Tutorial Azure Function

Azure Functions is a serverless computing service provided by Microsoft Azure. It allows developers to build and deploy small pieces of code, called functions, that can be executed in response to events or triggers. These functions can be written in various programming languages, such as C#, JavaScript, Python, and more.

Azure Functions provide a scalable and cost-effective solution for running your code in the cloud. They can be used for a wide range of scenarios, including data processing, event-driven applications, real-time analytics, and more. With Azure Functions, you only pay for the compute resources used during the execution of your functions, making it an efficient option for managing workloads.

By leveraging Azure Functions, developers can focus on writing the business logic of their applications without worrying about infrastructure management. Azure Functions seamlessly integrate with other Azure services, allowing you to build powerful and flexible applications.

Overall, Azure Functions provide a convenient and efficient way to develop and deploy serverless applications in the cloud, enabling developers to build scalable and event-driven solutions with ease.

How to start

In order to easily deploy an azure function, one needs 2 tools:

- Azure CLI https://learn.microsoft.com/en-us/cli/azure/install-azure-cli
- azure-function module https://pypi.org/project/azure-functions/

Create python virtual environment

```
python -m venv .venv
source .venv/bin/activate
```

Create a local function

```
pip install azure-functions
func init --python
func new --name HttpExample --template "HTTP trigger" --authleve
```

This part will help you generate a simple hello world application in python

```
@app.route(route="SoaFunction", auth_level=func.AuthLevel.ANONYM
def HttpExample(req: func.HttpRequest) -> func.HttpResponse:
    logging.info('Python HTTP trigger function processed a reque

    name = req.params.get('name')
    if not name:
        try:
            req_body = req.get_json()
        except ValueError:
            pass
        else:
            name = req_body.get('name')

    if name:
        return func.HttpResponse(f"Hello, {name}. This HTTP trig
    else:
        return func.HttpResponse(
             "This HTTP triggered function executed successfull
             status_code=200
        )
```

Deploy the function

In the following part, it is required to login into the azure console, such that the resource group and research storage are created

```
az login
az group create --name AzureFunctionsQuickstart-rg --location <R
az storage account create --name <STORAGE_NAME> --location <REGI
```

Then, simply create the function in azure

```
az functionapp create --resource-group AzureFunctionsQuickstart-
```

Furthermore, the last step would be to publish the function

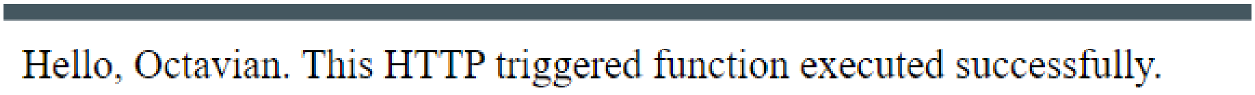
```
func azure functionapp publish <APP_NAME>
```

And this is how the azure function in python is created, accepting http requests

We can then send requests to the generated url for the function

```
https://<functionName>.azurewebsites.net/api/<function>?name=Oct
```

Which will result



Hello, Octavian. This HTTP triggered function executed successfully.