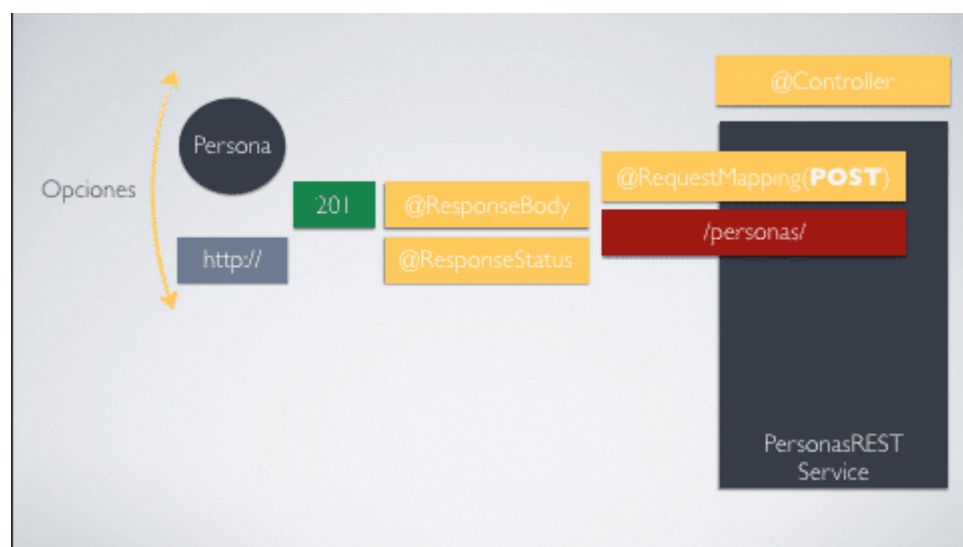


Hoy publico mi primer Webinar : **Crear un Servicio REST Sólido con Spring Framework**. Varias personas me han comentado que faltan temas que sean un poco más compactos y que se definan como píldoras formativas útiles para los desarrolladores . No es un tema sencillo de abordar porque siempre es complejo definir algo que sea útil y que sea compacto.



Intentare todos los meses generar algún contenido de este estilo. ¿Cuál es el contenido y el objetivo de este primer Webinar ?. Vamos a verlo a continuación.

Webinar :Crear un Servicio REST Sólido con Spring Framework

Este Webinar esta orientado a todas aquellas personas que quieran construir servicios REST sólidos con Spring Framework. Spring y REST son tecnologías que aportan muchas opciones a la hora de construir estos Servicios y en muchas ocasiones es difícil saber si uno esta construyendo los servicios de forma correcta ya que su flexibilidad es enorme y siempre existen varios enfoques.

No solo eso sino que muchos de los ejemplos que aparecen a nivel de búsquedas de Google son muy específicos de una versión concreta de Spring y un manejo muy puntual de un API REST no siendo validos para muchos desarrolladores.

El porque de este Webinar

Este Webinar te ayudará a perder el miedo con las diferentes anotaciones de Spring a nivel de REST como son: `@RequestMapping` , `@ResponseStatus` `@RestController`. Te permitirá gestionar de forma correcta peticiones GET , POST ,PUT , DELETE y PATCH. Te enseñara a como construir respuestas con `ResponseEntities` , gestionar los tipos de `StatusCodes` , manejar errores o como integrar DTOS (Data Transfer Objects) dentro de un Servicio REST.

Ademas abordará como ha sido la evolución de los servicios REST en Spring Framework desde la versión 3 a la versión 5 . Sirviéndonos como un buen punto de partida a la hora de comenzar a diseñar nuestra API REST estemos en la versión de Spring que estemos.

[Apuntarse al Webinar con el cupón](#)

Otros artículos relacionados

- [¿Que es REST?](#)
- [REST API Design](#)
- [Invocando un Servicio REST con HttpClient](#)