

Réseaux et Programmation

Introduction aux services Web

Les services Web s'appuient sur des commandes http GET et POST comme nous allons le voir.

Pour faciliter les tests nous allons utiliser des formulaires comme clients.

1 Les formulaires pour faciliter le passage de paramètres dans l'URL

Jusqu'à là les paramètres sont fournis à la main, dans le bon ordre, dans la syntaxe même de l'URL demandée.

Un formulaire HTML de positionner dans une page Web des zones interactives qui permettent à la fois la saisie des paramètres mais aussi la création de l'URL demandée.

Exemple :

```
<form action="http://www.example.com/cgi-bin/example.sh" method="get">  
  Enter a username: <input type="text" name="username"></input><br>  
  <input type="radio" name="whatToDo" value="remove" checked="checked">Value<br>  
  <input type="radio" name="whatToDo" value="create">Create<br>  
  <input type="submit" name="subbtn" value="Submit">  
</form>
```

Les différents types d'éléments existant pour une balise `<input>` sont : `buttoncheckbox`, `file`, `hidden`, `image`, `password`, `radio`, `reset`, `submit`, `text`.

Le type `submit` effectuera à la fois l'affectation de la valeur "Submit" au paramètre `submit` mais déclenchera aussi la requête sur l'URL créée et paramétrée.

QUESTION 1 : Rajouter la commande POST dans les commandes traités par le serveur WEB que vous avez développé au TD précédent et tester la avec un formulaire.

Remarque : vous pouvez trouver plus d'explications sur la création d'un formulaire sur la page http://www.w3schools.com/html/html_forms.asp.

2 Du Web au Services Web

2.1 Introduction

Comme nous l'avons vu en cours les Services Web se caractérisent par :

- Une adresse (URL)
- Un protocole de transport et un protocole/format de codage des informations échangées (Binding)
- Une API qui correspond à l'ensemble des requêtes que vous pouvez faire sur le service Web (Contrat)

Réseaux et Programmation

Introduction aux services Web

Pour les services web de type REST, les commandes HTTP utilisées sont GET, POST, PUT, DELETE. Le format des données échangées n'est pas imposé. Il s'agit bien souvent d'un format JSON, XML, ... pour les faciliter de parsing qui sont offertes par les plateformes pour ces formats.

Pour les services web de type SOAP (WS-SOAP), la commande HTTP utilisée est GET. Le format des données échangées est SOAP. Dans ce cas le protocole HTTP est juste une couche de transport supplémentaire et SOAP le format de sérialisation des messages échangés pour des invocations de méthodes à distance.

2.2 Clients de services web

Parmi de nombreux fournisseurs de Services Web, vous trouverez sur www.dataaccess.com et en particulier le service « numberToWords »

<http://www.dataaccess.com/webservicesserver/numberconversion.wso?op=NumberToWords>

En visualisant la page vous comprendrez que ce service est accessible de deux manières : comme un service web REST et comme un WS-SOAP. Vous trouverez de même un formulaire pour tester le service.

QUESTION 1 : Ecrivez votre propre formulaire pour tester ce service Web en mode REST

QUESTION 2 : Ecrivez un client en python qui lira au clavier un nombre et affichera la chaîne correspondante retournée par le service. L'invocation du service se fera en mode REST puis en mode SOAP en s'inspirant des exemples de la page Web si dessus.

2.3 Fournisseur de service web

Nous allons étudier maintenant les bases de la conception d'un service Web.

QUESTION 1 : En reprenant le serveur Web du TD précédent, que devez-vous rajouter pour implémenter un service Web REST qui implémente une méthode « incr », qui récupère un nombre et retourne son incrément ?

QUESTION 2 : Implémentez ce service web et testez-le avec un formulaire ou un client python. Quel format de donnée avez-vous choisi ?

QUESTION 3 : Que devrions-nous modifier pour implémenter ce service en mode WS-SOAP ?