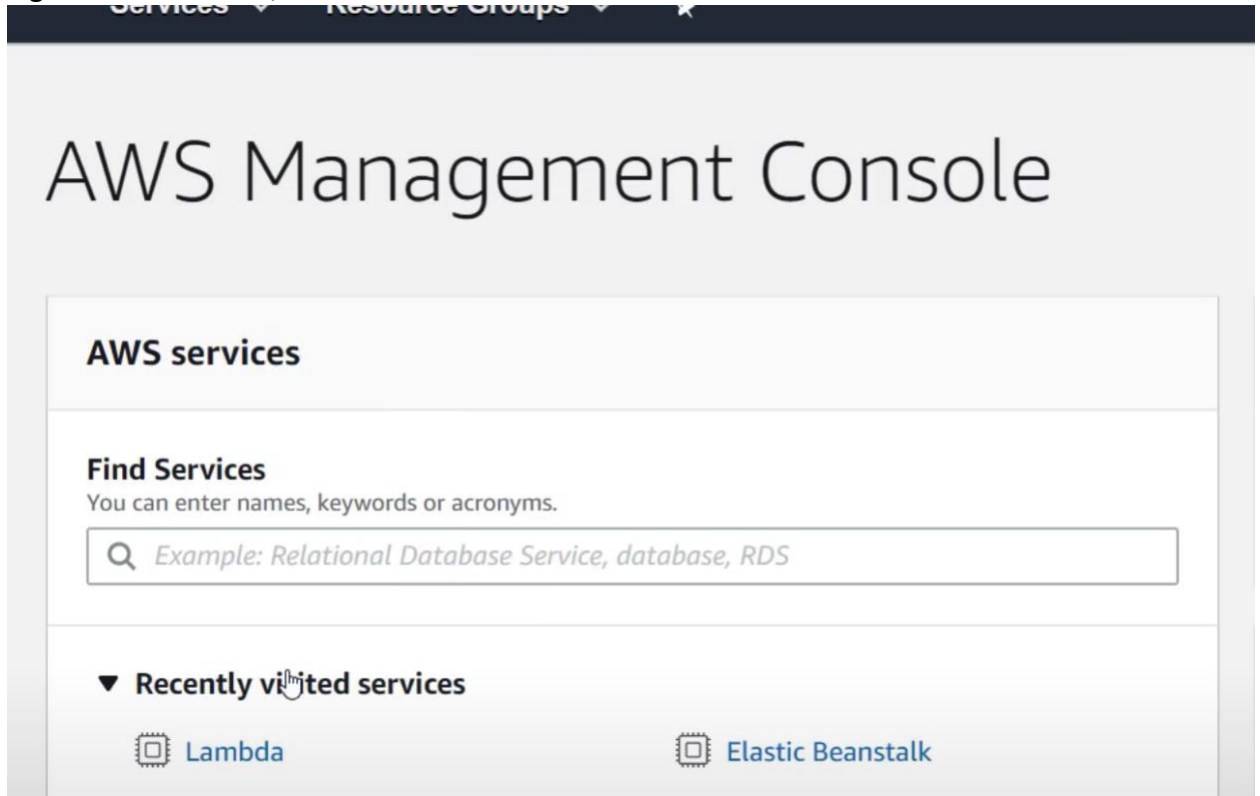
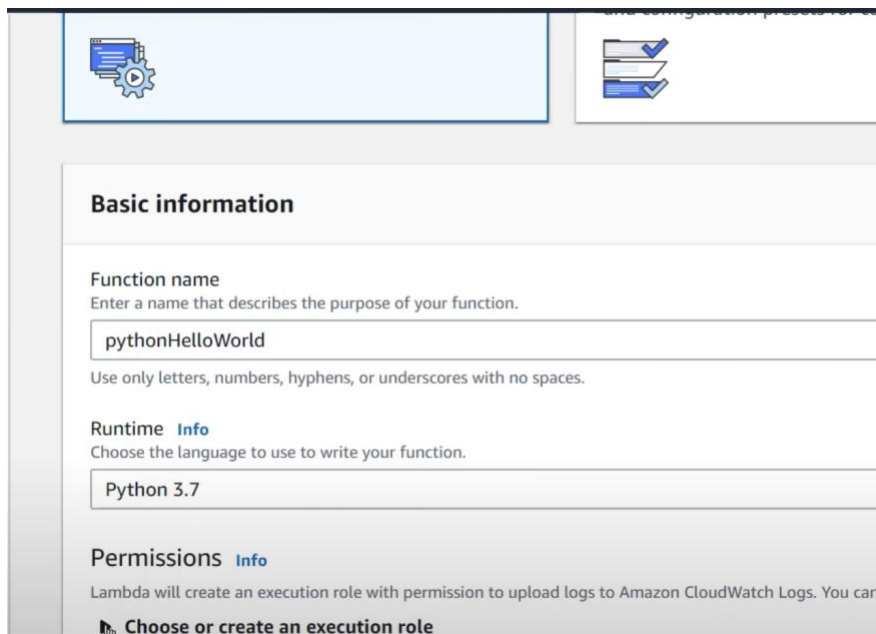


## AWS Lambda w/ Python

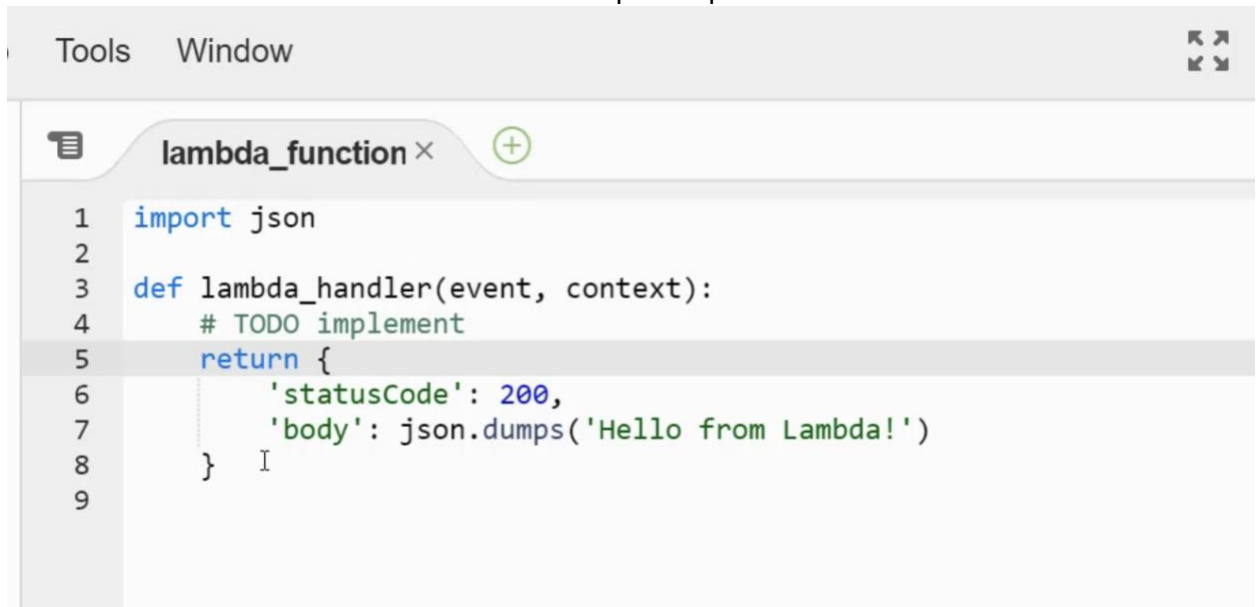
1. Sign to AWS console, and head into Lambda



2. Make sure Runtime is under the right selection: Python, since we're going to only use this language for the project.



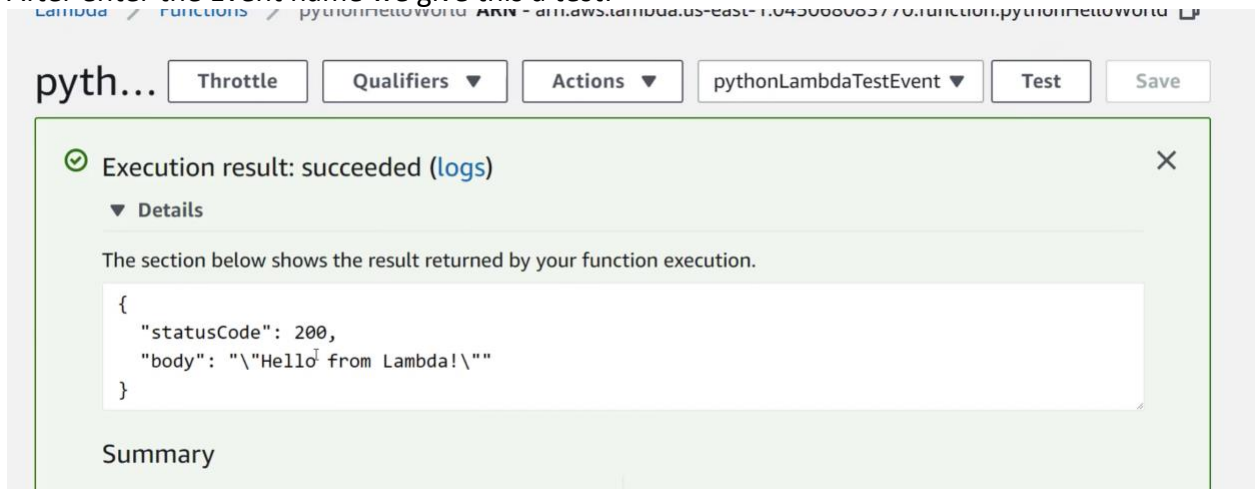
3. Function code: event and context are lambda specific parameters.



The screenshot shows a code editor with a tab labeled 'lambda\_function'. The code is as follows:

```
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
```

4. After enter the Event name we give this a test:



The screenshot shows the AWS Lambda console interface. At the top, there are buttons for 'Throttle', 'Qualifiers', 'Actions', and a dropdown menu showing 'pythonLambdaTestEvent'. To the right are 'Test' and 'Save' buttons. Below these is a green box indicating a successful execution:

✓ Execution result: succeeded (logs)

▼ Details

The section below shows the result returned by your function execution.

```
{
  "statusCode": 200,
  "body": "\"Hello from Lambda!\""
}
```

Summary

All the details we can inspect 

5. Boto3 can be imported directly without installing;  
S3: service object, then we can call various functions on this



The screenshot shows a code editor with the following Python code:

```
1 import json
2 import boto3
3
4 s3 = boto3.resource('s3')
5
```

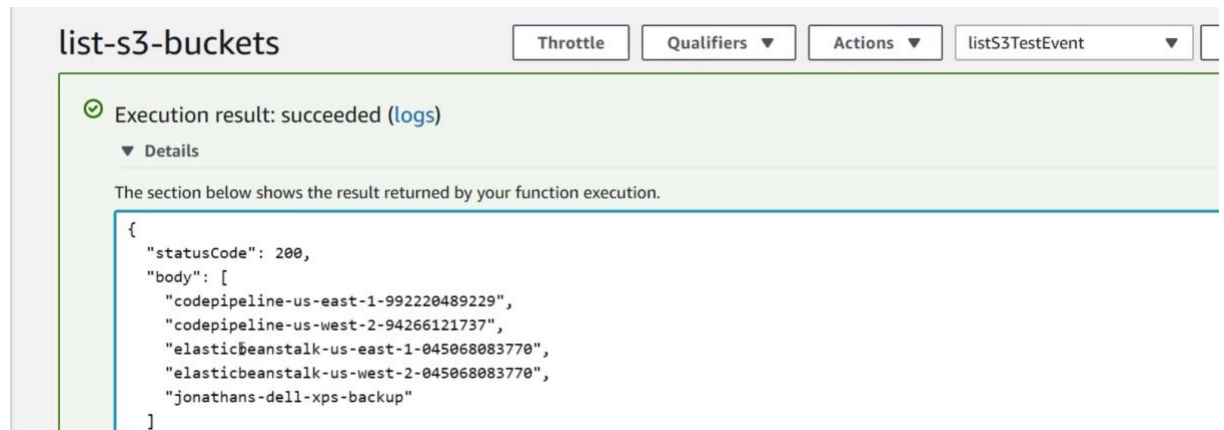
6. Define empty list;  
Return bucket list to 'body'

```

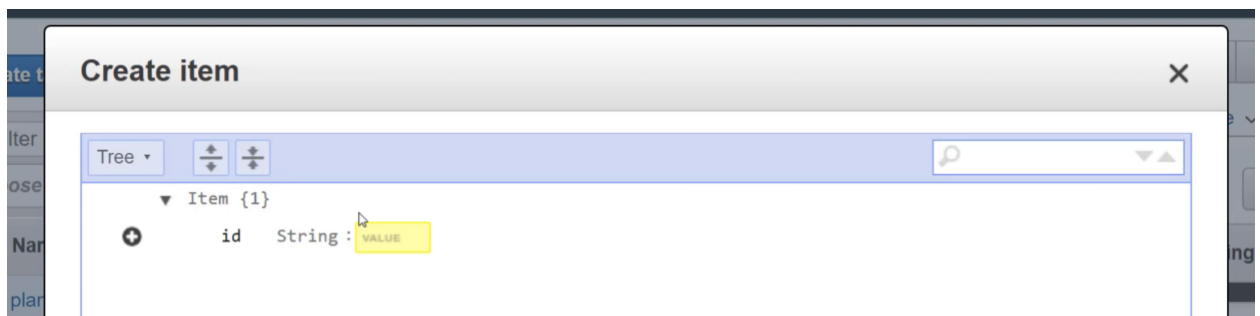
6 def lambda_handler(event, context):
7     bucket_list = []
8     for bucket in s3.buckets.all():
9         print(bucket.name)
10        bucket_list.append(bucket.name)
11    return {
12        'statusCode': 200,
13        'body': bucket_list
14    }

```

7. Make sure you give access to the test function



8. Set up a test table to have something to work on:



\*Manually adding here, need to learn a more advanced way.

9. CREATE a variable to hold our table (line 5)

```

2 import boto3
3
4 dynamodb = boto3.resource('dynamodb')
5 table = dynamodb.Table('planets')
6

```

10. `.get_item` is inside the `boto3` library

```

7 def lambda_handler(event, context):
8     response = table.get_item(
9         Key={
10             'id': 'mercury'
11         }
12     )
13

```

This should return the mercury item for us

11. Voila

▼ Details

The section below shows the result returned by your function execution.

```

{
  "statusCode": 200,
  "body": {
    "Item": {
      "id": "mercury",
      "temp": "sizzling hot"
    },
    "ResponseMetadata": {
      "RequestId": "ARGH3CA2QVH5QUS2EMEU9J7HA3VV4KQNS05AEMVJF66Q9ASUAAJG",
      "HTTPStatusCode": 200,

```

Summary

12. Add new item to the table:

```

3
4 dynamodb = boto3.resource('dynamodb')
5 table = dynamodb.Table('planets')
6
7 def lambda_handler(event, context):
8     table.put_item(
9         Item={
10             'id': 'neptune',
11             'temp': 'super cold'
12         }
13     )
14     return {
15         'statusCode': 200,
16         'body': 'Item added'
17     }
18

```

(Line8) adding new item

13. Hit test then received the new item

<input type="checkbox"/>	mercury	sizzling hot	
<input type="checkbox"/>	neptune	super cold	