



Colecciones y APIs

Sesión Conceptual 02



- Objetivo de la sesión
- Activación de conceptos clave

- Conceptualización
- Ejercicios
- Quiz

- Cierre



Inicio



20 minutos

{desafío}
latam_

*/*Comprender que es una API*/*

Objetivo



Desarrollo



130 minutos

/* Introducción a API */

API REST

- API, acrónimo de Application Program Interface, es una interfaz de acceso que nos permite comunicar programas.
- Existen diversos tipos de APIs, Particularmente hablaremos de una en especial llamada API REST y cuando utilicemos el concepto de API de ahora en adelante nos estamos refiriendo a una API REST.

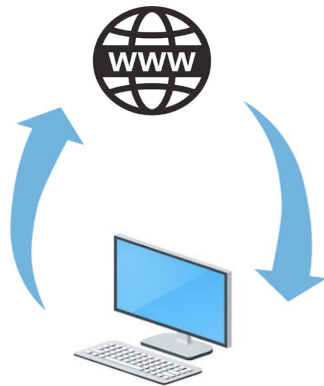
Métodos HTTP

Para indicar el tipo de operación que se llevará a cabo, existen métodos HTTP para cada caso.

- GET : Obtiene un recurso o una colección de recursos
- POST : Crear un recurso o una colección de recursos
- PUT : Actualizar un recurso o una colección de recursos
- DELETE : Elimina un recurso o una colección de recursos

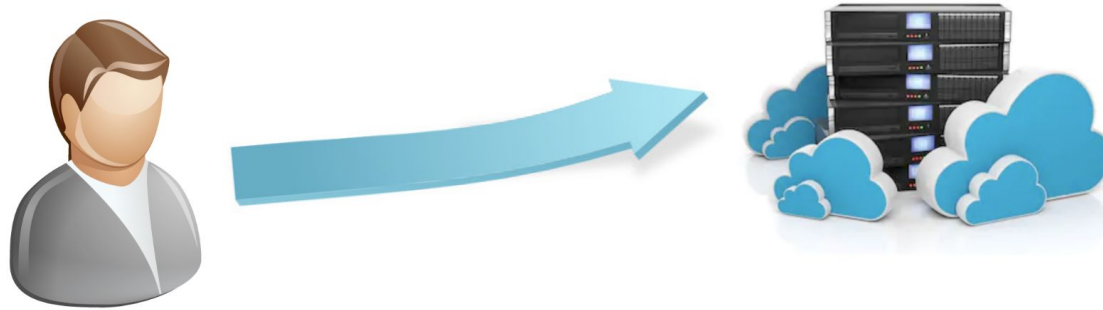
API REST

- En términos muy simplificados una API REST es un programa para otro programa que sigue una serie de restricciones para estandarizar la interconexión (interfaz).
- Es decir, es un programa fácil de ser consultado desde cualquier programa. Esto es muy útil para integrar sistemas y comunicar distintas aplicaciones.



¿Cómo usar un API?

A través de un cliente hacemos un request (solicitud) a una dirección y obtendremos como resultado un response (respuesta).



- JSON es acrónimo de Javascript Object Notation.
- Es un formato para enviar información en texto plano, fácilmente legible por humanos y fácilmente analizable por lenguajes de programación. Hoy en día es uno de los formato más utilizado para enviar información entre sistemas.

Eclipse Jersey es un framework REST que proporciona una implementación JAX-RS (JSR-370) y más.

JAX-RS: Java API for RESTful Web Service.

API del lenguaje de **Java** que proporciona soporte para la creación de servicios web de acuerdo con el estilo REST.



Quiz

{desafío}
latam_



/* Consumiendo API desde Java */

HTTP GET

Realizando GET

Para obtener un recurso:

```
Client client = ClientBuilder.newClient();  
WebTarget target = client.target("https://jsonplaceholder.typicode.com").path("posts");  
Invocation.Builder invocationBuilder = target.request(MediaType.APPLICATION_JSON);  
Response respuestaPublicaciones = invocationBuilder.get();  
List<Publicacion> listaPublicaciones = respuestaPublicaciones.readEntity(new GenericType<List<Publicacion>>() {});  
listaPublicaciones.forEach(pub -> System.out.println(pub.getTitle()+" - "+pub.getBody()));
```

HTTP POST

Realizando POST

Para crear un recurso:

```
Client client = ClientBuilder.newClient();  
WebTarget target = client.target("https://jsonplaceholder.typicode.com").path("posts");  
Invocation.Builder invocationBuilder = target.request(MediaType.APPLICATION_JSON);  
Response publicacionRespuesta = invocationBuilder.post(Entity.entity(publicacion,  
    MediaType.APPLICATION_JSON));  
System.out.println(publicacionRespuesta);
```

HTTP PUT

Realizando PUT

Para actualizar un recurso:

```
Client client = ClientBuilder.newClient();  
WebTarget target = client.target("https://jsonplaceholder.typicode.com").path("posts").path("1");  
Invocation.Builder invocationBuilder = target.request(MediaType.APPLICATION_JSON);  
Response publicacionRespuesta = invocationBuilder.put(Entity.entity(publicacion, MediaType.APPLICATION_JSON));  
System.out.println(publicacionRespuesta);
```

HTTP DELETE

Realizando DELETE

Para eliminar un recurso:

```
Client client = ClientBuilder.newClient();  
WebTarget target = client.target("https://jsonplaceholder.typicode.com").path("posts").path("101");  
Invocation.Builder invocationBuilder = target.request(MediaType.APPLICATION_JSON);  
Response publicacionRespuesta = invocationBuilder.delete();  
System.out.println(publicacionRespuesta);
```




Quiz

{desafío}
latam_





Cierre



30 minutos

**¿Existe algún concepto que no
hayas comprendido?**

**Volvamos a revisar los conceptos que más te
hayan costado antes de seguir adelante**

Reflexionemos



*Academia de
talentos digitales*

www.desafiolatam.com



/DesafioLatam



/DesafioLatam



/DesafioLatam



/DesafioLatam