

# Large Scale Computing

## Lab 6

Oscar Teeninga


### 1. Hello World


Proszę stworzyć funkcję typu “Hello world!” oraz odpowiednią usługę REST. Wywołanie proszę zademonstrować za pomocą zewnętrznej aplikacji (np. curl).

### hello\_world\_oteeninga\_function

ThrottleCopy ARNActions

▼ Function overview Info

 hello\_world\_oteeninga\_function

 Layers (0)

+ Add trigger

+ Add destination


Description

-

Last modified

35 seconds ago

Function ARN

 arn:aws:lambda:us-east-1:365105900035:function:hello\_world\_oteeninga\_function

### Code source Info

Upload from

File Edit Find View Go Tools Window Test Deploy Changes deployed

Go to Anything (⌘ P)

Environment

hello\_world\_oteenir  
lambda\_function.py

lambda\_function x

```
1 import json
2
3 def lambda_handler(event, context):
4     # TODO implement
5     return {
6         'statusCode': 200,
7         'body': json.dumps('Hello from Lambda!')
8     }
9
```

9:1 Python Spaces: 4

### Triggers (1)


RefreshEnableDisableFix errorsDeleteAdd trigger

Find triggers

< 1 >

☐

Trigger



API Gateway: hello\_world\_oteeninga\_function-API

arn:aws:execute-api:us-east-1:365105900035:2n47k17yl8/\*/hello\_world\_oteeninga\_function

API endpoint: [https://2n47k17yl8.execute-api.us-east-1.amazonaws.com/default/hello\\_world\\_oteeninga\\_function](https://2n47k17yl8.execute-api.us-east-1.amazonaws.com/default/hello_world_oteeninga_function)

► Details

```
oscarteeninga@Oscars-MacBook-Pro-16 ~ % curl --re
quest GET https://2n47k17yl8.execute-api.us-east-
1.amazonaws.com/default/hello_world_oteeninga_fun
ction
"Hello from Lambda!"%
```

## 2. Czy funkcja uruchamia się tylko na jednym serwerze?

Czy funkcja uruchamia się tylko na jednym serwerze? Proszę sprawdzić co się stanie w przypadku wielu jednoczesnych requestów. Funkcja może np. wywoływać komendę `hostname` przy pomocy modułu `child_process`. Proszę wygenerować obciążenie, tak aby widać było skalowanie infrastruktury.

```
1 for i in {1..100}
2 do
3     curl --request GET
4         https://2n47k17yl8.execute-api.us-east-1.amazonaws.com/default/hello_world_oteeninga_function
5 done
```

```
1 import json
2 import socket
3
4
5 def lambda_handler(event, context):
6     # TODO implement
7     return {
8         'statusCode': 200,
9         'body': json.dumps(socket.gethostname())
10    }
```

```
oscarteeninga@Oscars-MacBook-Pro-16 lab6 % ./requests.sh
"169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254
.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253"
"169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254
.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253"
"169.254.95.253""169.254.95.253""169.254.95.253""169.254.199.101""169.254.199.101""169.254.199.101""169.254.199.101""169
.254.199.101""169.254.199.101""169.254.199.101""169.254.199.101""169.254.95.253""169.254.95.253""169.254.95.253""169.254
.95.253""169.254.199.101""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253"
""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.25
4.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253"
""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.25
4.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253"
""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.25
4.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253""169.254.95.253"
""169.254.95.253""169.254.95.253""169.254.95.253%"
```

Widzimy, że hostname się zmienia co jakiś czas, zatem AWS Lambda jest skalowalna i uruchamia się na kilku serwerach.

### 3. Funkcja reagująca na postawienie nowego pliku tekstowego w S3.

Proszę stworzyć funkcję reagującą na powstanie nowego pliku (tekstowego) w bucketie S3. Funkcja powinna do każdego nowego pliku stworzyć plik w (innym) bucketie wyjściowym, plik powinien zawierać oryginalny tekst i dopisek " uploaded at ".

\*Podpowiedź: funkcje getObject i putObject dostępne w aws-sdk dla S3\*

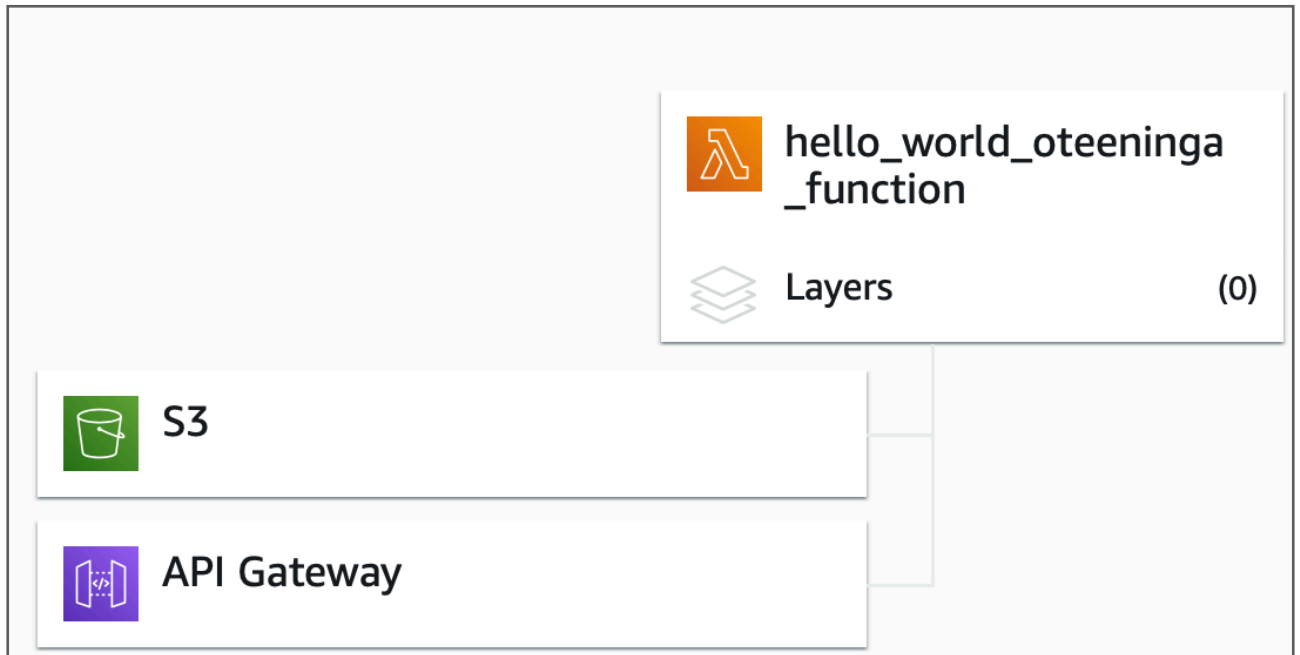
Utworzyłem jeden dodatkowy bucket, naszym wejściowym będzie ten powstały na laboratorium 5.

Buckets (2) <a href="#">Info</a>						Copy ARN	Empty	Delete	Create bucket
Buckets are containers for data stored in S3. <a href="#">Learn more</a>									
<input type="text" value="Find buckets by name"/>					< 1 >				
	Name ▲	AWS Region ▼	Access ▼	Creation date ▼					
<input type="radio"/>	bucket-oscarteeninga	US East (N. Virginia) us-east-1	Objects can be public	December 1, 2021, 20:57:17 (UTC+01:00)					
<input type="radio"/>	bucket2-oscarteeninga	US East (N. Virginia) us-east-1	Objects can be public	December 1, 2021, 23:05:49 (UTC+01:00)					

Napisałem program, który reaguje na dodanie do S3.

```
1 import boto3
2 import json
3 import urllib.parse
4
5
6
7 def lambda_handler(event, context):
8     s3 = boto3.client('s3')
9     bucket_output = 'bucket2-oscarteeninga'
10    file_name = 'oteeninga_sample.txt'
11
12    try:
13        bucket = event['Records'][0]['s3']['bucket']['name']
14        key = urllib.parse.unquote_plus(event['Records'][0]['s3']['object']['key'], encoding='utf-8')
15
16        response = s3.get_object(Bucket=bucket, Key=key)
17        file = response['Body'].read().decode('utf-8')
18        print("File: " + str(file))
19        file_extended = (str(file) + ' by s3').encode('ascii')
20        s3.put_object(Body=file_extended, Bucket=bucket_output, Key=file_name + 'copy')
21        # TODO implement
22        return {
23            'statusCode': 200,
24            'body': json.dumps(response['ContentType'])
25        }
26    except Exception as e:
27        print(e)
28        return {
29            'statusCode': 500,
30            'body': json.dumps(str(e))
31        }
```

Dodałem trigger na dodanie czegokolwiek do S3.



Stworzyłem przykładowy plik oteeninga\_sample.txt i wrzuciłem go do pierwszego bucketu.

Amazon S3 > bucket-oscarteeninga

### bucket-oscarteeninga [Info](#)

**Objects** | Properties | Permissions | Metrics | Management | Access Points

**Objects (1)**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

[Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	<a href="#">oteeninga_sample.txt</a>	txt	December 1, 2021, 23:03:20 (UTC+01:00)	66.0 B	Standard

No i niestety nie zadziałało, bo nie ma pliku w nowym bucketcie

Amazon S3 > bucket2-oscarteeninga

### bucket2-oscarteeninga [Info](#)

**Objects** | Properties | Permissions | Metrics | Management | Access Points

**Objects (0)**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

[Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
No objects You don't have any objects in this bucket.					