Licence Ingénierie Logicielle et Système d'Information

TP N°5 SERVEUR HTTP ET PAGE WEB

Exercice N°1 :Page web statique

1. **Créer un serveur web http « server.py »en utilisant le code suivant :**

```python
import http.server
import socketserver
#le port et addresse
port=80
address=("",port)
#gestionnaire
handler=http.server.SimpleHTTPRequestHandler
httpd=socketserver.TCPServer(address,handler)
print(f"serveur demaré sur le port {port}")
httpd.serve_forever()
```

2. **Créer une page HTML « index.html » qui affiche un message de connexion « serveur web bien connecté en utilisant le code suivant**

```html
<HTML>
<HEAD><TITLE>page html en python</TITLE></HEAD>
<BODY>
    <H1>Bienvenue</H2>
    <P>Serveur connecté</P>
    </DIV></BODY></HTML>
```

3. **Démarrer le serveur web** `python -m http.server --cgi 8000`

Exercice N°2 : Page Web Dynamique

En schématisant on peut considérer que l'accès à une page web depuis une machine s'effectue selon une architecture client/serveur. La machine client émet une requête de page web et la machine serveur lui répond en lui envoyant cette page. Pour les pages web les machines dialoguent selon le protocole HTPP. La page web retournée par le serveur peut être statique : c'est un fichier d'extension .html qui se trouve dans

l'arborescence du serveur et contenant du code HTML/CSS qui sera interprété par le navigateur du client pour afficher la page. De nos jours, les sites contiennent souvent des pages web dynamiques dont le contenu peut varier selon le client qui la demande. Dans ce cas le serveur exécute d'abord un script qui récupère les variables envoyées par le client (à travers un formulaire ou un cookie par exemple) puis génère du code HTML/CSS. Le serveur retourne la page web au client. Ce dernier reçoit le code HTML/CSS mais pas le code du script générateur. Les scripts qui servent à générer des pages web sur le serveur sont généralement écrits en langage PHP. C'est la technique la plus rapide, mais l'interface de programmation CGI (Common Gateway Interface) permet d'utiliser n'importe quel langage de script dont l'interpréteur est installé sur le serveur. Dans cet exercice les scripts seront écrits en langage python.

Dans cet exercice nous allons écrire trois scripts Python :

1. **Un serveur web dynamique serverhttpcgi.py**
2. **Une page HTML « index1.py » peut contenir un formulaire avec différents champs (Nom, prénom). Lorsque le client a rempli tous les champs, il clique sur un bouton envoyer et les données sont transmises au serveur.**
3. **result.py pour générer une page web qui affichera le résultat du traitement du formulaire : nom, prénom**

- **Pour écrire un formulaire en HTML on utilise une balise <form> où l'on va placer des balises de champ telles que <input> ou <select> voici un exemple index1.py qui va générer le formulaire.**

Index1.py

```
import cgi
print("content type: texte/html")
print("<h1>bienvenur<h1>")
html="""<!Doctype html>

<HEAD><TITLE>page html en python</TITLE></HEAD>
 <BODY>
    <H1>Inscription</H2>
    <form methode="post" action="result.py">
     <p>Prénom :<input type="text" name="prenom"></p>
     <p>Nom :<input type="text" name="nom"></p>
     <input type="submit" value="envoyer"></p>
     </form>

    </DIV></BODY>
```

```
</HTML>
"""

print(html)
```

Quelques explications :

- La balise Haut du formulaire<form> comprend deux attributs obligatoires method="post" pour préciser la méthode d'envoi et action="result.py" pour la référence au script qui traitera le formulaire.
- Il existe deux méthodes d'envoi : "post" la plus courante et "get" la plus ancienne qui transmet les données dans l'URL sous la forme d'une séquence de couples variable=valeur placés après un ? et séparés par un & comme ci-dessous : http://localhost/python/result.py?nom=xxxxx&prenom=yyyyy
- Un champ de saisie comme :<input type="text" name="prenom"></p> permet de saisir du texte qui sera envoyé au script de traitement comme valeur de la variable prenom. L'attribut id est facultatif, il permet de référencer la balise dans le code HTML, il peut servir de cible pour un fichier de style en CSS
- Il existe d'autres types de champs de saisie <input> comme "textarea" pour des textes longs, "password" pour les mots de passe, "file" pour l'envoi de fichiers, "checkbox" pour les cases à cocher (plusieurs choix possibles) ou "radio" pour les boutons de radio (un seul choix possible).
- Enfin pour envoyer les données du formulaire il faut insérer un bouton d'envoi avec un champ<input type= 'submit' value = 'valider'/ > la valeur de l'attribut value étant le texte affiché dans le bouton

- Un script qui traite le formulaire précédent

Le script traitement.py ci-après réceptionne le formulaire envoyé par la page web générée par index1.py et génère une page web en réponse. Les données sont récupérées dans un objet de la classe cgi.FieldStorage() du module CGI.
On assigne cet objet à une variable form avec form = cgi.FieldStorage()
Les champs du formulaire peuvent être récupérés par le biais de leur attribut name : par exemple le champ d'attribut name="prenom" est stocké dans l'objet form['prenom']. Sa valeur est accessible par form['prenom'].value.
La fonction cgitb.enable() du module cgitb (pour cgi trace back), qui permet, en phase de développement, l'affichage des messages d'erreurs Python dans le navigateur (sinon ce dernier n'affiche que les messages d'erreurs du serveur Apache et ceux-ci ne nous disent rien des erreurs internes au code Python).

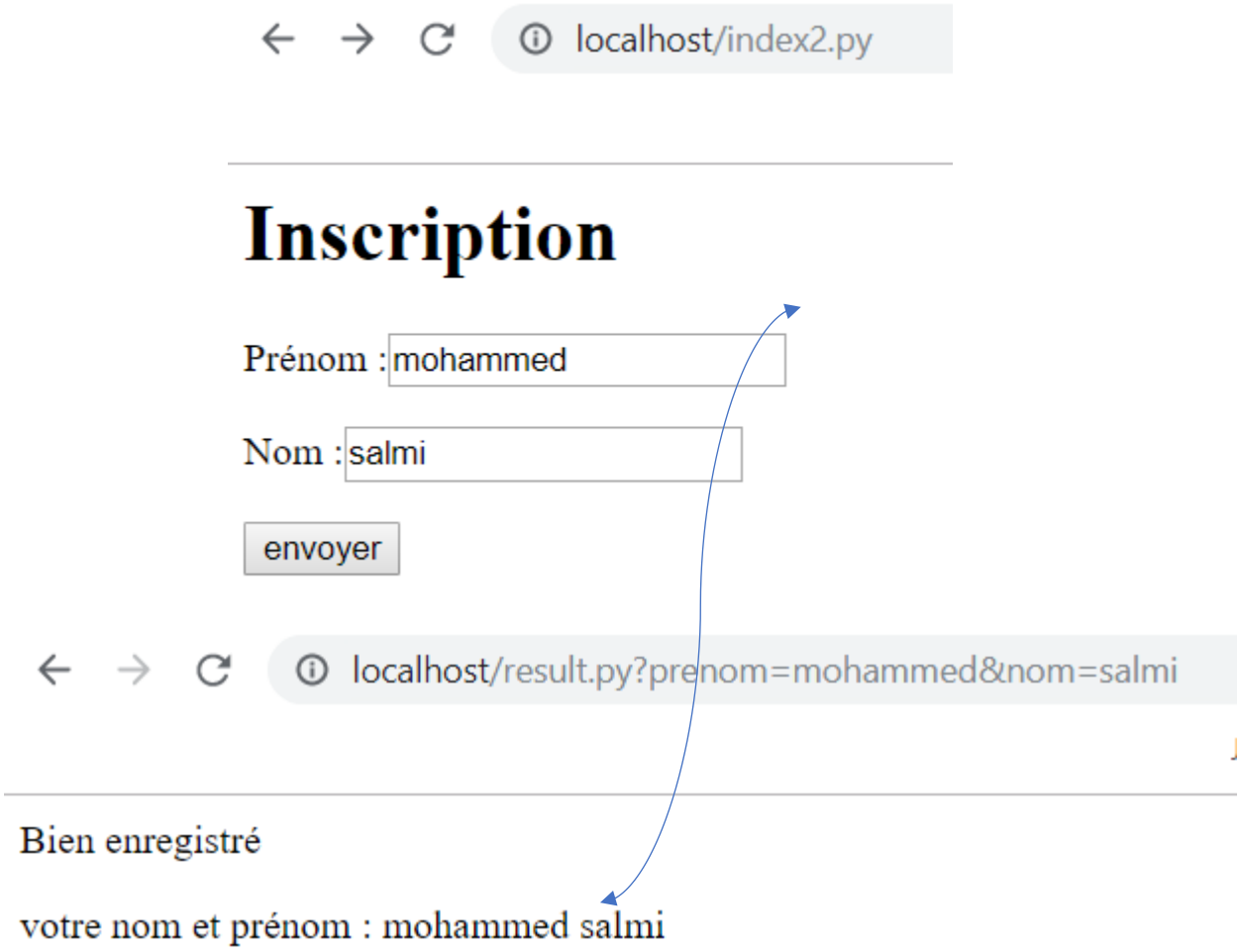
Result.py

```
#Common Gateway Interface support
```

```
import cgi
#gestionnaire d'exceptions pour les scripts CGI
import cgitb
cgitb.enable()
form = cgi.FieldStorage()
if form.getvalue("prenom")!= None :
    prenom = form.getvalue("prenom")
    nom= form.getvalue("nom")
else:
    raise Exception("pseado non transmit")
print("content type: texte/html")
print("<h1>bienvenur<h1>")
html="""<!Doctype html>

<HEAD><TITLE>Resultat</TITLE></HEAD>
 <BODY>
    <P>Bien enregistré</P>
    </DIV></BODY>
</HTML>
"""
print(html)
print(f"votre nom et prénom :  {prenom} {nom}")
```

Exemple :


localhost/index2.py
Inscription
Prénom : mohammed
Nom : salmi
envoyer
localhost/result.py?prenom=mohammed&nom=salmi
Bien enregistré
votre nom et prénom : mohammed salmi