



Bootcamp Go

► Práctica 1 Go Web

Objetivo

El objetivo de esta guía práctica es que podamos afianzar los conceptos sobre **Arquitectura Web**, **JSON**, entre otros vistos en este módulo. Para esto, vamos a plantear una serie de ejercicios simples e incrementales (trabajaremos y agregaremos complejidad a lo que tenemos que construir) a lo largo del módulo.

Problema

Un supermercado necesita un sistema para gestionar los productos frescos que tienen publicados en su página web. Para poder hacer esto, necesitan un **servidor** que ejecute una API que les permita manipular los productos cargados desde distintos clientes. Los campos que conforman un producto son:

Nombre	Tipo de dato JSON	Tipo de dato GO	Descripción Ejemplo
id	number	int	Identificador en conjunto de datos 15
name	string	string	Nombre caracterizado Cheese - St. Andre
quantity	number	int	Cantidad almacenada 60
code_value	string	string	Código alfanumérico característico S73191A
is_published	boolean	bool	El producto se encuentra publicado o no True
expiration	string	string	Fecha de vencimiento 12/04/2022
price	number	float64	Precio del producto 50.15

¿Are you ready? 😊👍



Ejercicio 1 : Iniciando el proyecto

Debemos crear un repositorio en github.com para poder subir nuestros avances. Este repositorio es el que vamos a utilizar para llevar lo que realicemos durante las distintas prácticas de Go Web.

- Primero debemos clonar el repositorio creado, luego iniciar nuestro proyecto de go con con el comando `go mod init`.
- El siguiente paso será crear un archivo `main.go` donde deberán cargar en una slice, desde un archivo JSON, los datos de productos. Esta slice se debe cargar cada vez que se inicie la API para realizar las distintas consultas.

El archivo para trabajar es el siguiente:

[Archivo JSON](#)



Ejercicio 2 : Creando un servidor

Vamos a levantar un servidor en el puerto 8080. Para probar nuestros endpoints haremos uso de [postman](#).

- Crear una ruta `/ping` que debe respondernos con un string que contenga `pong` con el status `200 OK`.
- Crear una ruta `/products` que nos devuelva la lista de todos los productos en la slice.
- Crear una ruta `/products/:id` que nos devuelva un producto por su id.
- Crear una ruta `/products/search` que nos permita buscar por **parámetro** los productos cuyo **precio sean mayor a un valor priceGt**.