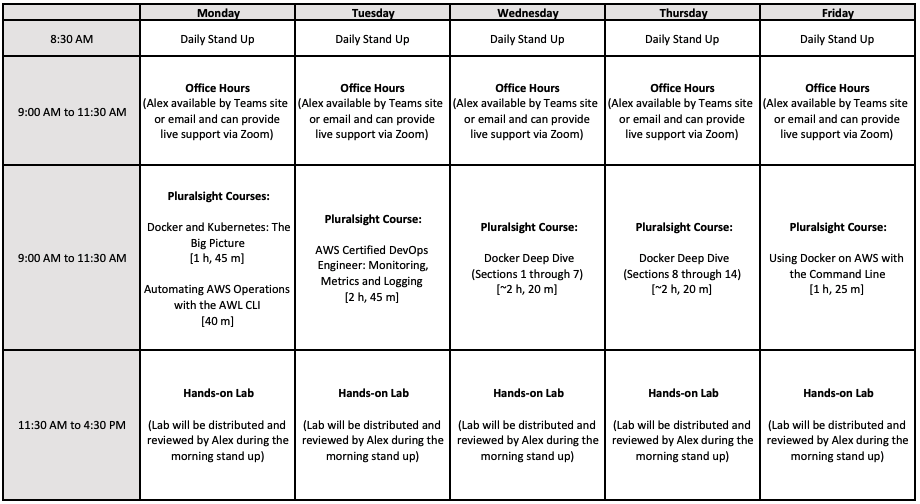
Use the class cheat sheet and materials for reference: <https://bitbucket.org/lmcohort2/materials/src/master/cheatsheet.md>



## Qwiklab

[Creating an Amazon Virtual Private Cloud (VPC) with AWS CloudFormation](https://amazon.qwiklabs.com/focuses/8642?catalog_rank=%7B%22rank%22%3A1%2C%22num_filters%22%3A0%2C%22has_search%22%3Atrue%7D&parent=catalog&search_id=4480600)

## Weather - Infrastructure as code

We are going to define all of our requirements in CloudFormation templates (CFTs). That way, we can delete everything and recreate from code!

### 1. Finish your weather lambdas

|  |
| --- |
| # Build code zip file with dependencies  rm function.zip  pip3 install --target . -r requirements.txt  zip -r9 function.zip .  # Create dependencies  aws iam create-role --role-name **<Your role name>** --assume-role-policy-document file://role-policy-document.json  # Give our role permission to write to cloudwatch logging  aws iam attach-role-policy --role-name **<Your role name>** --policy-arn "arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole"  # Deploy lambda  aws lambda create-function --function-name **<YOUR\_NAME>**-thumbnail --role "arn:aws:iam::535146832369:role/**<Your role name>**" --handler "thumbnail.handler" --runtime "python3.7" --zip-file fileb://function.zip  # Update lambda  aws lambda update-function-code --function-name **<YOUR\_NAME>**-thumbnail --zip-file fileb://function.zip |

### 2. Create a CloudFormation Template for our lambdas

Instead of creating every resource one-at-a-time, let's write a CFT that has all of our resources in it. But do this iteratively. Get the role defined, then the first lambda, etc.

One nifty thing you can do is use the following command that will upload your zip file to an s3 bucket, then output an updated version of your CFT to point to it. Just put "Code: /path/to/function.zip" instead of the documented fields. Otherwise, upload your .zip file to an s3 bucket you create.

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/using-cfn-cli-package.html>

|  |
| --- |
| #Create your bucket  aws s3 mb s3://**<YOUR\_NAME>**-lm-code-bucket  # Package code  aws cloudformation package --template-file cft.yaml --s3-bucket **<YOUR\_NAME>**-lm-code-bucket) |

**Helpful** **Docs**:

Grabbing individual attributes off another defined resource: [GetAtt](https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/intrinsic-function-reference-getatt.html#intrinsic-function-reference-getatt-examples)

[Lambda Function Properties](https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-lambda-function.html#aws-resource-lambda-function-properties)

[IAM Role Properties](https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-iam-role.html#aws-resource-iam-role-properties)

[API Gateway](https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/AWS_ApiGateway.html)

Here is a skeleton of your CFT, as a starting point. you can put this in "CFT.yaml" or any filename that seems appropriate.

|  |
| --- |
| AWSTemplateFormatVersion: '2010-09-09'  Resources:  # Required role for our lambda  LambdaRole:  Type: AWS::IAM::Role  Properties:  # TODO  FunctionName: **<YOUR\_NAME>**-cf-get-forecast-lambda  Tags:  - Key: lm-cohort-2  Value: true  # Location lambda definition  GetLocationLambda:  Type: AWS::Lambda::Function  Properties:  # TODO  FunctionName: **<YOUR\_NAME>**-cf-get-location-lambda  Tags:  - Key: lm-cohort-2  Value: true  # Forecast lambda definition  GetForecastLambda:  Type: AWS::Lambda::Function  Properties:  # TODO  FunctionName: **<YOUR\_NAME>**-cf-get-forecast-lambda  Tags:  - Key: lm-cohort-2  Value: true  ############ BONUS TODO, add API Gateway Definitions (this is hard from scratch) ############ |

You know you are done with this part when you can delete everything in AWS regarding your weather app and redeploy it with the following command.

|  |
| --- |
| aws cloudformation create-stack --capabilities CAPABILITY\_NAMED\_IAM --stack-name **<YOUR\_NAME>**-weather-lambda --template-body "$(aws cloudformation package --template-file cft.yaml --s3-bucket **<YOUR\_NAME>**-lm-code-bucket)" |

### 3. (If you have time) Make a Bamboo Build Plan to deploy our lambdas from 0's to heroes