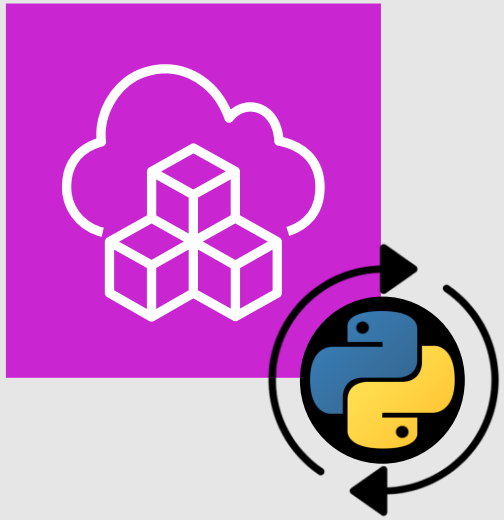
SESSION 2

The AWS Cloud Development Kit (CDK) and Python



arias.ivan@gmail.com

DigitalCloud TRAINING

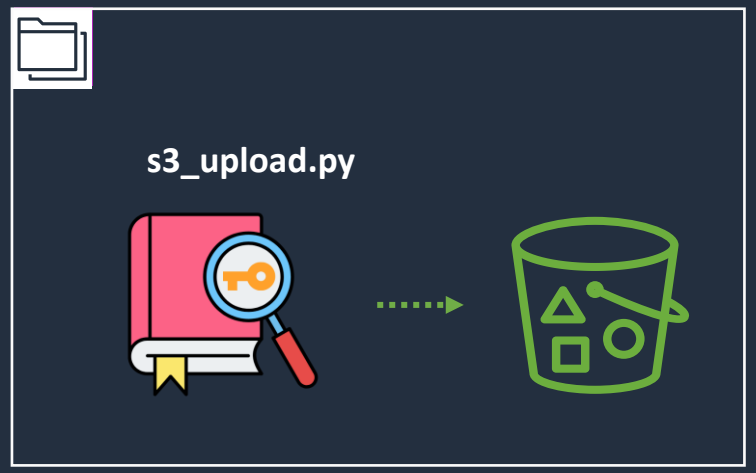
Session 2 Overview

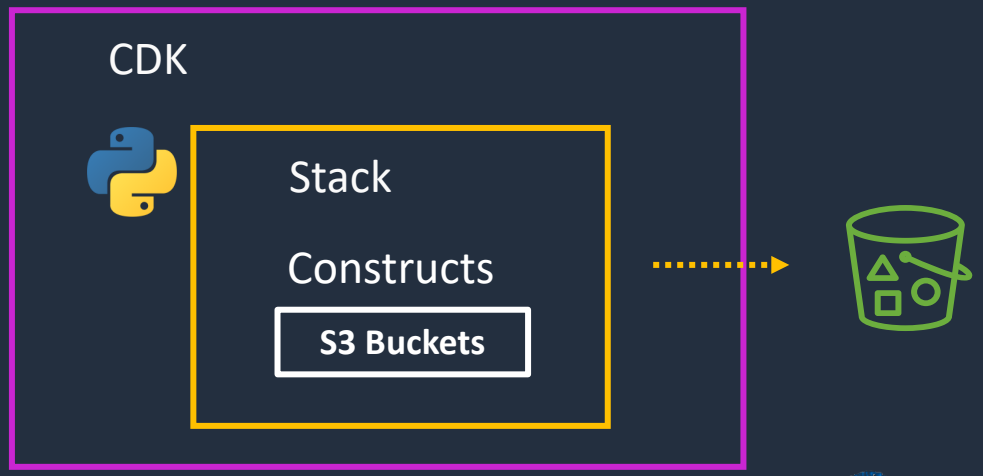
1 Dictionaries

2 AWS CDK

3 Functions

4 Hands-on Exercises





arias.ivan@gmail.com

DigitalCloud TRAINING

Dictionaries

What are dictionaries and why use them?

- As of Python version 3.7, dictionaries are an ordered collection of key-value pairs
- Dictionaries are mutable
- Think of a python dictionary as a real-world dictionary
- Each key is unique

```
1 baked_dish_ingredients = {
2     "quiche": "eggs, cheese, spinach, onions, and mushrooms",
3     "chicken pot pie": "chicken, vegetables, and gravy"
4 }
5
```

How to work with dictionaries:

- Creating a dictionary

```
baked_dish_ingredients = {}
```

- Accessing dictionary values

```
baked_dish_ingredients["quiche"]
```

- Modifying dictionary values

```
baked_dish_ingredients["empanadas"] = 3
```

- Adding and removing key-value pairs

```
baked_dish_ingredients[3.14] = "key lime pie"
```

```
baked_dish_ingredients.pop("empanadas")
```

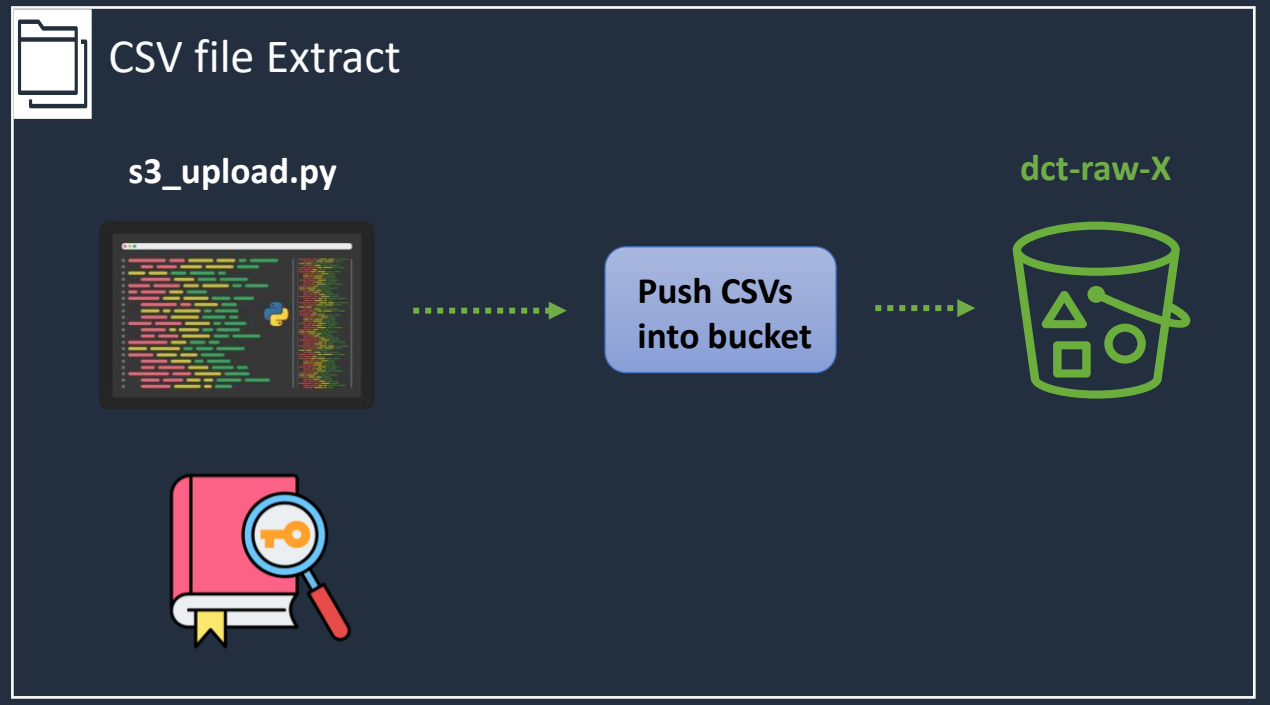
- Dictionary methods

```
baked_dish_ingredients.keys()
baked_dish_ingredients.values()
baked_dish_ingredients.items()
```


arias.ivan@gmail.com

DigitalCloud TRAINING

Hands-on Exercise: Implement Dictionary in s3_upload.py



arias.ivan@gmail.com

Questions?

arias.ivan@gmail.com

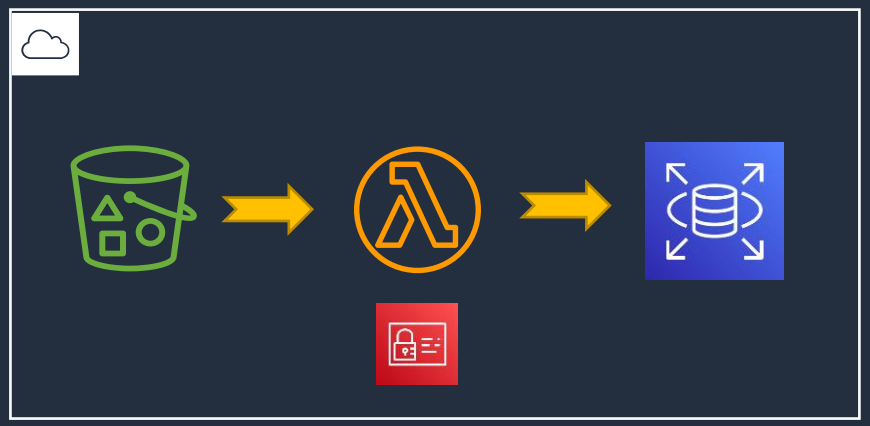
DigitalCloud TRAINING

Amazon Cloud Development Kit (CDK)

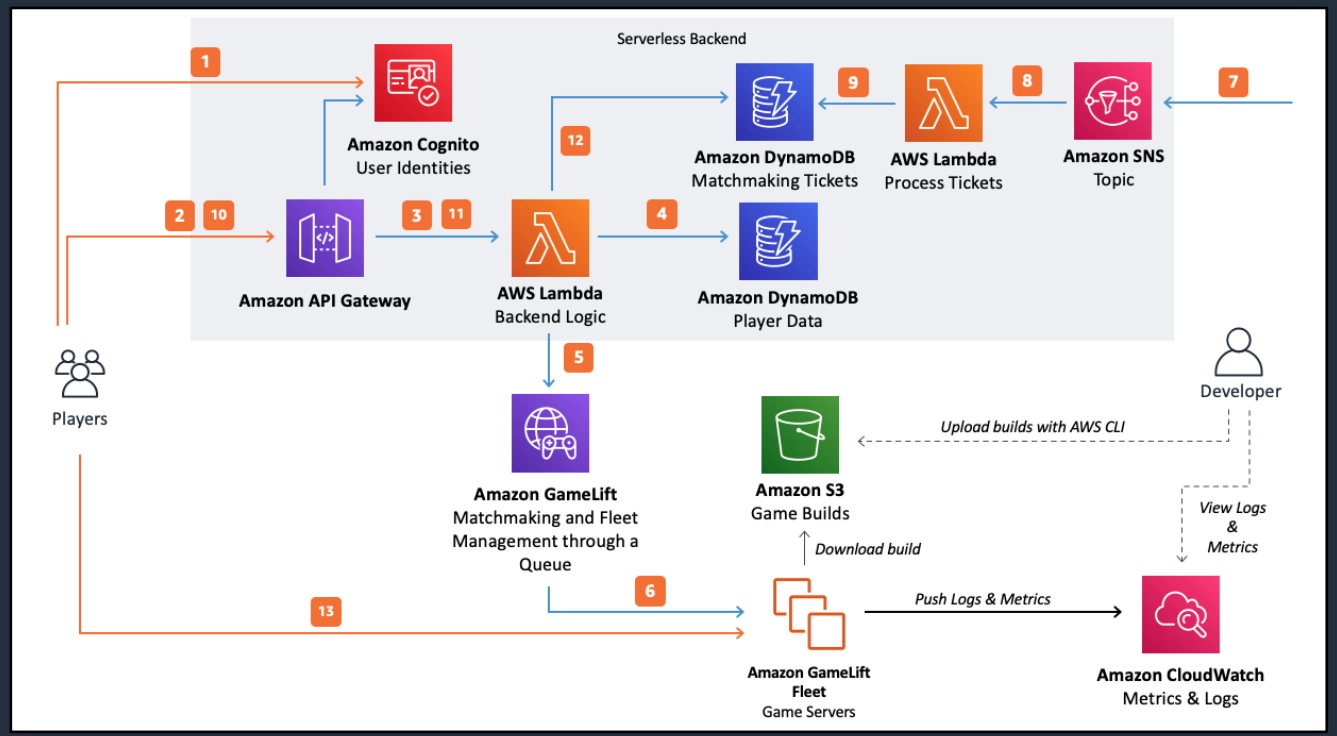
Resource Creation Through AWS Console

- **Manual Creation**

AWS Console



Manual creation does not scale very well.



arias.ivan@gmail.com

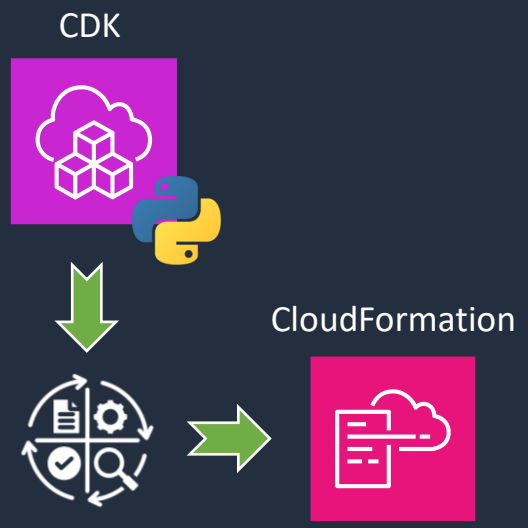
DigitalCloud TRAINING

Amazon Cloud Development Kit (CDK) Cont.

What is the AWS CDK?

An open-source software development framework for **defining cloud infrastructure in code** and provisioning it through AWS CloudFormation.

The CDK supports TypeScript, JavaScript, **Python**, Java, C#/.Net, and Go. You can use any of these supported programming languages to define reusable cloud components.



Infrastructure as Code (IaC)

Applying the same rigor of application code development to infrastructure provisioning.

Applications **cannot** be created if the code is not written according to the rules of the programming language.

When code is compiled or built into applications, we expect a consistent application to be created, and **the build is repeatable and reliable**.

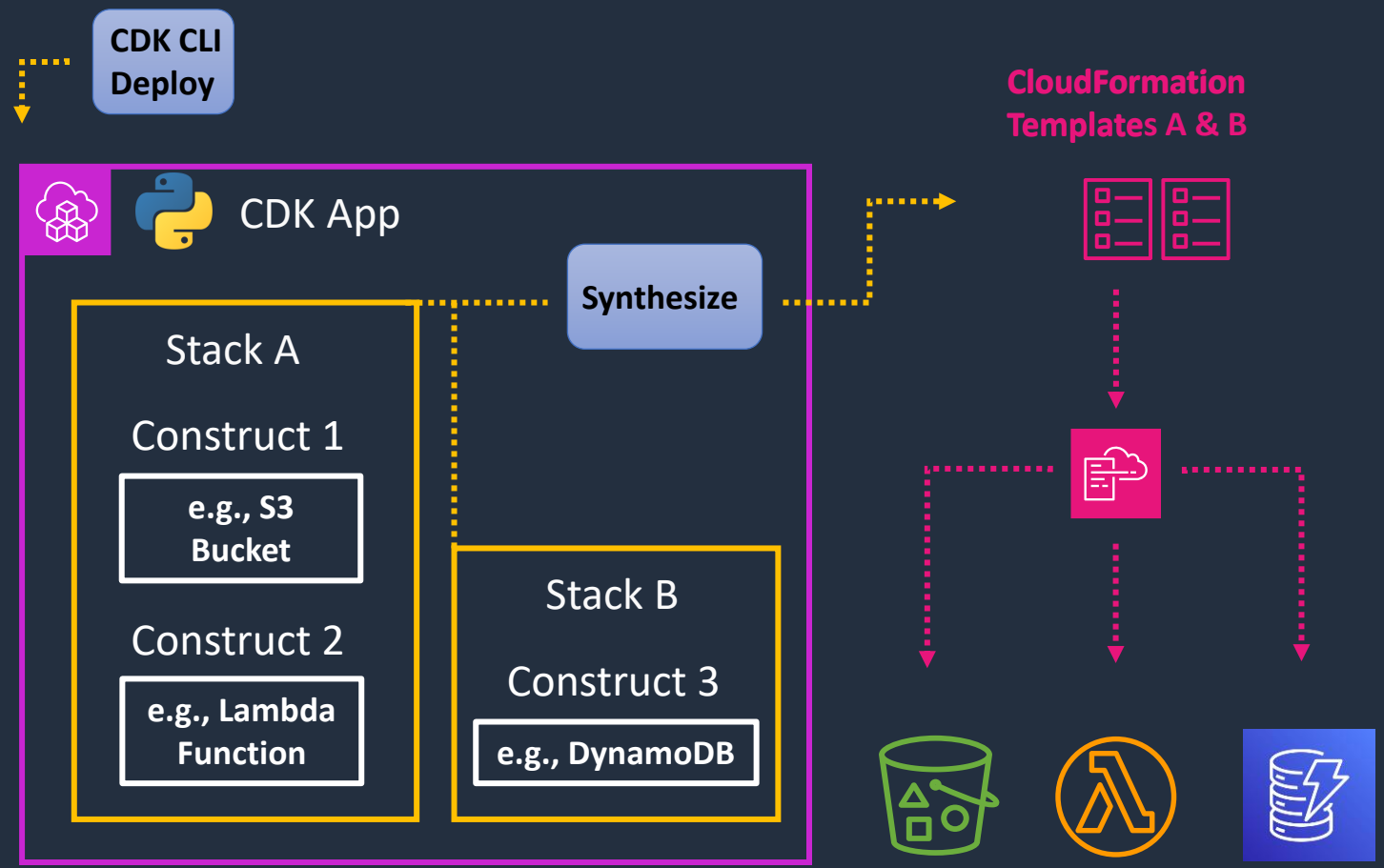
```
 9  class DctAppStack(Stack):
10
11      def __init__(self, scope: Construct, construct_id: str, **kwargs) -> None:
12          super().__init__(scope, construct_id, **kwargs)
13
14          buckets = {'my-bucket1': None, 'my-bucket2': None, 'my-bucket3': None}
15
16          for bucket_id in buckets:
17              buckets[bucket_id] = s3.Bucket(self, bucket_id)
18
```


arias.ivan@gmail.com

DigitalCloud TRAINING

Amazon CDK Cont.

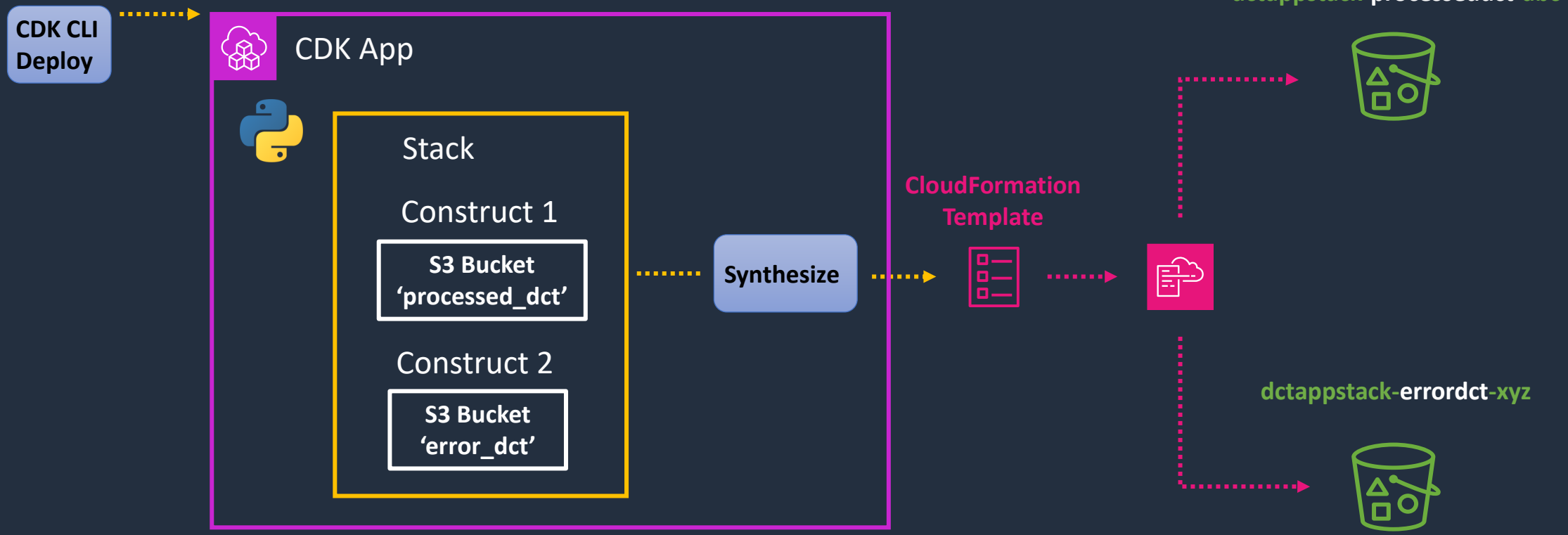
- A CDK **app** is a container (main directory) for all your resources defined in code (python).
- A **construct** is a component within your app that represents one or more AWS CloudFormation resources and their configuration.
- A **stack** is a collection of AWS resources that you can manage as a single unit
- All constructs that represent AWS resources must be defined, directly or indirectly, within the scope of a Stack construct.



arias.ivan@gmail.com

DigitalCloud TRAINING

Hands-on Exercise: CDK App



arias.ivan@gmail.com

Questions?

arias.ivan@gmail.com

DigitalCloud TRAINING

10 Minute Break

arias.ivan@gmail.com

Functions

What is a Function?

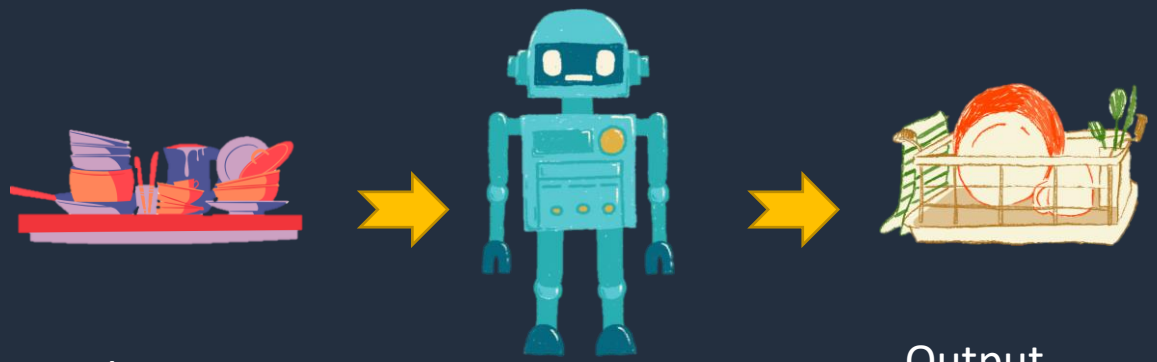
- A block of organized and reusable code that performs a specific task.
- Functions help break a program into more manageable pieces.
- Essential for code that is:
  - Efficient
  - Maintainable
  - Scalable
- Built-in & user-defined functions.
- Think of functions as **encapsulated behaviors.**

```python
def wash_dish(dish_type):
    if dish_type == 'plate':
        wash_method = 'scrubbing hard'
    elif dish_type == 'cup':
        wash_method = 'gently washing'
    elif dish_type in ['spoon', 'fork']:
        wash_method = 'polishing'
    else:
        wash_method = 'rinsing'

    print(f"Washing a {dish_type} by {wash_method}...")
    return f"The {dish_type} is now clean!"

dirty_dishes = ['plate', 'cup', 'bowl', 'spoon', 'fork']

for dish in dirty_dishes:
    clean_message = wash_dish(dish)
    print(clean_message)
```



Inputs

wash dish function

Output

Ways to work with functions

- Calling a function.
- Passing data to a function.
- Receiving data back from a function.

arias.ivan@gmail.com

DigitalCloud TRAINING

Questions?

arias.ivan@gmail.com

DigitalCloud TRAINING

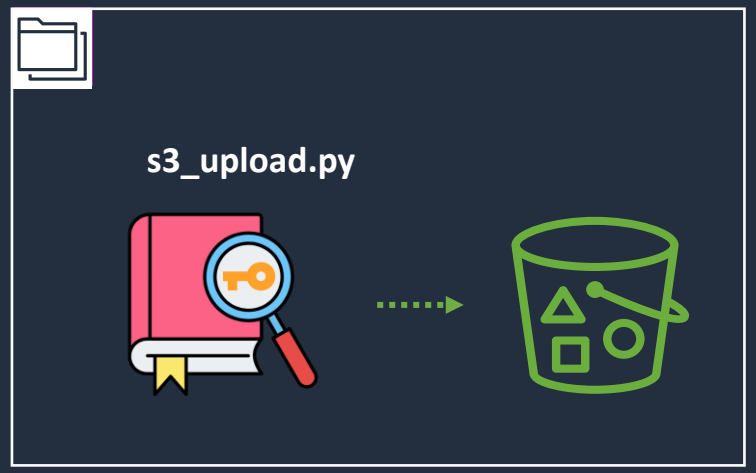
Conclusion – Session 2 Topics

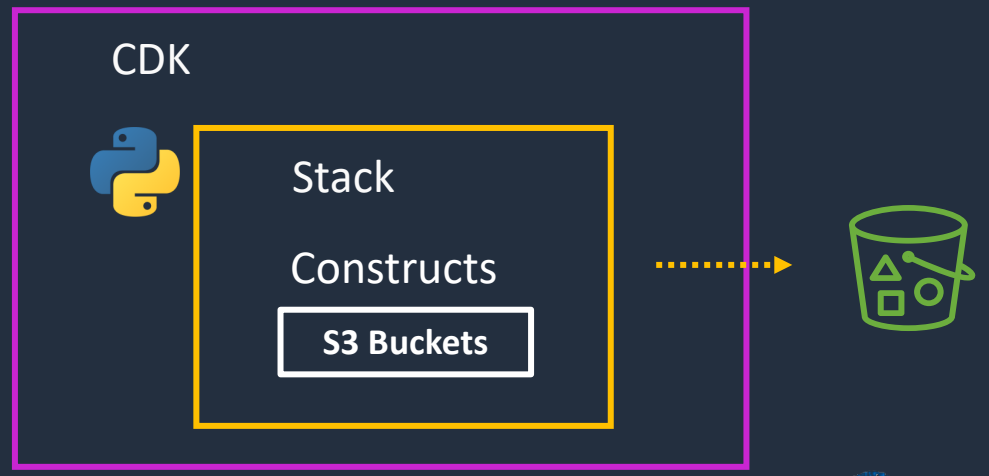
1 Dictionaries

2 AWS CDK

3 Functions

4 Hands-on Exercises



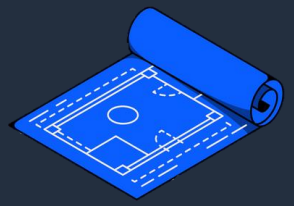


arias.ivan@gmail.com

DigitalCloud TRAINING

S3 Preview : Lambda Functions and Python Classes

1 Python Classes



2 Lambda Functions

3 Hands-on Exercises

arias.ivan@gmail.com

DigitalCloud TRAINING

HW: Define and Declare a Lambda Function with AWS CDK

Objective:

Your task is to define and declare an AWS Lambda function within the existing DctAppStack.

Resources:

Use the AWS CDK Documentation for aws_cdk.aws_lambda.Function as a reference: AWS CDK Lambda Function Documentation.

Instructions:

- Import **aws_lambda** at the top of your file under **RemovalPolicy as rp**.
- Use the aws_lambda.Function class to create a new Lambda function. Provide the following parameters:
  1. **scope**: Use ***self*** to specify that this Lambda function is a part of the current stack
  2. **id**: Give your Lambda function a unique identifier within the stack, use the string **'DataValidator'**
  3. **runtime**: Specify the runtime as ***aws_lambda.Runtime.PYTHON_3_10***
  4. **timeout**: Set the maximum execution time for your function to 10 seconds.
     - *Hint*: Find the timeout parameter in the documentation to figure out what values it accepts.
  5. **handler**: Indicate the handler function. Use the string ***'lambda_function.lambda_handler'***
  6. **code**: Point to the location of your Lambda function's code.
     - Use ***aws_lambda.Code.from_asset("***INSERT-RELATIVE-PATH-TO-DATAVALIDATOR-DIRECTORY***")***
     - Hint: Relative Path, from the perspective of top dct_app directory, to the DataValidator directory in the src folder.
2. Test your CDK code by running cdk deploy and verifying that the lambda function was properly created.

```
lambda_dv = aws_lambda.Function(self, 'DataValidator',
    <param_for_runtime>=<runtime_value>,
    <param_for_timeout>=<duration_value>,
    <param_for_handler>=<handler_function>,
    <param_for_code>=<code_location>
)
```


arias.ivan@gmail.com

Final Questions?

arias.ivan@gmail.com