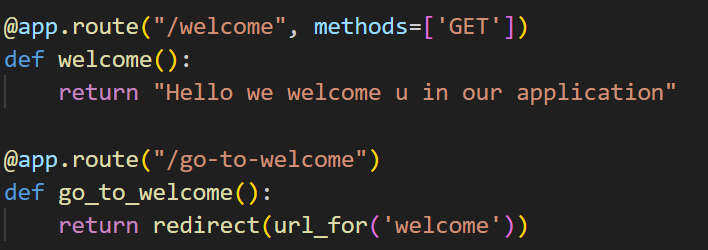




Question 8:

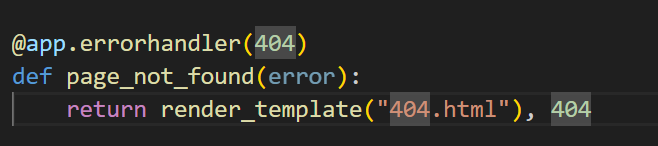
Question 9 : How do you redirect to a different route in Flask?

we redirect to a different route using the redirect() function along with url\_for() to generate the target URL.



Question 10: How do you handle errors in Flask (e.g., 404)?

we can handle errors like 404 Not Found, using **error handlers** with the @app.errorhandler() decorator.

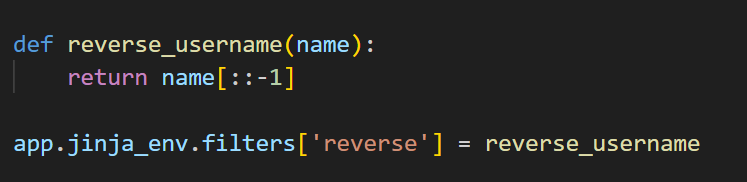


Question 11: How do you structure a Flask app using Blueprints?

**Blueprints** in Flask helps you organize large applications into smaller, reusable modules. It keeps your project **clean, modular, and scalable**.

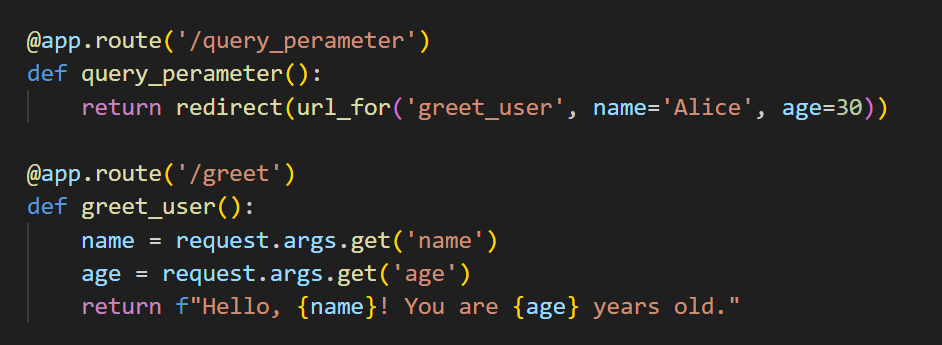
Question 12: How do you define a custom Jinja filter in Flask?

First we write a function then we registering it with flask’s Jinja environment. Then we use it where we want.



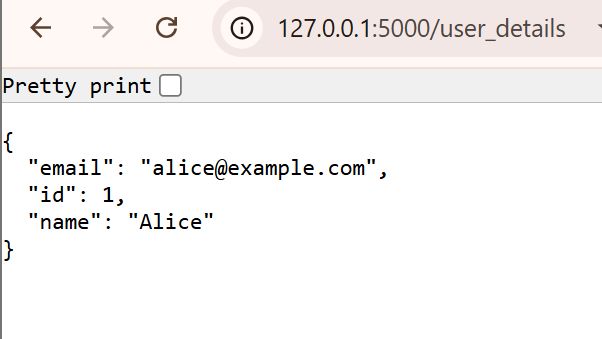
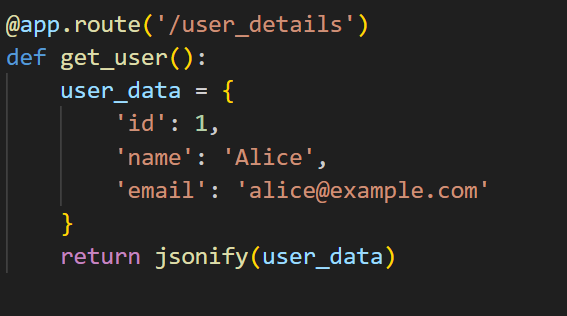
Question 13: How can you redirect with query parameters in Flask?

we can redirect with **query parameters** using redirect() and url\_for(), and by passing additional arguments as keyword arguments.



Question 14: How do you return JSON responses in Flask?

we can return **JSON responses** using the jsonify() function, which automatically formats your data as JSON and sets the correct Content-Type header (application/json).



Question 15: How do you capture URL parameters in Flask?

We can capture the URL parameters in flask using angle brackets(< >) outside of placing variables in route path.

