

# 코드로 서버리스 API 인프라 구축하기

with Serverless Framework

천명욱

# 목차

1. 코드로 서버리스 API 인프라 구축하기
2. 목차
3. 멋진 함수를 만들었는데...
4. 다른 사람이 사용했으면 좋겠다!
5. API 는 어떻게 만들지?
6. 배포도 해야 한다고? 서버를 임대해야 해?
7. 나는 그냥 내가 만든 함수 하나 알리고 싶은건데...
8. 좋은 방법 없을까?
9. AWS Lambda 를 써보자
10. 아직도 뭔가 번거롭다...
11. 만약
12. Serverless Framework 설치
13. 새로운 프로젝트 만들기
14. 디렉터리 구조
15. handler.py
16. serverless.yml
17. 배포하기
18. 배포된 API 확인하기
19. 맛보기 후에는 더 깊이 알아보기!
20. Q & A
21. 감사합니다

# 멋진 함수를 만들었는데...

```
# prime.py
def is_prime(number):
    if number <= 1:
        return False

    if number <= 3:
        return True

    if number % 2 == 0 or number % 3 == 0:
        return False

    i = 5
    while i * i <= number:
        if number % i == 0 or number % (i + 2) == 0:
            return False
        i += 6

    return True
```



# 다른 사람이 사용했으면 좋겠다!

배포하고 싶다.  
API로 만들고 싶다.

GET

/pet/{petId} Find pet by ID

PUT

/pet Update an existing pet

DELETE

/pet/{petId} Deletes a pet

POST

/pet/{petId}/uploadImage uploads an image

API 는 어떻게 만들지?



새로운 프레임워크를 배워야하네...

# 배포도 해야 한다고? 서버를 임대해야 해?

AWS EC2 인스턴스 만들고

SSH 로 접속해서 소스 코드 받고

nginx 설치 하고 WSGI 연결해주고

계속 켜두면 비용 청구되니 다 쓰고 EC2 꺼줘야하고..

나는 그냥 내가 만든 함수 하나 알리고 싶은건데...

좋은 방법 없을까?



# AWS Lambda 를 써보자

## 장점

- 간단한 함수에 적합하다.
- 프레임워크를 배우지 않아도 된다.
- 호출 한 만큼만 비용이 청구된다.

## AWS Lambda

서버에 대한 걱정 없이 코드를 실행할 수 있습니다.

사용한 컴퓨팅 시간에 대해서만 요금을 지불하면 되고 코드가 실행되지 않을 때는 요금이 부과되지 않습니다. Lambda를 사용하여 사실상 모든 유형의 애플리케이션 또는 백엔드 서비스의 코드를 실행할 수 있고 이를 관리할 필요는 전혀 없습니다.

## 작동 방식

실행

.NET

Java

Node.js

Python

Ruby

사용자 지정 런타임

```
1 def lambda_handler(event, context):  
2     print(event)  
3     return 'Hello from Lambda!'  
4
```

# 아직도 뭔가 번거롭다...

- 초기 설정도 많고 콘솔에서 하나하나 입력하기 너무 귀찮다.
- 매번 코드 업로드 해야 하는 것도 번거롭다.
- api gateway 설정도 해야하고...

[Lambda](#) > [함수](#) > 함수 생성

## 함수 생성 [정보](#)

다음 옵션 중 하나를 선택하여 함수를 생성합니다.

### ☒ 새로 작성

간단한 Hello World 예제는 시작하십시오.

### ☐ 블루프린트 사용

샘플 코드 및 구축 Lambda 애플리케이션을 위한 구성 사전 설정을 일반적인 사용 사례를 살펴봅니다.

### ☐ 컨테이너 이미지

함수에 대해 배포할 컨테이너 이미지를 선택합니다.

## 기본 정보

### 함수 이름

함수의 용도를 설명하는 이름을 입력합니다.

myFunctionName

공백 없이 문자, 숫자, 하이픈 또는 밑줄만 사용합니다.

### 런타임 [정보](#)

Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Python 3.12

### 아키텍처 [정보](#)

함수 코드에 대해 원하는 명령 세트 아키텍처를 선택합니다.

☒ x86\_64

☐ arm64

### 권한 [정보](#)

기본적으로 Lambda는 Amazon CloudWatch Logs에 로그를 업로드하는 권한을 가진 실행 역할을 생성합니다. 이 기본 역할은 나중에 트리거를 추가할 때 사용자 지정할 수 있습니다.

▶ 기본 실행 역할 변경

# 만약

- 설정 파일 하나로 필요한 모든 인프라를 생성할 수 있다면?
- 코드를 업로드 하지 않아도 자동으로 배포된다면?
- 모든 인프라를 다시 명령어 한번으로 삭제할 수 있다면?

serverless  framework

# Serverless Framework 설치

```
npm install serverless -g
```

# 새로운 프로젝트 만들기

```
serverless
```

```
workspace $ serverless
```

```
Creating a new serverless project
```

```
? What do you want to make? AWS - Python - HTTP API
```

```
? What do you want to call this project? math
```

```
✓ Project successfully created in math folder
```

```
? Register or Login to Serverless Framework No
```

```
? Do you want to deploy now? No
```

주의: 적절한 권한을 가진 access key 설정이 필요.

## 권한 정책 (5)

사용자에게 직접 연결된 정책을 통해 또는 그룹을 통해 권한을 정의합니다.



제거

권한 추가 ▼

🔍 검색

필터링 기준 유형

모든 유형 ▼

< 1 >



<input type="checkbox"/>	정책 이름	▲	유형 ▼	연결 방식:
<input type="checkbox"/>	<a href="#">AmazonAPIGatewayAdministrator</a>		AWS 관리형	직접
<input type="checkbox"/>	<a href="#">AWSCloudFormationFullAccess</a>		AWS 관리형	직접
<input type="checkbox"/>	<a href="#">AWSLambda_FullAccess</a>		AWS 관리형	직접
<input type="checkbox"/>	<a href="#">IAMFullAccess</a>		AWS 관리형	직접
<input type="checkbox"/>	<a href="#">SystemAdministrator</a>		AWS 관리형 - 직무	그룹 <a href="#">SystemAdmin...</a>

# 디렉터리 구조

```
math $ tree
```

```
.  
├── README.md  
├── handler.py  
├── prime.py  
└── serverless.yml
```



# handler.py

```
import json

from prime import is_prime

def handle(event, context):
    number = int(event["queryStringParameters"]["number"])

    if is_prime(number):
        return f"{number} is prime!"

    return f"{number} is not prime!"
```

# serverless.yml

```
service: math
frameworkVersion: '3'

provider:
  name: aws
  runtime: python3.9
  region: ap-northeast-2

functions:
  prime:
    handler: handler.handle
    events:
      - httpApi:
          path: /
          method: get
```

# 배포하기

```
serverless deploy
```

```
math $ serverless deploy
```

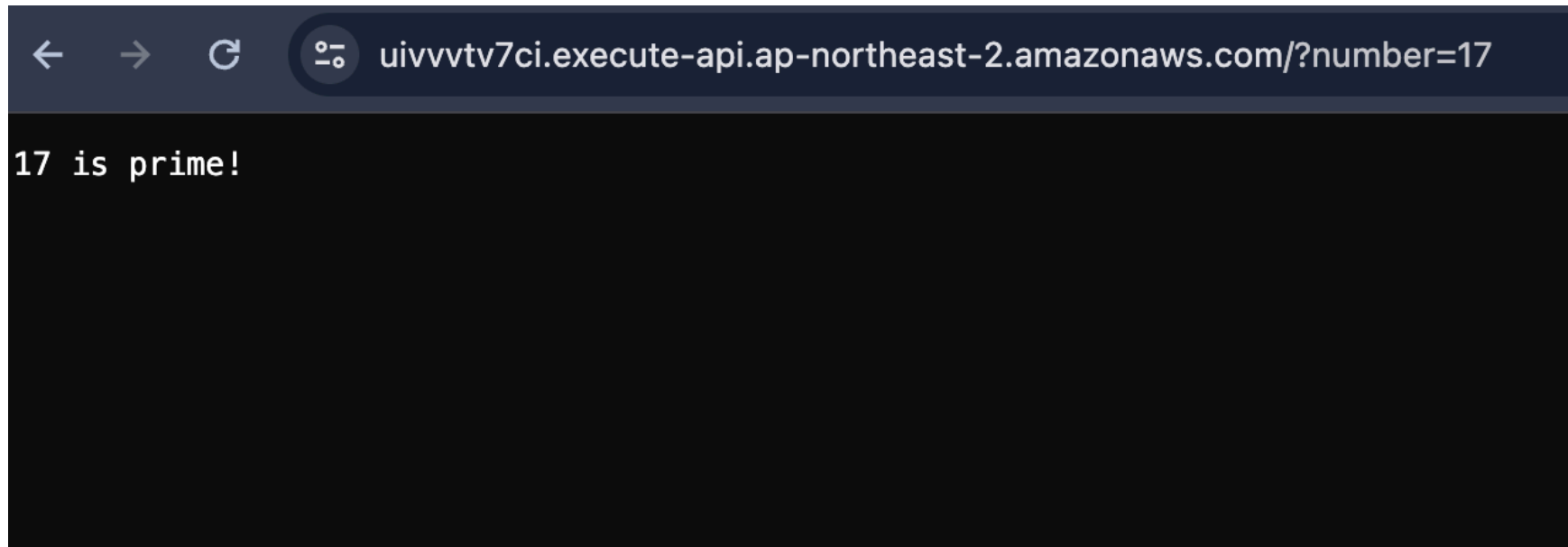
```
Deploying math to stage dev (ap-northeast-2)
```

```
✓ Service deployed to stack math-dev (91s)
```

```
endpoint: GET - https://uivvv7ci.execute-api.ap-northeast-2.amazonaws.com/  
functions:
```

```
  prime: math-dev-prime (2.3 kB)
```

# 배포된 API 확인하기



맛보기 후에는 더 깊이 알아보기!

## Q & A

- CloudFormation 으로 하면 되는 거 아닌가요?
- 현업에서도 사용하나요?
- 비용은 어떻게 되나요?
- apig 3.5\$ per million requests
- lambda 0.2\$ per million requests

감사합니다

twins1040@gmail.com

