

Kotlin で AWS Lambda 動かしてみた

Server-Side Kotlin Meetup vol.9
@Kaito-Dogi



自己紹介

- ❑ どぎー
- ❑ 株式会社ゆめみ
- ❑ Android エンジニア
- ❑ AWS 勉強中 🔥

 @Kaito_Dogi

 @Kaito-Dogi



サーバーサイド Kotlin

レス
サーバー ~~サード~~ Kotlin

サーバーレスとは

- ❑ サーバーを**意識せずに**システムを構築
 - ❑ サーバーの事前準備や、メンテナンスなどの管理が不要
 - ❑ 「アプリケーションの開発」という本質に集中できる
 - ❑ 「サーバーがない」という意味ではない！
- ❑ コンピューティングやデータストア

サーバーレスとは

- ❑ サーバーを**意識せずに**システムを構築
 - ❑ サーバーの事前準備や、メンテナンスなどの管理が不要
 - ❑ 「アプリケーションの開発」という本質に集中できる
 - ❑ 「サーバーがない」という意味ではない！
- ❑ コンピューティングやデータストア

サーバーレスコンピューティングとは

- ❑ 実行したいコードをアップロードするだけ
- ❑ 負荷の増減に柔軟に対応できる（スケーラブル）
 - ❑ リクエストに応じて自動スケーリング
- ❑ 利用した分だけ費用がかかる（従量課金制）
- ❑ イベント駆動
 - ❑ イベントが発生したときのみコンピューティング
- ❑ AWS Lambda, Cloud Functions など

サーバーレスコンピューティングとは

- ❑ 実行したいコードをアップロードするだけ
- ❑ 負荷の増減に柔軟に対応できる（スケーラブル）
 - ❑ リクエストに応じて自動スケーリング
- ❑ 利用した分だけ費用がかかる（従量課金制）
- ❑ イベント駆動
 - ❑ イベントが発生したときのみコンピューティング
- ❑ **AWS Lambda**, Cloud Functions など

AWS Lambda とは



- ❑ AWS のサーバーレスコンピューティングサービス
- ❑ AWS の他のサービスと連携
- ❑ Java のランタイムを提供
 - ❑ Kotlin も動かせるのでは…？

AWS Lambda とは



- ❑ AWS のサーバーレスコンピューティングサービス
- ❑ AWS の他のサービスと連携
- ❑ Java のランタイムを提供
 - ❑ Kotlin も動かせるのでは…？
 - ❑ **Kotless** というフレームワークがあるらしい！

Kotless を使ってみよう 🎉



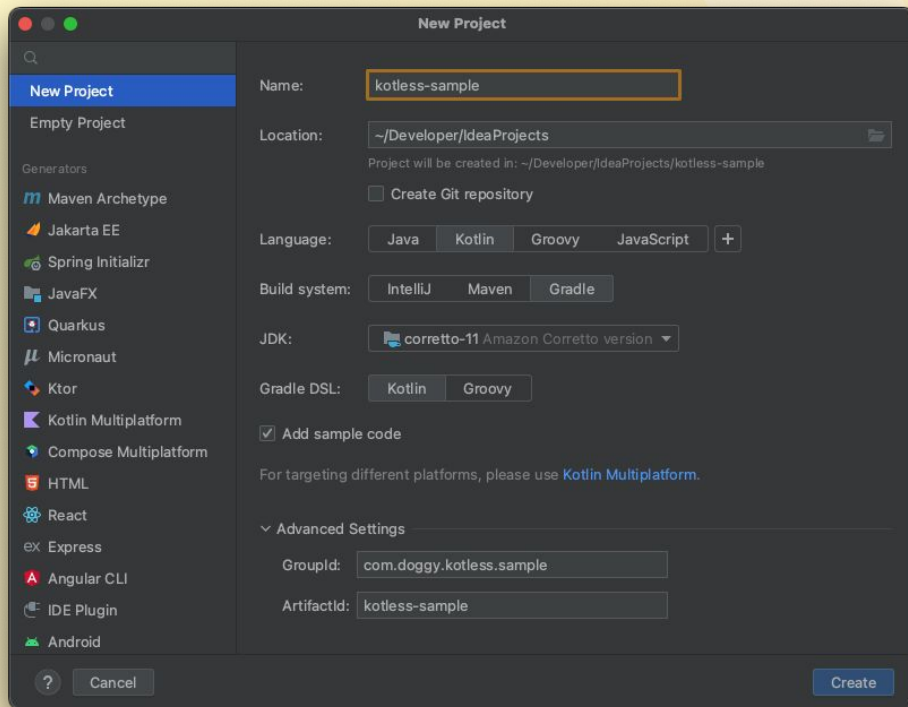
Kotless とは

- ❑ Kotlin のサーバレス・フレームワーク
- ❑ JetBrains 社が開発
- ❑ 2つの特徴
 - ❑ **DSL でアプリケーションを実装**できる
 - ❑ Kotless, Ktor, Spring Boot
 - ❑ **Gradle タスクでデプロイ**できる

下準備

- ❑ S3 のバケットを作成
 - ❑ Terraform の tfstate ファイル
 - ❑ アプリケーションの jar ファイル
- ❑ AWS CLI のクレデンシャルを取得
 - ❑ 必要に応じて IAM ユーザーの作成

プロジェクトの作成



Gradle ビルドファイルの編集

build.gradle.kts

```
plugins {  
    kotlin("jvm") version "1.6.10" apply true  
    id("io.kotless") version "0.2.0" apply true  
}  
  
repositories {  
    mavenCentral()  
    maven(url = uri("https://packages.jetbrains.team/maven/p/ktls/maven"))  
}  
  
dependencies {  
    implementation("io.kotless", "kotless-lang", "0.2.0")  
    implementation("io.kotless", "kotless-lang-aws", "0.2.0")  
    // 省略  
}  
  
tasks.withType<KotlinCompile>() {  
    kotlinOptions.jvmTarget = "11"  
}
```

Kotlin 1.6.20 以上
未対応…

Issue #121 参照

使用する DSL によって
必要な依存関係が異なる

Gradle ビルドファイルの編集

build.gradle.kts

```
kotless {  
    config {  
        aws {  
            storage {  
                bucket = "doggy-kotless-sample"  
            }  
            profile = "default"  
            region = "us-east-1"  
        }  
    }  
}  
  
webapp {  
    lambda {  
        kotless {  
            packages = setOf("com.doggy.kotless.sample")  
        }  
        memoryMb = 1024  
        timeoutSec = 120  
    }  
}  
}
```

作成した S3 バケットの
名前を指定

取得した
クレデンシャルを指定

アプリケーションの
パッケージを指定

Kotless DSL のみ

アプリケーションの実装

Main.kt

```
package com.doggy.kotless.sample

import io.kotless.dsl.lang.http.Get

@Get("/")
fun hello(): String = "Hello Server-Side Kotlin Meetup!"
```

パッケージの指定に
注意！

デプロイ

terminal

```
% ./gradlew plan
```

```
~~ 省略 ~~
```

```
Plan: 14 to add, 0 to change, 0 to destroy.
```

```
% ./gradlew deploy
```

```
~~ 省略 ~~
```

```
Apply complete! Resources: 14 added, 0 changed, 0 destroyed.
```

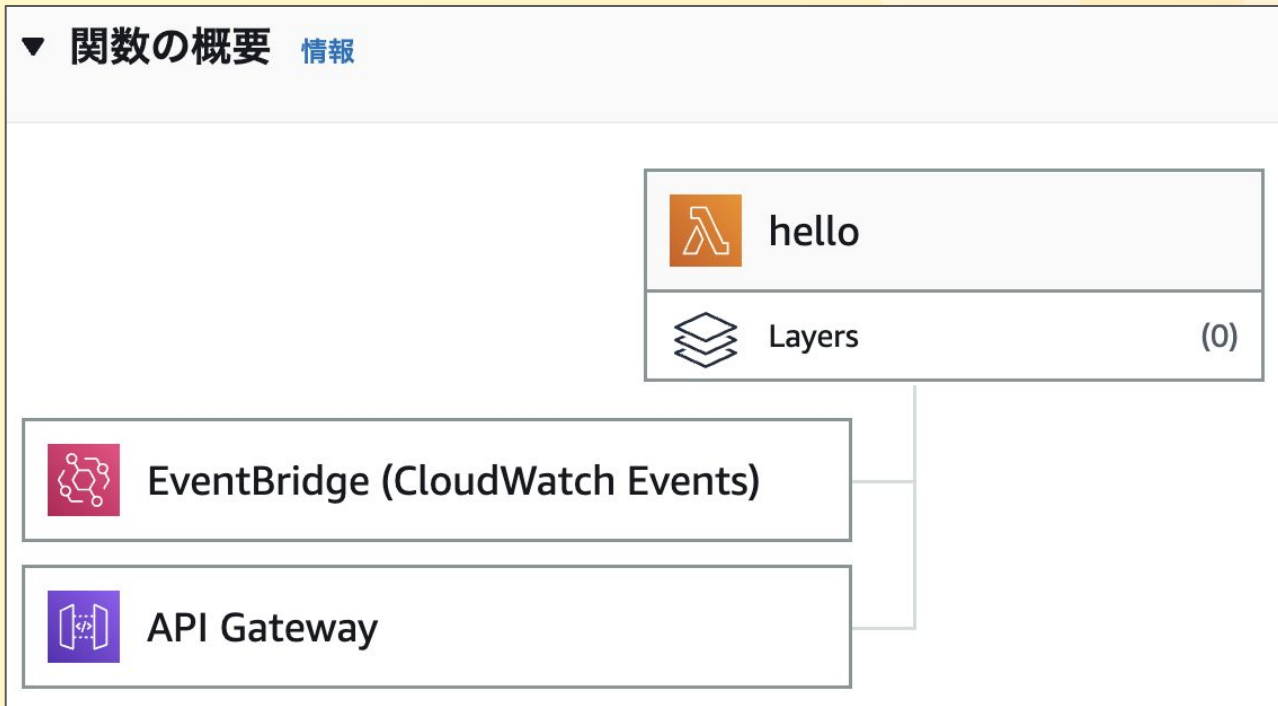
```
Outputs:
```

```
application_url = https://u7nx1b226c.execute-api.us-east-1.amazonaws.com/1
```

**Terraform の plan コマンドを実行
どのような変更が必要か確認**

**Terraform の apply コマンドを実行
実際にインフラの構成を変更**

ダッシュボードを確認



ダッシュボードを確認

トリガー (2) 情報

🔄

エラーを修正

編集

削除

トリガーを追加

🔍

トリガーを検索

< 1 >

<input type="checkbox"/>	トリガー
<input type="checkbox"/>	<div><div></div><div><div>EventBridge (CloudWatch Events): autowarm-hello</div><div>arn:aws:events:us-east-1:433790284342:rule/autowarm-hello</div><div>ルールの状態: ENABLED</div><div>▶ 詳細</div></div></div>
<input type="checkbox"/>	<div><div></div><div><div>API Gateway: kotless-sample</div><div>arn:aws:execute-api:us-east-1:433790284342:u7nx1b226c/*/GET/</div><div>API エンドポイント: https://u7nx1b226c.execute-api.us-east-1.amazonaws.com/1/</div><div>▶ 詳細</div></div></div>

エンドポイントを確認

API を叩いてみる

terminal

```
% curl https://u7nx1b226c.execute-api.us-east-1.amazonaws.com/1
```

```
Hello Server-Side Kotlin Meetup!
```



使ってみた感想

- ❑ AWS を触れて楽しかった 😊
 - ❑ いろんなサービスを連携させたい！
- ❑ Kotless の開発止まっているのが残念…
 - ❑ 最終リリースが 2021/10/19
 - ❑ 最近の Java や Kotlin のバージョンに対応していない
 - ❑ 実際の開発では使えなそう 😞
- ❑ サーバーやインフラの勉強頑張りたい 🔥
 - ❑ 初めて知ったことが多かった (Terraform など)
- ❑ Java や Gradle 周りをちゃんと理解したい 😇

サンプルコード

<https://github.com/Kaito-Dogi/kotless-sample>



参考記事

- ❑ サーバーレス・コンピューティングとは | IBM
<https://www.ibm.com/jp-ja/topics/serverless>
- ❑ Serverless Computing - AWS Lambda - Amazon Web Services
<https://aws.amazon.com/lambda/>
- ❑ Cloud Functions | Google Cloud
<https://cloud.google.com/functions>
- ❑ サーバーレスって何が便利なの？ AWS でサーバーレスを構築するためのサービスをグラレコで解説
<https://aws.amazon.com/jp/builders-flash/202003/awsgeek-serverless>
- ❑ Kotlin Serverless Framework
<https://site.kotless.io/>
- ❑ JetBrains/kotless: Kotlin Serverless Framework
<https://github.com/JetBrains/kotless>
- ❑ **KotlessでKotlinのアプリケーションをAWS Lambdaにデプロイする**
<https://blog.takehata-engineer.com/entry/deploy-kotlin-applications-to-aws-lambda-using-kotless>
- ❑ Provisioning Infrastructure - Terraform CLI | Terraform | HashiCorp Developer
<https://developer.hashicorp.com/terraform/cli/run>

ありがとうございました 🙌