

INF3005 – Programmation web avancée

Interfaces programmables

Jacques Berger

Objectifs

Introduction à JSON

Mise en place de services web

Invocation de services

Prérequis

Ajax

API

L'interface HTML est destinée à être consommée par des humains

Les logiciels consomment également des données provenant d'applications web

API

API : Application Programming Interface

Utilisation d'une interface programmable de l'application

Interagir avec l'application, au travers du code d'un autre logiciel

API

Au travers d'un API, on échange des données structurées, sans élément de présentation

Interopérabilité

Pour régler les problèmes d'interopérabilité, les données doivent être échangés dans un format standardisé

Un format texte, avec un encodage de caractères connu

Exemples : XML, JSON

JSON

JavaScript Object Notation

Sous-ensemble du langage Javascript

JSON

Format texte

Interopérabilité

Simple et léger

Lisible par une machine et un humain

JSON

Extension habituelle : .json

MIME Type : application/json

Le MIME Type sert à spécifier le type de données qu'on envoie dans une requête HTTP

JSON

```
{  
  "type": "book",  
  "title": "Théorie de la musique",  
  "author": [  
    "A. Danhauser"  
  ],  
  "year": 1996  
}
```

Syntaxe

Aucun mot-clé réservé

Sensible à la casse

Concepts

Objets

Tableaux

Objet

Un objet JSON est délimité par des accolades {}

Il contient des couples propriété/valeur

Le nom de la propriété est une chaîne de caractères

Similaire à un dictionnaire en Python

Objet

La valeur peut être :

Une chaîne de caractères

Un nombre

true

false

null

Un objet

Un tableau

Objet

La valeur est séparée de la propriété par un ":"

```
"type": "book"
```

Les couples sont séparés par une virgule

```
{  
  "type": "book",  
  "title": "Théorie de la musique"  
}
```

Objet

```
{  
  "type": "book",  
  "title": "Test-Driven Development by Example",  
  "author": "Kent Beck",  
  "year": 2003  
}
```


Tableau

Array

Exactement comme un tableau en Javascript

Liste de valeurs ordonnées, sans nom

Le tableau est délimité par des crochets []

Les valeurs sont séparées par une virgule

Tableau

```
[  
  "Erich Gamma",  
  "Richard Helm",  
  "Ralph Johnson",  
  "John Vlissides"  
]
```

Exemple complet

Voir exemple

Encodage des données

Un document JSON est encodé en utf-8

Vérification syntaxique

Vérifier la syntaxe un document JSON pour s'assurer de sa conformité est toujours une bonne idée

Outil en ligne : JSONLint
<http://jsonlint.com/>

Services web

L'API d'une application est un ensemble de services web

Un service web offre une fonctionnalité à distance

- Obtenir des données

- Créer une ressource

- Modifier une ressource

- Etc.

Services web

Dans une application web, une route offrant une interface programmable sur une fonctionnalité est un service web

L'API est l'ensemble des services web de l'application

Services web

L'architecture privilégiée actuellement pour la conception de services web est REST

On parle de RESTful web services

REST

REST est une architecture simple mais elle inclut beaucoup plus de concepts qui ne seront pas présentés dans ce cours

REST

Dans sa forme la plus courante, un service REST est un service web qui échange des données en JSON

La sémantique des méthodes HTTP doit être appliquée à la lettre et les routes suivent certaines conventions

REST

Exemples de routes :

Lire une liste de livres

GET + /api/livres

Lire un livre

GET + /api/livre/<id_livre>

REST

Créer un livre

POST + /api/livre

Modifier un livre

PUT + /api/livre/<id_livre>

Supprimer un livre

DELETE + /api/livre/<id_livre>

Flask

Une route Flask peut envoyer du JSON grâce à la fonction `jsonify`

Définit automatiquement le MIME Type de la réponse HTTP

Flask

Voir exemples

Client

Pour tester des services REST, il est utile d'utiliser un client REST intégré à votre navigateur

Voir exemple

Ajax

Il est courant d'utiliser des services REST pour des appels Ajax

Voir exemples

Javascript

En Javascript, on transforme une chaîne de caractères contenant du JSON en objet Javascript avec `JSON.parse()`

On transforme un objet Javascript en chaîne de caractères contenant du JSON avec `JSON.stringify()`

Documentation

Les services REST seront invoqués par des développeurs

Une bonne documentation est importante pour chaque service

Documentation

La documentation doit contenir :

- L'URL du service

- Les paramètres de l'URL

- Le format des données reçues

- Le format des données à envoyer

- Les codes de statut possibles

Plus loin...

JSON

<http://www.json.org/json-fr.html>

JSONLint

<http://jsonlint.com/>