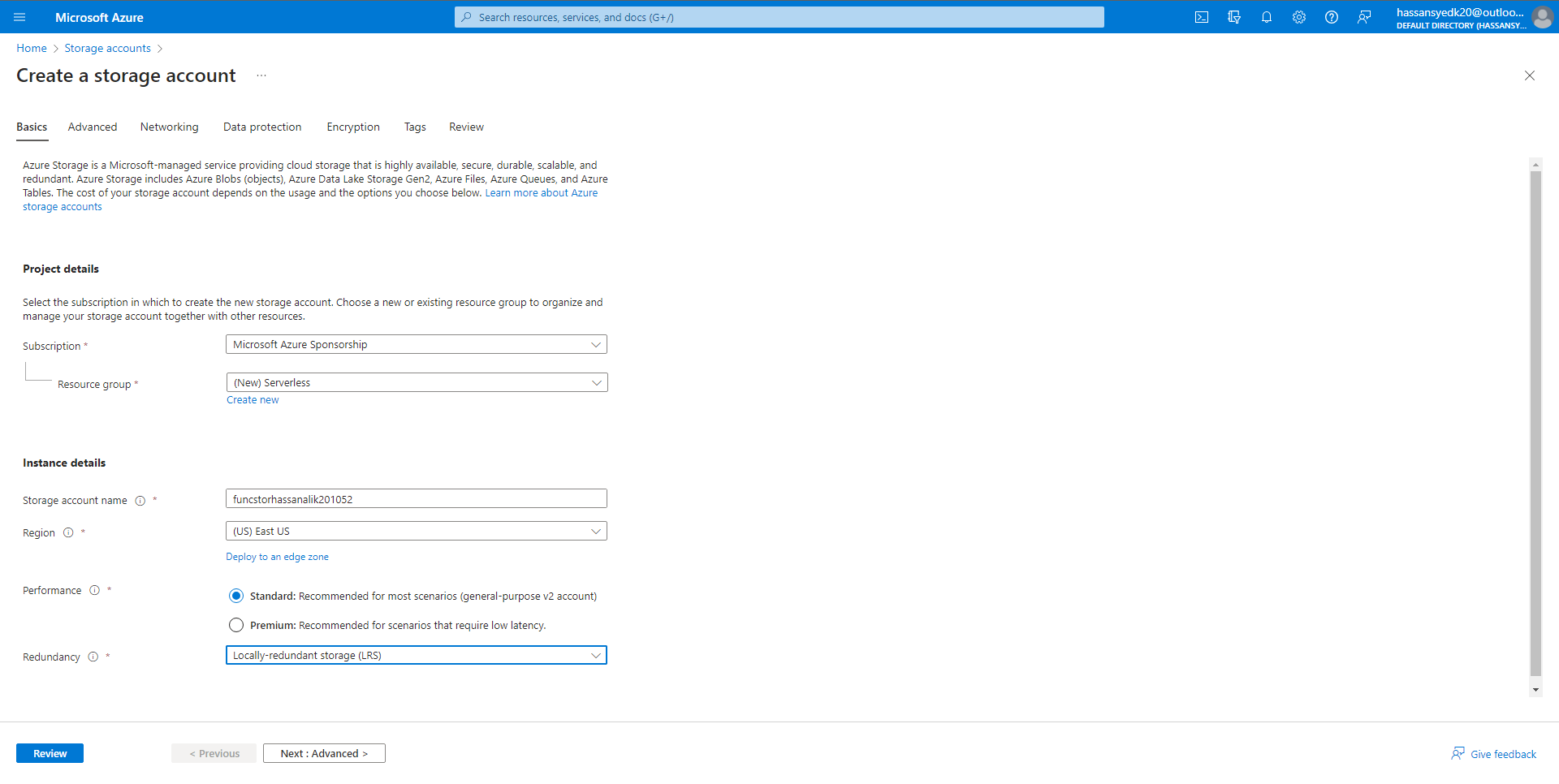
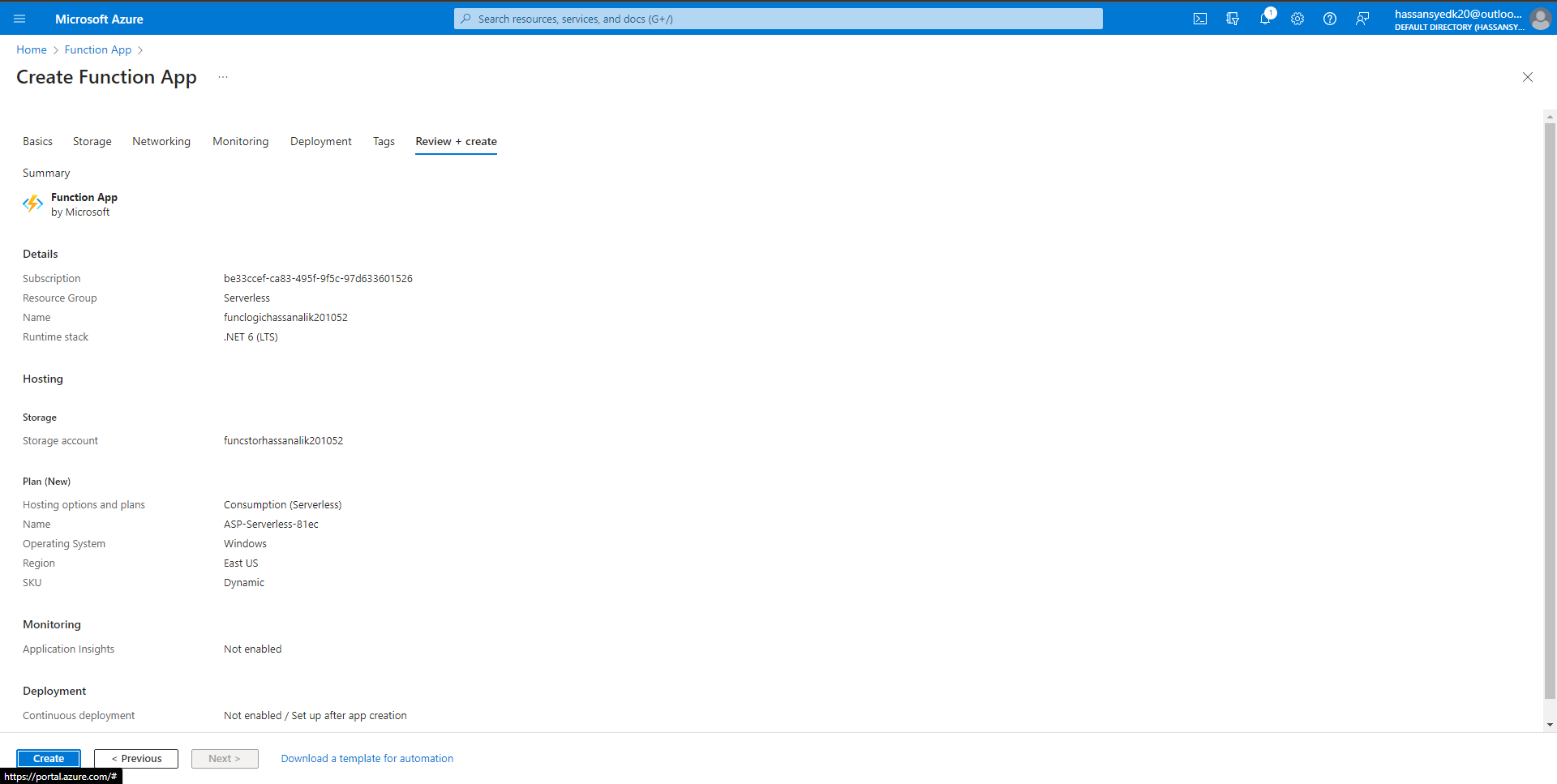
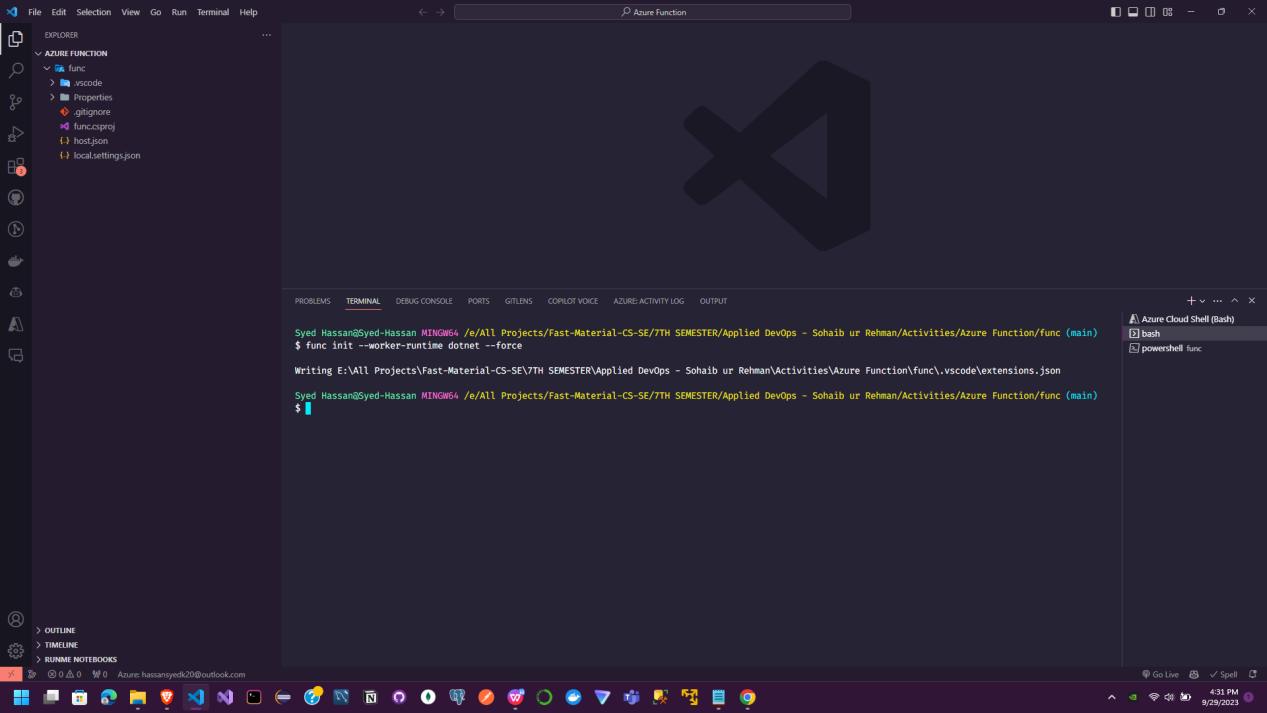
**Hassan Ali - K201052 - BSE-7B**

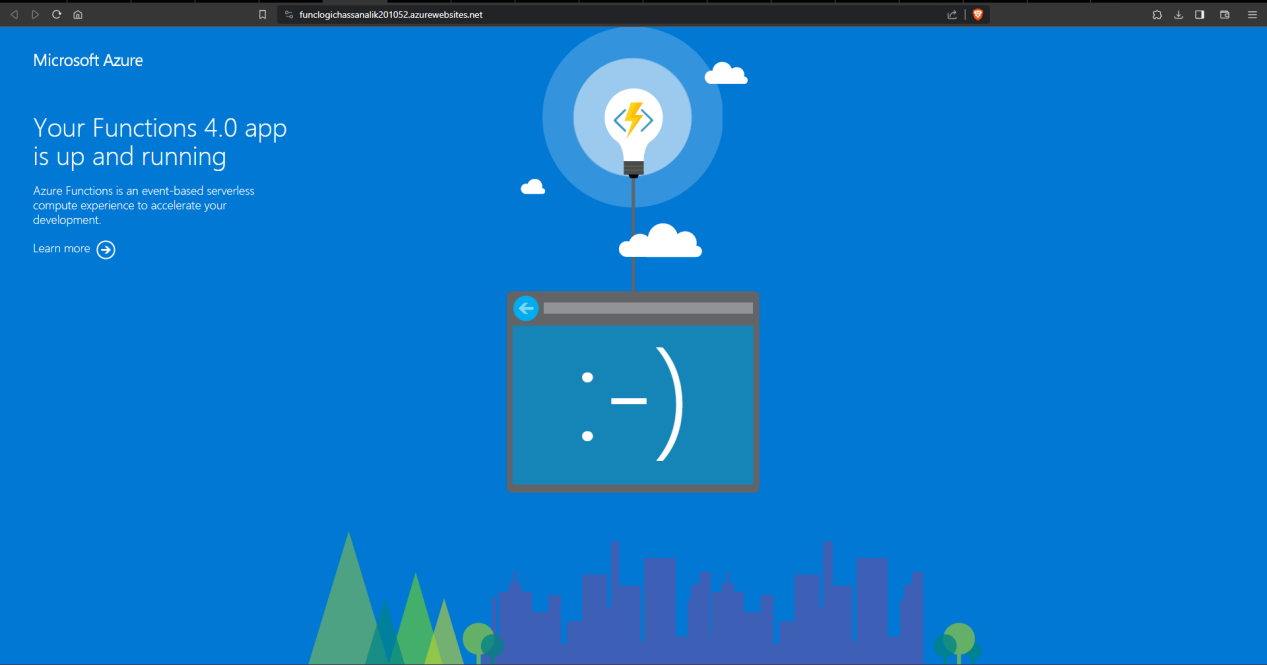
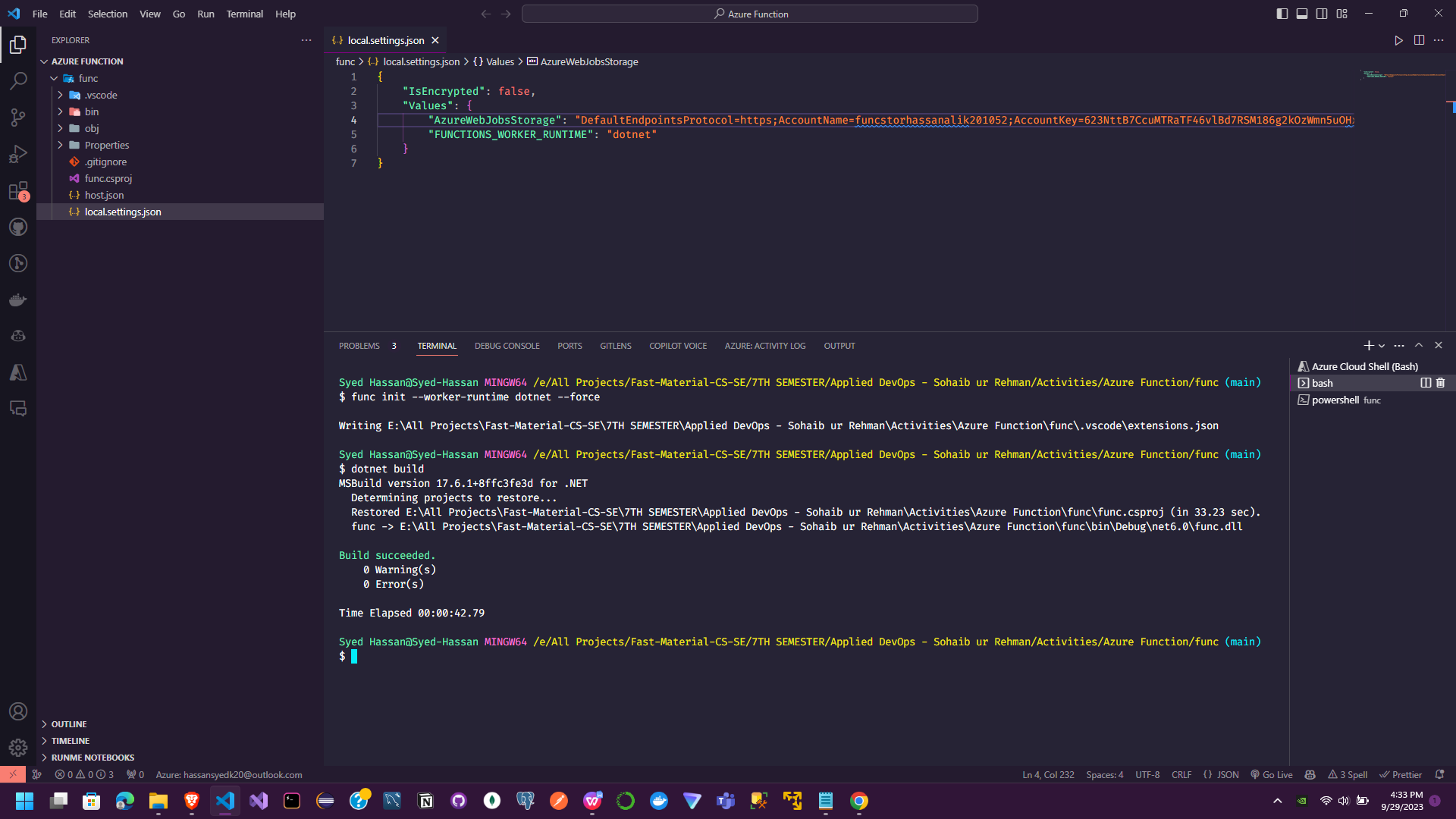
**Lab 02: Implement task processing logic by using Azure Functions**

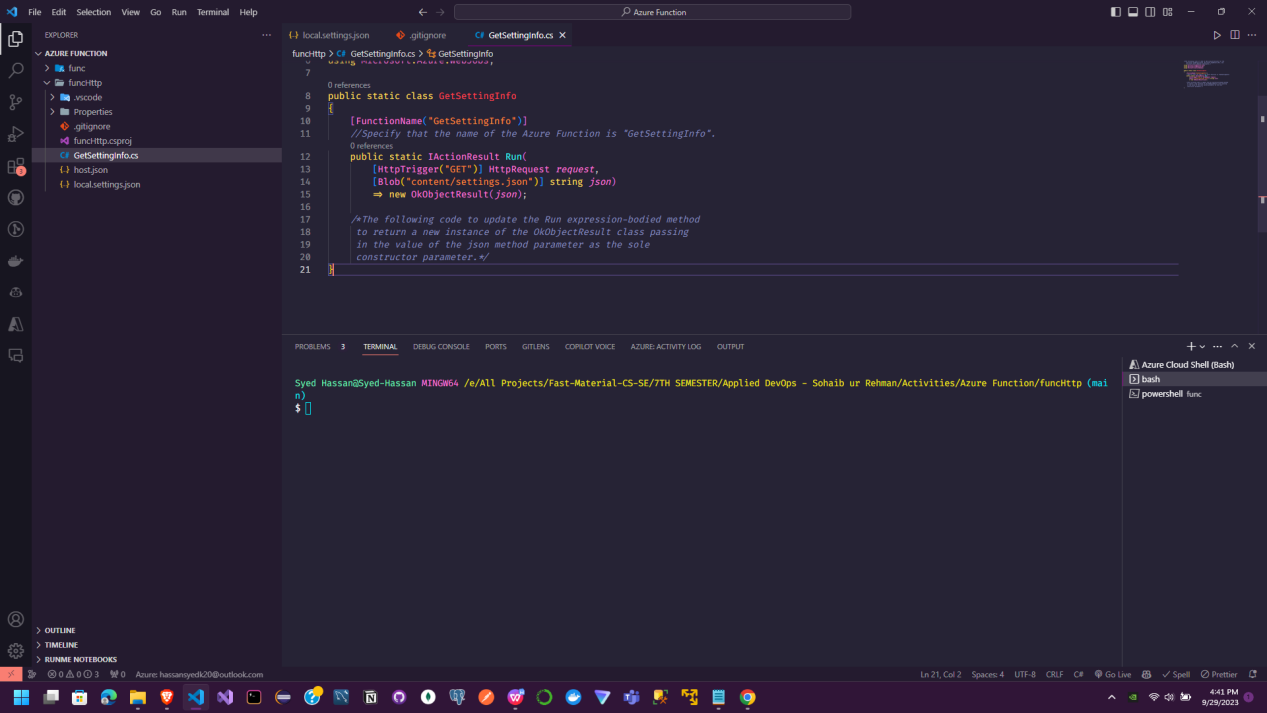


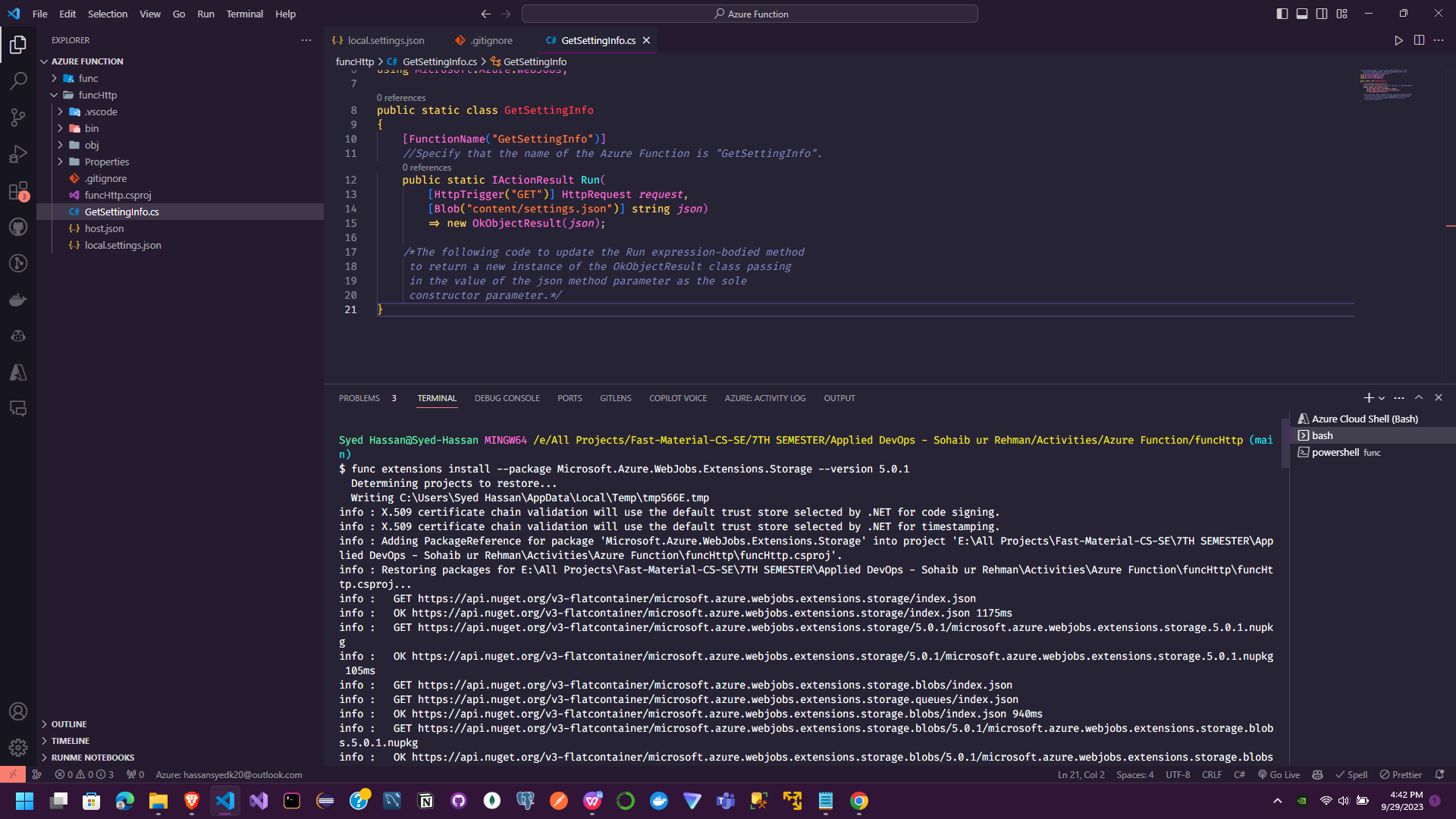
Connection String: DefaultEndpointsProtocol=https;AccountName=funcstorhassanalik201052;AccountKey=623NttB7CcuMTRaTF46vlBd7RSM186g2kOzWmn5uOHxdxkgfvZRB2ZwZAEDQjTF4Sx6UMz0GFT9I+ASttIdMhw==;EndpointSuffix=core.windows.net

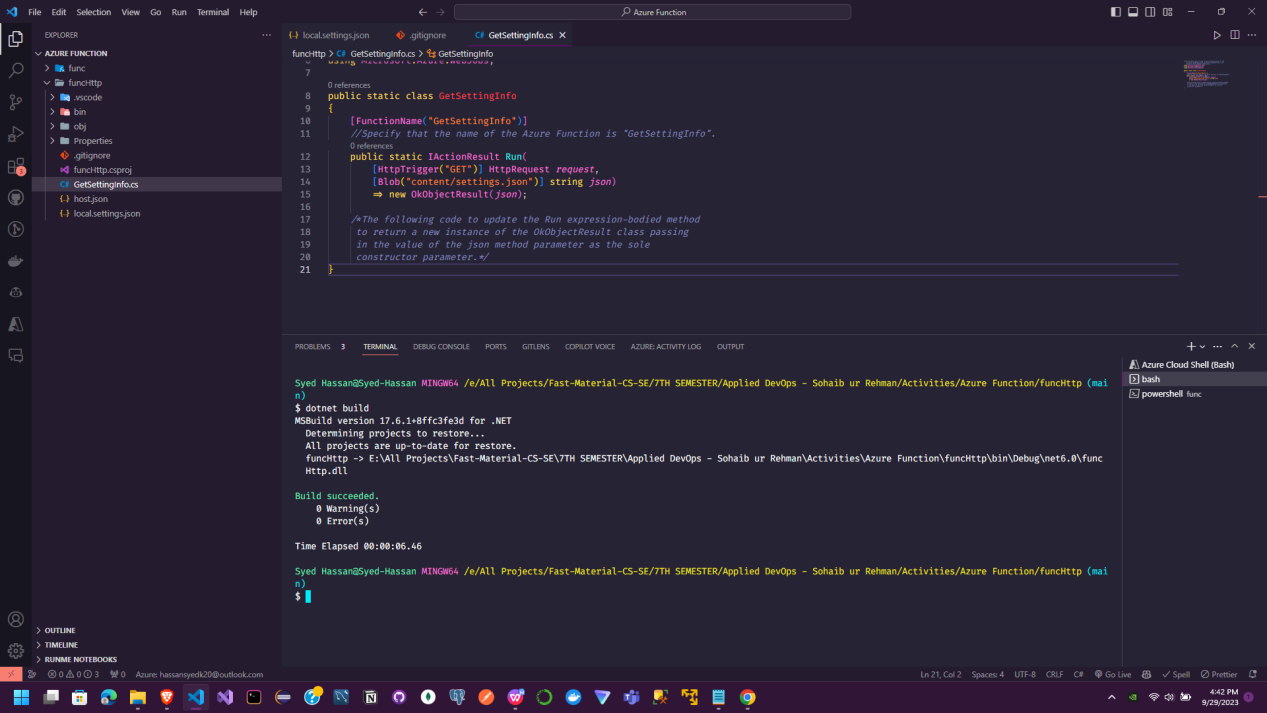




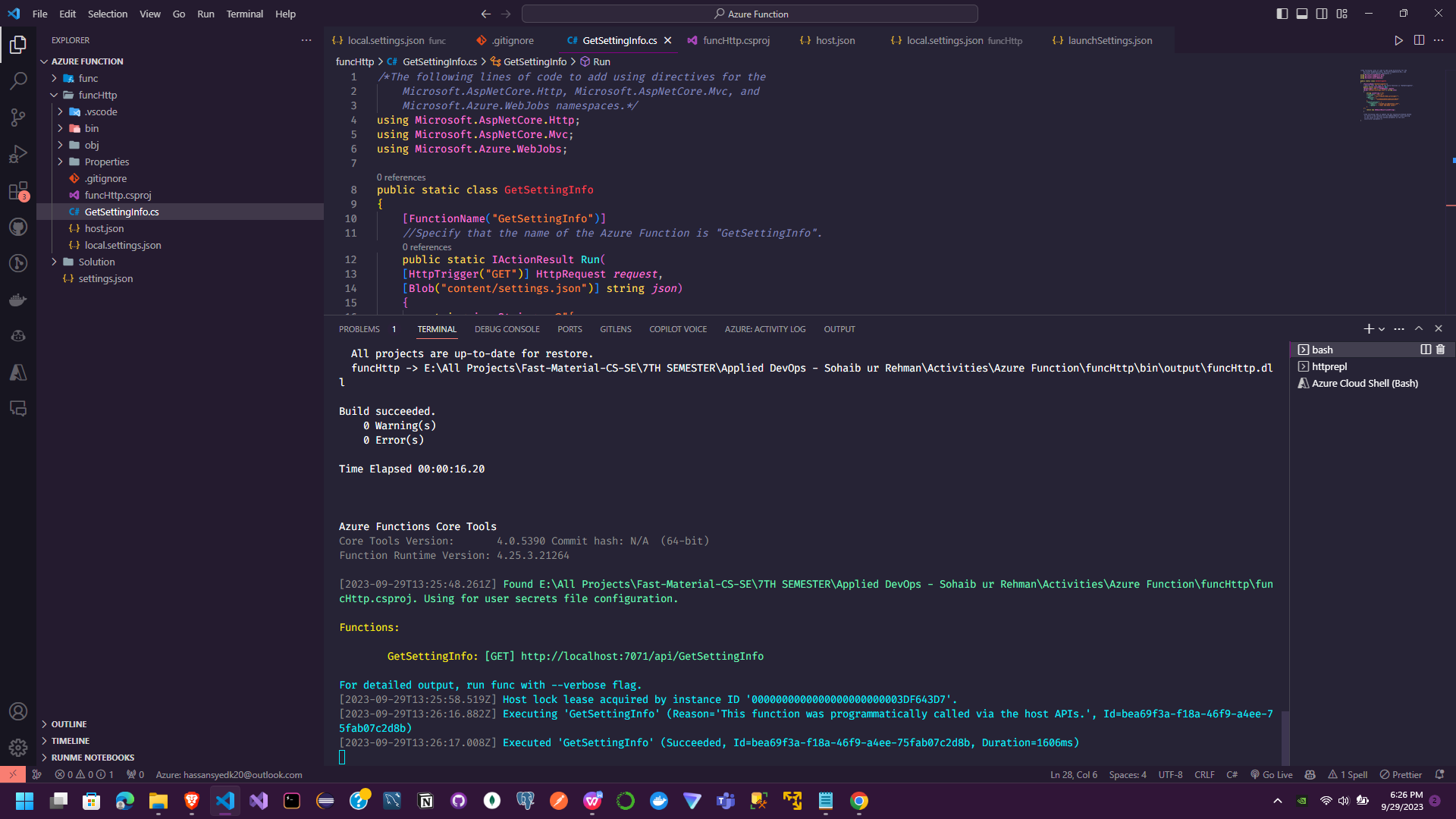


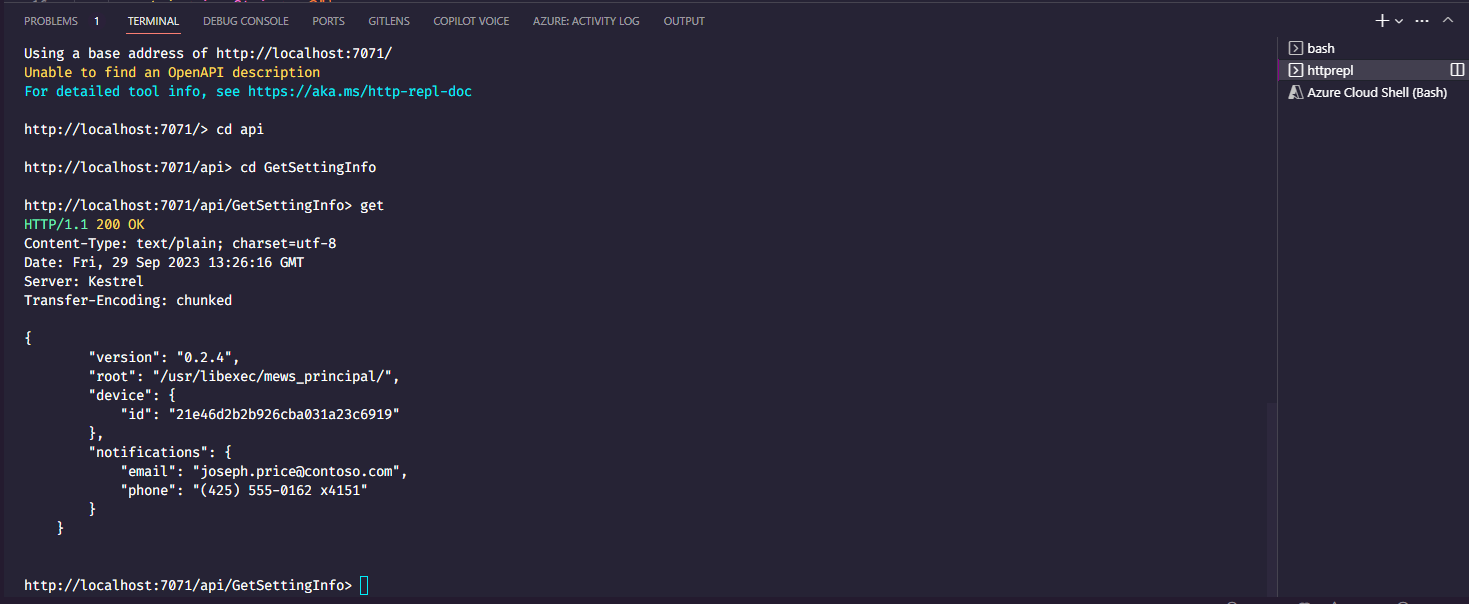












Here are the major steps for implementing task processing logic using Azure Functions in the lab scenario, and the reasons why each step is important:

1. \*\*Create Azure resources\*\* like a Storage account and Function app. This provides the necessary infrastructure to develop and host the Azure Functions. The Storage account is used to store function code and bindings. The Function app provides the compute hosting.

2. \*\*Configure a local Azure Functions project\*\* using the Azure Functions Core Tools. This allows you to develop the functions locally before deploying to the cloud. The local.settings.json file is updated with the Storage account connection string for development purposes.

3. \*\*Create an HTTP triggered function\*\*. This function will be invoked over HTTP and demonstrates how to process requests and return responses. The function echoes back the request body.

4. \*\*Create a schedule triggered function\*\*. This function runs on a schedule to demonstrate background processing. The function writes a log message each time it runs.

5. \*\*Create a function that integrates with Azure Storage\*\*. This function retrieves a blob from Azure Storage to demonstrate connecting Functions to other services. The function returns the blob contents in the HTTP response.

6. \*\*Deploy the function project to Azure\*\*. After developing and testing locally, the function project is deployed to the Function app in Azure. This makes the functions accessible over HTTP.

In summary, the steps cover creating the required resources, developing the functions locally, implementing different function triggers, integrating with other services, and deploying the final solution to the cloud. This provides a comprehensive example of using Azure Functions for task processing logic.