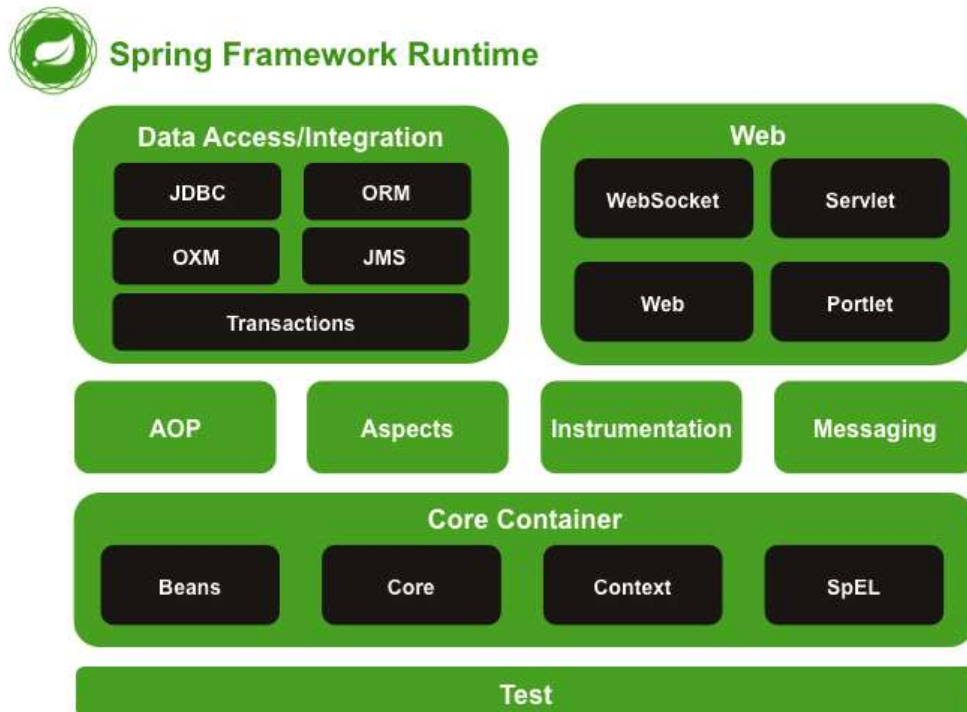


DESARROLLO WEB CON SPRING



UNIDAD 09 SERVICIOS REST

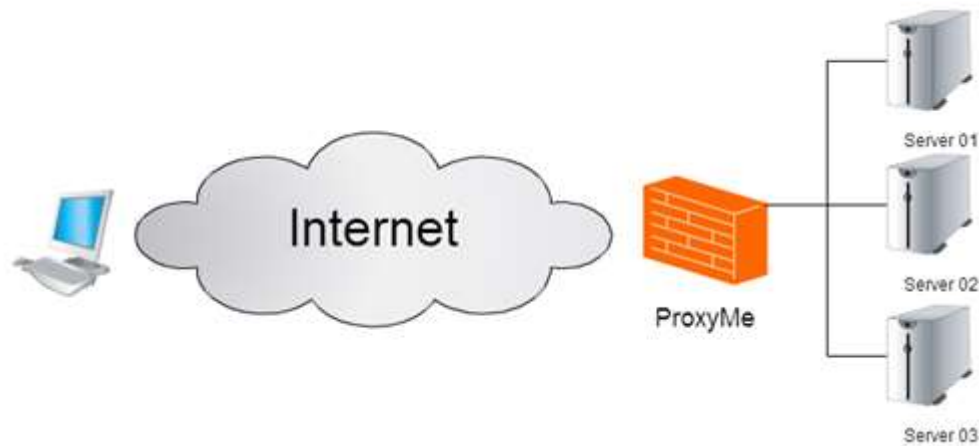
Eric Gustavo Coronel Castillo
gcoronelc.github.io
I N S T R U C T O R

CONTENIDO

INTRODUCCIÓN	3
INTRODUCCIÓN	3
¿QUÉ ES UN WEB SERVICE?	4
¿QUÉ ES API?	5
¿QUÉ ES REST?	6
¿QUÉ ES RESTFUL?	7
REPRESENTATIONAL STATE TRANSFER.....	8
FUNDAMENTOS.....	8
CARACTERÍSTICAS	9
VENTAJAS	10
POSTMAN.....	15
CURSOS VIRTUALES	16
CUPONES.....	16
FUNDAMENTOS DE PROGRAMACIÓN	16
JAVA ORIENTADO A OBJETOS	17
PROGRAMACIÓN CON JAVA JDBC	18
PROGRAMACIÓN CON ORACLE PL/SQL.....	19

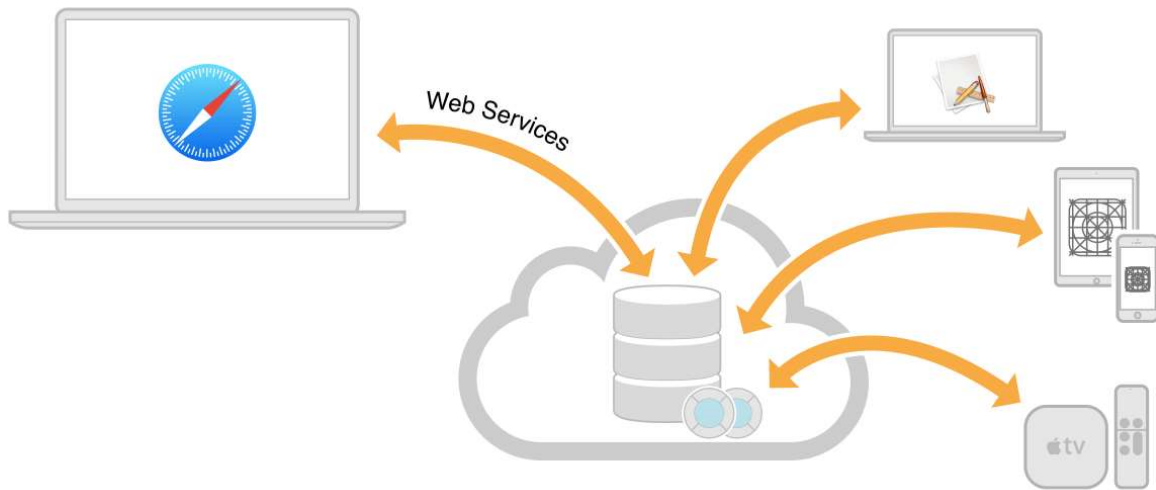
INTRODUCCIÓN

Introducción



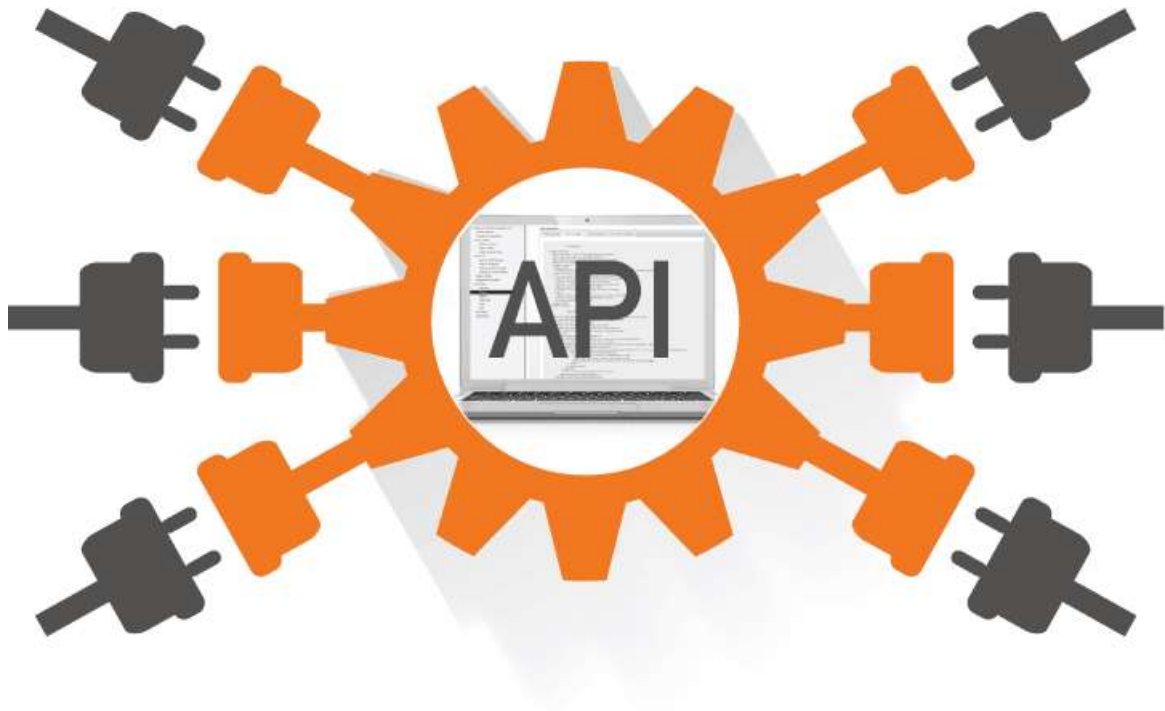
Internet es una comunicación constante entre servidor y cliente, hacer que esta comunicación sea lo más eficaz posible es el principal objetivo de las tecnologías web.

¿Qué es un Web Service?



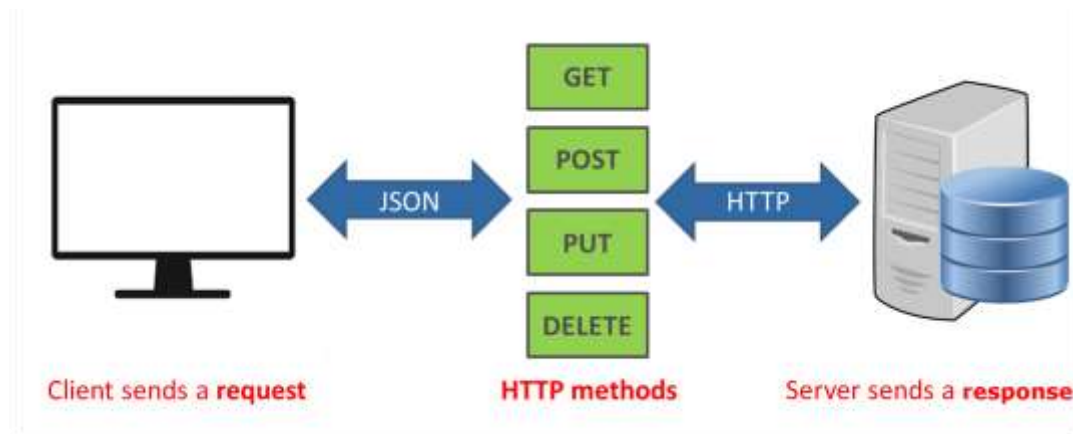
Un web service es un programa, diseñado para el intercambio de información máquina a máquina, sobre una red. Entonces esto hace que una computadora o programa pueda solicitar y recibir información de otra computadora o programa. A quien solicita la información se le llama cliente y a quien envía la información se le llama servidor.

¿Qué es API?



La palabra viene de **Application Programming Interface**, y no es más que un programa que permite que otros programas se comuniquen con un programa en específico, por ejemplo, Facebook. A diferencia de los web services, las API no necesariamente deben comunicarse entre una red, pueden usarse entre dos aplicaciones en una misma computadora.

¿Qué es REST?



REST es la abreviatura de **RE**presentational **State** **R**ransfer.

REST es una interfaz para conectar varios sistemas basados en el protocolo HTTP (uno de los protocolos más antiguos) y nos sirve para obtener y generar datos y operaciones, devolviendo esos datos en formatos muy específicos, como XML y JSON.

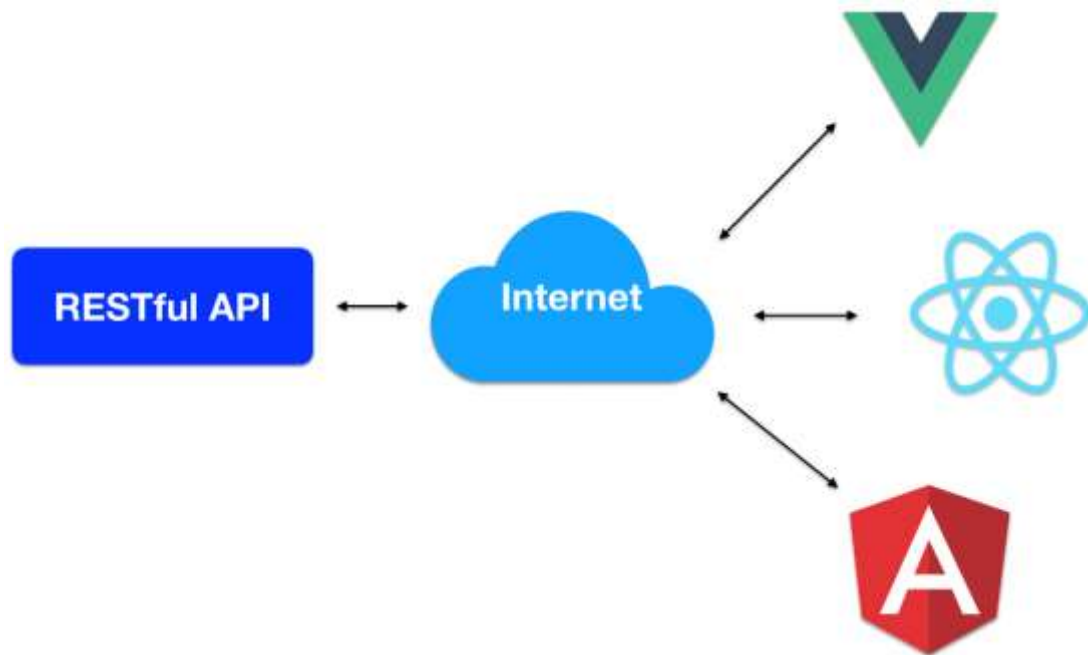
El formato más usado en la actualidad es el formato JSON, ya que es más ligero y legible en comparación al formato XML. Elegir uno será cuestión de la lógica y necesidades de cada proyecto.

REST se apoya en HTTP, los verbos que utiliza son exactamente los mismos, con ellos se puede hacer GET, POST, PUT y DELETE.

Entre sus características clave, podemos encontrar las siguientes que quizás, son las que más definen la arquitectura REST:

- **Simplicidad** de una interfaz uniforme
- **Modificabilidad** de los componentes para satisfacer las necesidades cambiantes (incluso mientras la aplicación se está ejecutando)
- **Visibilidad** de la comunicación entre componentes por agentes de servicio
- **Portabilidad** de los componentes moviendo el código del programa con los datos
- La **fiabilidad** es la resistencia al fallo en el sistema en presencia de fallos dentro de componentes, conectores o datos

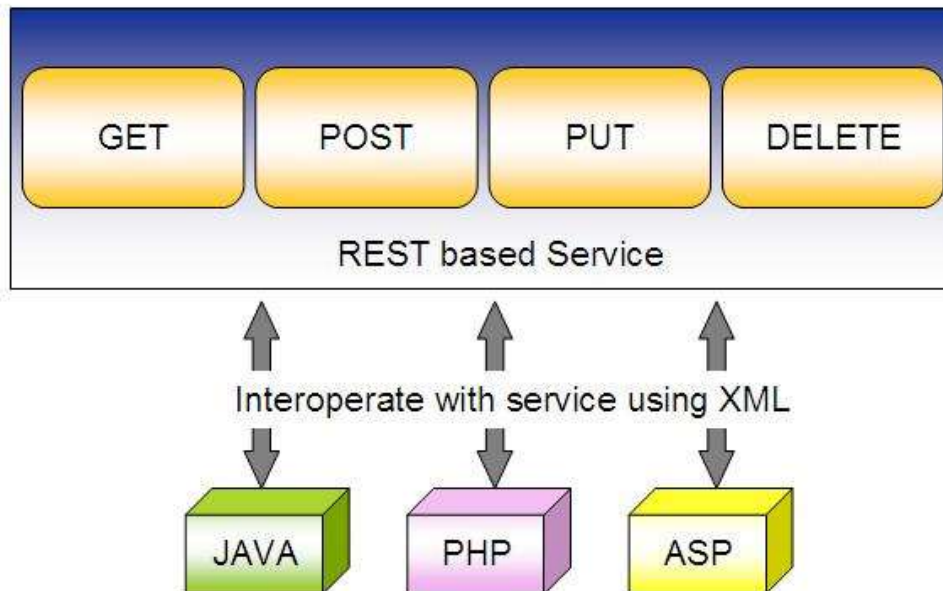
¿Qué es RESTful?



Los dos conceptos clave son necesarios ya que un servicio Web RESTful es aquél servicio web que está basado en la arquitectura REST. Los servicios Web RESTful se basan en recursos. Un recurso es una entidad, la cual se almacena principalmente en un servidor y el cliente solicita el recurso utilizando servicios Web RESTful.

RESENTATIONAL STATE TRANSFER

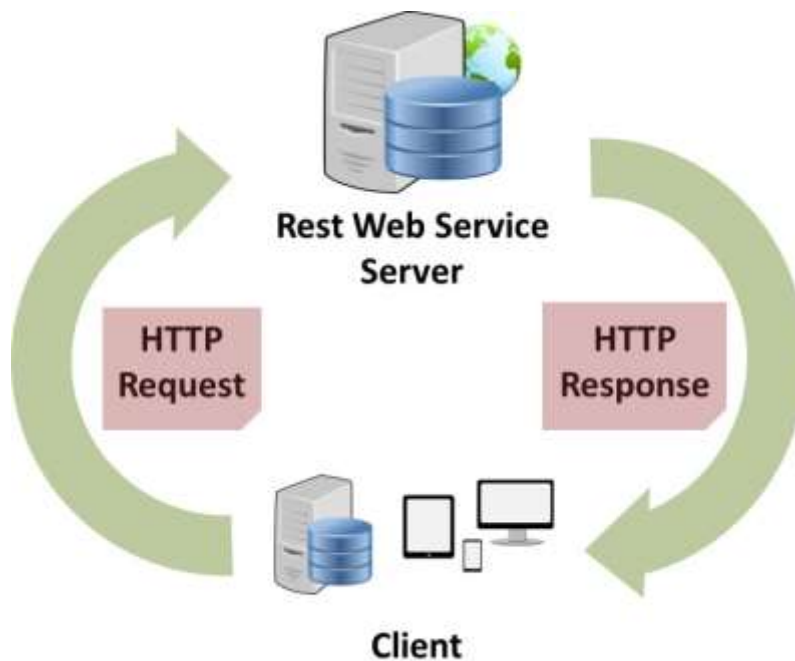
Fundamentos



REpresentational **S**tate **T**ransfer (REST) es el estilo de arquitectura de software de la World Wide Web, esto es, bajo el protocolo HTTP.

Consiste en una serie de directrices para mejorar las comunicaciones cliente-servidor a través de la web.

Características



Todo lo que transportamos a través de las comunicaciones de internet es un recurso y está representado por su formato:

- image/jpeg
- video/mpeg
- text/html
- text/xml

Cada recurso debe estar representado por un identificador único y debe ser accesible, para ello debe existir una URI única para cada recurso.

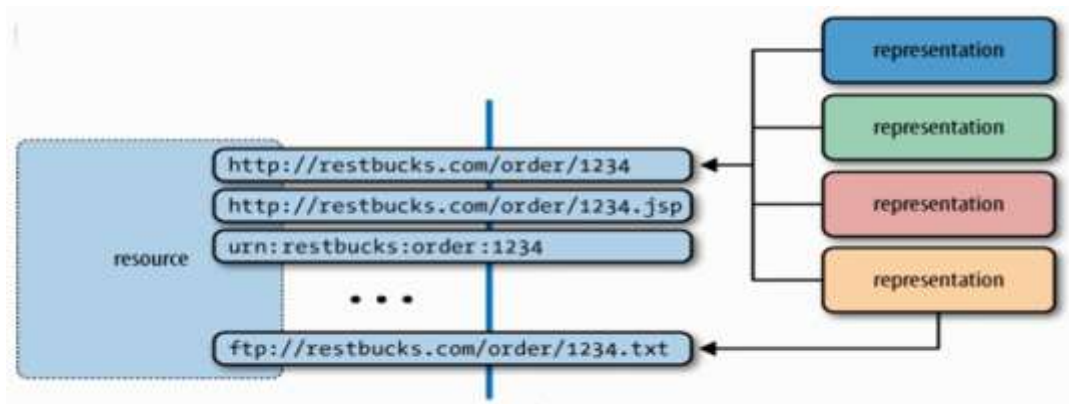
Utiliza los métodos estándares HTTP: GET, POST, DELETE, HEAD, OPTIONS, TRACE, CONNECT. Conocidos como verbos.

Los recursos pueden tener múltiples representaciones, por ejemplo, puedes solicitar los datos de un cliente en XML o JSON.

Se trata de una comunicación **SIN ESTADO**, quiere decir que cada petición es independiente, y retorna un estado final.

Ventajas

Separación del recurso y su representación. Un recurso puede ser representado en múltiples formatos.



Es la cabecera HTTP la que especifica la representación en que espera el recurso. En la siguiente imagen el recurso se solicita en formato XML.

```
GET /data/balance/22082014 HTTP/1.1
Host: nombrehost-mi-servidor
Accept: text/xml

HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: 140

<?xml version="1.0" encoding="utf-8"?>
<envio fecha="22082014">
  <Item>Ejemplo Item</Item>
    <peso medida="Kilos">23</peso>
</envio>
```

En la siguiente imagen, el mismo recurso en formato JSON.

```
GET /data/balance/22082014 HTTP/1.1
Host: nombrehost-mi-servidor
Accept: application/json
```

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 120
```

```
{
  "envio": {
    "fecha": "22082014",
    "Item": "Ejemplo Item",
    "peso": {
      "-medida": "Kilos",
      "#text": "100"
    }
  }
}
```

REST, está diseñado para ser visible y simple, quiere decir que cada aspecto del servicio debe ser auto-descriptivo, siguiendo el lenguaje natural HTTP.



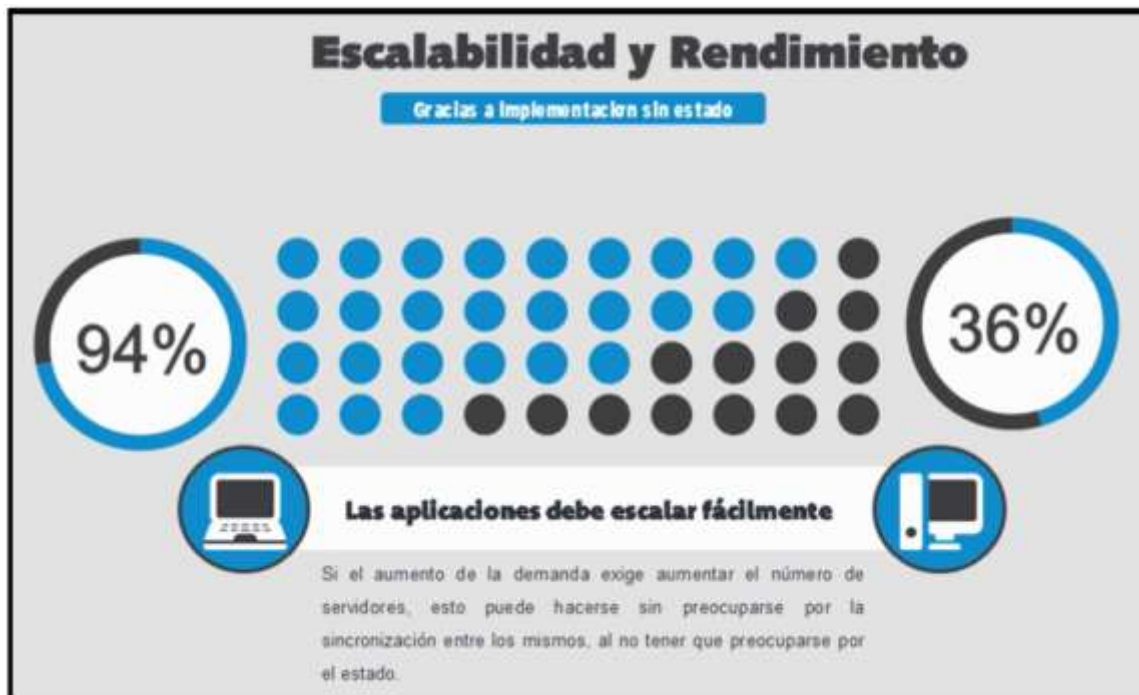
Los servicios REST son fiables.

Fiabilidad

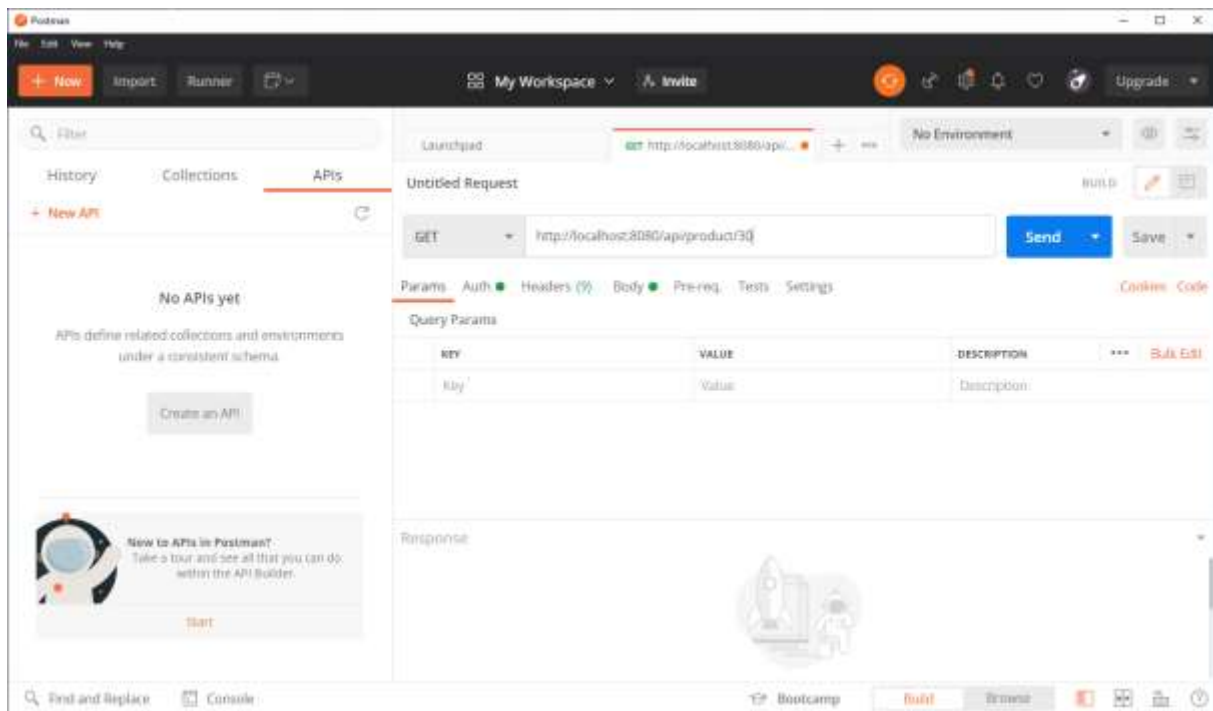


-  **REST garantiza**
-  **Métodos HTTP seguros**
no modifica o causa ningún tipo de cambio en su estado
-  **Métodos HTTP idempotentes**
lo que significa que su respuesta es siempre la misma

Los servicios REST son fáciles de escalar y mantener un rendimiento aceptable.



POSTMAN



E trata de una aplicación que permite probar servicios REST.

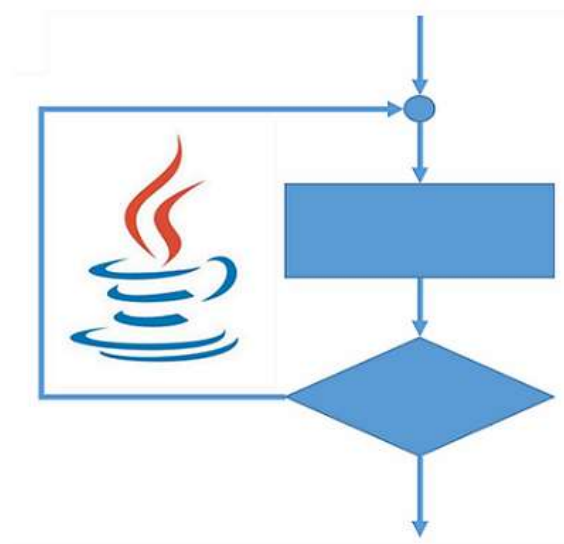
CURSOS VIRTUALES

CUPONES

En esta URL se publican cupones de descuento:

<http://gcoronelc.github.io>

FUNDAMENTOS DE PROGRAMACIÓN



Tener bases sólidas de programación muchas veces no es fácil, creo que es principalmente por que en algún momento de tu aprendizaje mezclas la entrada de datos con el proceso de los mismos, o mezclas el proceso con la salida o reporte, esto te lleva a utilizar malas prácticas de programación que luego te serán muy difíciles de superar.

En este curso aprenderás las mejores practicas de programación para que te inicies con éxito en este competitivo mundo del desarrollo de software.

URL del Curso: **<https://www.udemy.com/course/fund-java>**

Avance del curso: **<https://n9.cl/gcoronelc-fp-avance>**

Cupones de descuento: **<http://gcoronelc.github.io>**

JAVA ORIENTADO A OBJETOS



CURSO PROFESIONAL DE JAVA ORIENTADO A OBJETOS

Eric Gustavo Coronel Castillo
www.desarrollasoftware.com
I N S T R U C T O R

En este curso aprenderás a crear software aplicando la Orientación a objetos, la programación en capas, el uso de patrones de software y swing.

Cada tema está desarrollado con ejemplos que demuestran los conceptos teóricos y finalizan con un proyecto aplicativo.

URL del Curso: <https://bit.ly/2B3ixUW>

Avance del curso: <https://bit.ly/2RYGXIt>

Cupones de descuento: <http://gcoronelc.github.io>

PROGRAMACIÓN CON JAVA JDBC



PROGRAMACIÓN DE BASE DE DATOS ORACLE CON JAVA JDBC

Eric Gustavo Coronel Castillo

www.desarrollasoftware.com

I N S T R U C T O R

En este curso aprenderás a programar bases de datos Oracle con JDBC utilizando los objetos Statement, PreparedStatement, CallableStatement y a programar transacciones correctamente teniendo en cuenta su rendimiento y concurrencia.

Al final del curso se integra todo lo desarrollado en una aplicación de escritorio.

URL del Curso: <https://bit.ly/31apy0O>

Avance del curso: <https://bit.ly/2vatZOT>

Cupones de descuento: <http://gcoronelc.github.io>

PROGRAMACIÓN CON ORACLE PL/SQL

ORACLE PL/SQL



En este curso aprenderás a programar las bases de datos ORACLE con PL/SQL, de esta manera estarás aprovechando las ventajas que brinda este motor de base de datos y mejorarás el rendimiento de tus consultas, transacciones y la concurrencia.

Los procedimientos almacenados que desarrolles con PL/SQL se pueden ejecutarlos de Java, C#, PHP y otros lenguajes de programación.

URL del Curso: <https://bit.ly/2YZjfxT>

Avance del curso: <https://bit.ly/3bcqYb>

Cupones de descuento: <http://gcoronelc.github.io>