

Tabla de Contenidos



- [REST HTTP Return codes y GET](#)
- [REST HTTP Return codes y POST](#)
- [REST HTTP Return codes y DELETE](#)
- [REST HTTP Return codes y PUT](#)
- [Otros artículos relacionados](#)

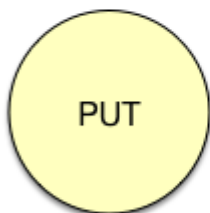
¿Cuales son los REST HTTP return codes más habituales o correctos? . Todos utilizamos hoy en día tecnologías REST y realizamos las peticiones clásicas de GET, POST, PUT y DELETE . Practicamente todo el mundo tiene muy claro que de esta forma se define un recurso REST



Obtiene



Inserta



Actualiza



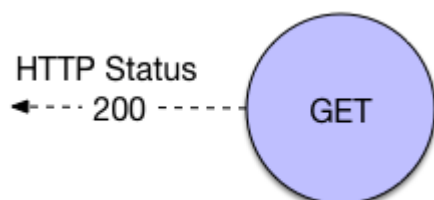
Borra

Ahora bien cada vez que realizo una de esas peticiones ¿qué debo devolver?. Recordemos

que REST es un estilo y no hay absolutos . No es lo mismo que manejar patrones de diseño que ante un problema A implica una solución B. Aquí las cosas pueden estar algo más abiertas.

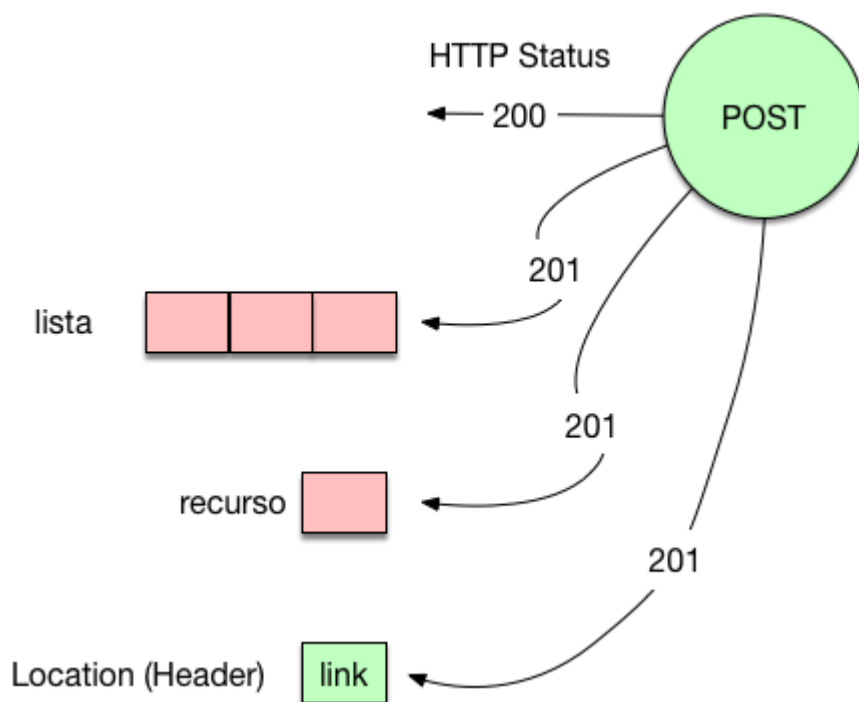
REST HTTP Return codes y GET

El primer caso es muy sencillo ante una petición GET , simplemente devolvemos un status code de 200 que es OK y el recurso o lista de recursos que necesitamos. No hay nada más que hacer.



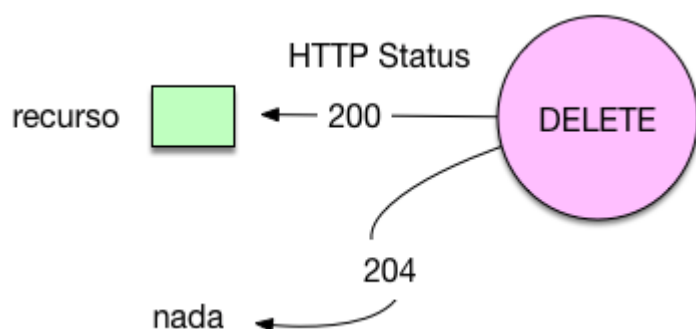
REST HTTP Return codes y POST

Este tipo de peticiones ya son mucho mas abiertas . Hay gente que opta por devolver simplemente un status code de 200 . Pero no es una gran solución, normalmente es preferible devolver un status code de 201 (recurso creado). Una vez esto está claro la pregunta que nos viene a la mente es si debemos de devolver algún tipo de contenido. Hay enfoques que devuelven la nueva lista de recursos , o devuelven el nuevo recurso creado . Son opciones posibles , pero probablemente la más flexible sea devolver una cabecera de location con la url del nuevo recurso.



REST HTTP Return codes y DELETE

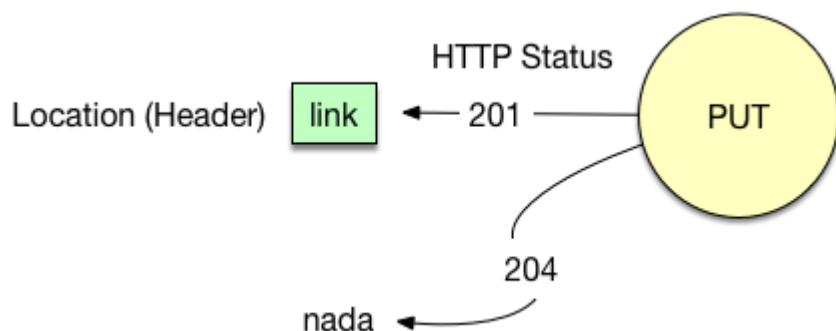
La casuística de los borrados es muy diferente a la de las inserciones pero básicamente existen dos opciones principales. Devolver un status code de 200 en el caso de que hayamos borrado y retornemos el registro borrado ya que el cliente quiera usarlo para algo. La otra opción es devolver un status code de 204 que implica que el recurso ha sido borrado pero no añadir contenido.



REST HTTP Return codes y PUT

Este caso es un poco más peculiar porque una orden de put puede hacer dos cosas o

modificar un recurso existente o si este recurso no existe puede generarlo. En el caso de que simplemente actualice el registro enviaremos un status code de 204 . Si lo que sucede es que se genera un nuevo recurso enviaremos un status code de 201 y un location header.



El diseño de arquitecturas REST siempre ha sido algo complejo.

Otros artículos relacionados

1. [Arquitecturas REST y sus niveles](#)
2. [Spring REST Client con RestTemplates](#)
3. [PostMan App con Spring REST](#)
4. [¿ Qué es REST ?](#)

Links Externos

[Introducción a REST](#)