# A Step by Step guide to create CRUD Restful APIs using Spring Boot

## Introduction

We will be implementing the below `CRUD Restful APIs` to manage items for a Catalogue Management System.

| HTTP <br/> Method | API Name | Path | Response <br/> Status Code |

| -------- | ----------------- | ---- | -------------------------- |

| POST | Create Catalogue Item | / | 201<br/>(Created) |

| GET | Get Catalogue Items | / | 200<br/>(Ok) |

| GET | Get Catalogue Item | /{sku} | 200<br/>(Ok) |

| PUT | Update Catalogue Item | /{sku} | 200<br/>(Ok) |

| DELETE | Delete Catalogue Item | /{sku} | 204<br/>(No Content) |

| POST | Upload Catalog Item Picture | /{sku}/image | 201<br/>(Created) |

## Technology stack for implementing the Restful APIs...

\* Zulu OpenJDK 8

\* Spring Boot v2.2.4

\* Spring Framework v5.2.3.RELEASE

\* Maven v3.6.3

\* IntelliJ Idea

## Running the Spring Boot Application

There are couple of ways to run a Spring Boot Application. During development, the ideal one would be to run the main class which is annotated with `SpringBootApplication` i.e, CrudCatalogueApplication.java in this project. And the other ways are running through maven.

### Run the application using Maven

Use the below command to run the Spring Boot application using Maven

```bash

~:\> mvn clean spring-boot:run

```

### Run the application using java -jar command

To run the application using `java -jar` command, we need to generate the package. Below are the maven and gradle command to generate the `jar` for the spring boot application.

```bash:title=maven

~:\> mvn clean package

~:\> java -jar target/catalogue-crud-0.0.1-SNAPSHOT.jar

```

```

~:\> java -jar build/libs/catalogue-crud-0.0.1-SNAPSHOT.jar

```

### Search queries

## POST

```bash

~:\ curl --location --request POST 'http://localhost:8080/cataloguemanagement/v1/' \

--header 'Content-Type: application/json' \

--data-raw '{

"id":100,

"sku":"FLST",

"name":"Clorox",

"description":"basic cleaning wipes ",

"category":"Cleaning",

"price":100

}'

```

## GET

```bash

~:\ curl --location --request GET 'http://localhost:8080/cataloguemanagement/v1/FLST' \

--data-raw ''

```

### Search Query

```bash

~:\ curl --location --request POST 'http://localhost:8080/cataloguemanagement/v1/search' \

--header 'Content-Type: application/json' \

--data-raw '{

"searchQuery":"wipes"

}'

```