

# WebServices

Consumindo WebServices em Android

# Diferenças

- SOAP:
  - Utiliza XML;
  - WSDL (Descreve a estrutura);
  - Overhead(cabeçalhos(meta-datas), serialização e XML).
- REST
  - Fléxivel utiliza JSON, XML ou texto puro;
  - Utiliza HTTP;
  - Geralmente mais leves;
  - Definição pode ficar ambígua.

# Consumindo Webservice REST

- Dificuldades:
  - Lidar com JSON grandes;
  - Controlar requisições HTTP's.
- Solução em forma de Libs:
  - Volley;
  - GSON;

# Volley

- Vantagens:
  - Controle de requisições http;
  - Dispensa uso de AsyncTask
- Desvantagens:
  - Envio de dados por JSON complicado;

# GSON

- Vantagens

- Conversão fácil e rápida de Objetos JSON para Objetos Java;
- Suporte a tipos complexos.

- Desvantagens

- Quase nenhuma, o único problema foi checagem de null quando o Objeto JSON não tem o Objeto Java esperado.

# Outras dificuldades

- Trabalhar com google API's requer que você crie uma chave no Google Developers Console;
- Trabalhar com JSON puro requer que você teste a todo momento se a tag do objeto esperado existe.

# JSON vs GSON

## Código em JSON e código em GSON

```
JSONArray livros = jsonList.getJSONArray(JSON_ITEMS);
```

```
for (int i = 0; i < livros.length(); i++) {  
    JSONObject jsonItem = livros.getJSONObject(i);  
    JSONObject jsonInfo = jsonItem.getJSONObject(JSON_VOLUME_INFO);  
    if (jsonList.has (JSON_ITEMS)) {  
        JSONArray livros = jsonList.getJSONArray(JSON_ITEMS);  
  
        for (int i = 0; i < livros.length(); i++) {  
            JSONObject jsonItem = livros.getJSONObject(i);  
            JSONObject jsonInfo = jsonItem.getJSONObject(JSON_VOLUME_INFO);  
  
            JSONObject jsonImage = null;  
            if (jsonInfo.has(JSON_IMAGE_LINKS)) {  
                jsonImage = jsonInfo.getJSONObject(JSON_IMAGE_LINKS);  
            }  
  
            String nome = jsonInfo.getString(JSON_TITLE);  
  
            String selfLink = jsonItem.getString(JSON_SELF_LINK);  
            String thumbnail = "";  
            if (jsonImage != null) {  
                thumbnail = jsonImage.getString(JSON_SMALL_THUMBNAIL);  
            }  
        }  
    }  
}
```

VS

```
Gson gson = new GsonBuilder().create();  
Library library = gson.fromJson(jsonList.toString(), Library.class);
```

# Demonstração

Vamos para o código e para o emulador!



# Conclusão

Git: [github.com/jpataide/Consumindo-WebService](https://github.com/jpataide/Consumindo-WebService)

Emails:

João Paulo - [jataide@informa.com.br](mailto:jataide@informa.com.br)

Smokoveck - [sflorencio@informa.com.br](mailto:sflorencio@informa.com.br)