

AU : 2020-2021 Niveau : L2	TD 2 PHP String	Enseignants : Sabrina Edded Fabrice Boissier
---	----------------------------------	---

Exercice 1 : URL (2)

- Indiquer le protocole de communication, le domaine complet, le port de connexion, puis la ressource et son chemin d'accès pour ces URL :
 - `ssh://pouletsdu49.anjou.fr:8080/ressource/cotcot.codac`
 - `http://avenue42.fr:3864/1337/8912.php`
- Construire l'URL pour les ressources suivantes :
 - Protocole : *web sécurisé*
Port : *747*
Sous-domaine : *pool08.datalake*
Domaine : *mystore*
TLD : *com*
Chemin : *Ressource/Report*
Fichier : *cour.des.comptes.2020*
 - Fichier : *fichier*
Domaine : *mamie*
Protocole : *web*
Sous-domaine : *tata.y0y0*
Chemin : *annuaire*
Port : *9999*
TLD : *fr*

Exercice 2 : Chaînes de caractères/Strings

Dans cet exercice, vous apprendrez succinctement à manipuler des chaînes de caractères. Les fonctions les plus utiles sont : **strlen**, **substr**, l'opérateur de concaténation « . » (le point), et l'opérateur d'accès à un caractère « [] » (les crochets).

- Écrire une fonction « *phrase()* » qui renvoie la chaîne de caractères :
Bonjour, aujourd'hui il fait plutôt beau.
- Testez les fonctions de base sur la chaîne précédente :
 - Utilisez *strlen* et affichez la taille de la chaîne de caractères
 - Utilisez *substr* pour afficher les 30 premiers caractères d'une string
 - Utilisez *substr* pour afficher les derniers caractères d'une string à partir du caractère 37
- Écrire une fonction « *remplace1(str1, str2)* » qui remplace les caractères 30 à 37 de la string *str1* par la string *str2*.
(Faites au plus simple : renvoyez une chaîne de caractère issue de la concaténation de plusieurs autres)

4. Testez les fonctions de base sur la chaîne précédente :
 - a. Utilisez *strlen* et affichez la taille de la chaîne de caractères
 - b. Dans une boucle *for*, affichez un caractère par ligne
 - c. Dans une boucle *for*, affichez chaque caractère et ne revenez à la ligne « que » lorsque le caractère courant est un(e) espace (que vous n'afficherez pas)
5. Écrire une fonction « *decoupeEspace(str)* » qui renvoie un tableau contenant dans chaque cellule un mot de la chaîne de caractères passée en paramètre (un mot est séparé d'un autre par un(e) espace)
6. Écrire une fonction « *my_strstr(str1, str2)* » qui renvoie un tableau contenant dans chaque cellule les mots de la chaîne *str1* séparés par n'importe quel caractère de la chaîne *str2*
7. Écrire une fonction « *remplace2(str1, nb, str2)* » qui remplace le *nb*-ième mot de la chaîne *str1* par le mot *str2*. Les mots de *str1* sont séparés par des espaces, et *str2* doit être intégralement copiée.
8. Utilisez les meilleures fonctions pour rendre la journée bien meilleure !
[C'est-à-dire, remplacez le « *plutôt* » par « *très* » dans la phrase de la partie 1]

PS : Il y a plusieurs façons de faire.

- a. Soit la version hyper simple : on donne en dur les paramètres, et la fonction ne gèrera QUE cette phrase.
- b. *[bonus à points]* Soit la version plus complexe où vous codez des fonctions résolvant plusieurs règles/contraintes :
 - deux fonctions qui cherchent le positionnement de début et de fin d'un mot dans une chaîne de caractère,
 - puis une fonction qui remplace une section de chaîne (caractère début, caractère fin) par un autre motSi vous codez ces fonctions et les utilisez uniquement en une seule ligne, vous aurez quasiment réalisé une approche *programmation fonctionnelle* au problème.
(en une seule ligne comme cet exemple : *fun1(fun2(), fun3(fun4()));*)