In this lab first we will test our code locally which we later on going to deploy at GCP Functions

In real Environment as well we follow same process, first we test our code locally and only then we push it to the cloud function. This function will be based on dotnet code

1. Go to ReadIt app and create a new folder naming order-function
2. Open CMD as admin and go to this folder in cmd  
   A screenshot of a computer program

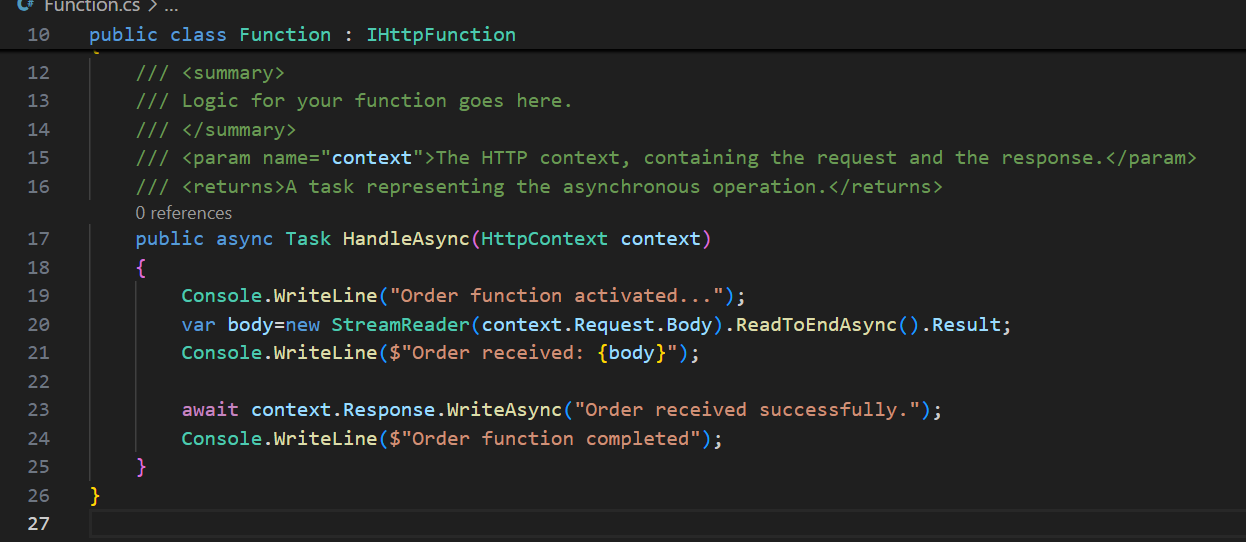
   AI-generated content may be incorrect.
3. First we will install the tool that will be required to create a google cloud function project in VS code. First run below command in CMD prompt  
   dotnet new -i Google.Cloud.Functions.Templates  
   Then run below command  
   dotnet new gcf-http  
   A screenshot of a computer program

   AI-generated content may be incorrect.
4. Now you will notice some files has been created in the order-function folder we created  
   A close up of a computer screen

   AI-generated content may be incorrect.
5. Open this folder in VSCode and make sure to click Yes if see message to add missing assets  
   A screenshot of a computer

   AI-generated content may be incorrect.
6. As of now it is just simple Hello function framework, but we will still test this if it runs locally. So hit F5. As we have seen in previous lab, generally when we build a app locally, it opens a web url, but not in this case  
   so we will open it manually. Copy the url as showing below which would look like something http://127.0.0.1:8080  
   A screenshot of a computer screen

   AI-generated content may be incorrect.
7. Open the URL in browser  
   A screenshot of a chat

   AI-generated content may be incorrect.
8. As we are able to access the webpage, it shows that function is working
9. So now we will use our own code. For this replace the Function.cs file from order-function.zip file to this existing file  
   now if we go through this new file, we can see some quite simple things.  
   Here first logs says “order function activated” will be created when the function is triggered, then we are going to read the body of the function  
   
10. Now again hit F5 to build this new code and it will again generate same url as like previous. But this time we cannot test it in the browser, as browser cannot set body of a request  
    So we will use a tool named postman
11. So first we need to download postman, for this we can use below link  
    <https://www.postman.com/downloads/>   
    Right click on downloaded file and run as administrator  
    If asked, click continue without account  
    A screenshot of a login form

    AI-generated content may be incorrect.
12. Then click open lightweight api client  
    A screenshot of a computer

    AI-generated content may be incorrect.
13. From drop down menu, select Post, paste the url, go to body and select raw. The from drop down menu select json  
    Now, open the second file(ordersample.json) from the zip folder and paste its data in postman and click send  
    A screenshot of a computer

    AI-generated content may be incorrect.
14. We should get a response as order received successfully  
    A screenshot of a computer

    AI-generated content may be incorrect.
15. Now go back to VSCode and scroll down, there should be able to see the order details which we put in the body  
    A computer screen shot of a black screen

    AI-generated content may be incorrect.
16. Now the above activity shows that our function or code is running, and also capable of receiving the requests