

Série d'exercices 3 : conditions

Exercice 1 - Compérateur

Ecrivez un programme qui demande deux nombres à l'utilisateur et affiche True si le premier nombre est **plus grand ou égal** au deuxième nombre, sinon le programme affiche False. Voici un exemple d'exécution du programme :

```
> Donnez un premier nombre : 19
> Donnez un deuxième nombre : 8
> True
```

Pour vérifier votre solution nommez votre script python `exercice1.py` et suivez les instructions à la fin de ce document.

Exercice 2 – Quiz simple

On veut écrire un quiz simple d'une seule question qui demande à l'utilisateur '*En quelle année Android a été commercialisé ?*'. La réponse est 2008. Cette réponse devra afficher '*Juste!*', toute autre réponse devra afficher '*Faux.*'.

Pour vérifier votre solution nommez votre script python `exercice2.py` et suivez les instructions à la fin de ce document.

Exercice 3 – Quiz avec indice

Cette fois le quiz posera la question '*En quelle année ASCII a été créé?*'. La réponse est 1963, si cette réponse est donnée le programme devra toujours afficher '*Juste!*'. En revanche, si l'utilisateur donne une autre année, le programme devra calculer la différence avec 1963 et la donner en indice. Par exemple si l'utilisateur entre l'année 1973, le programme devra afficher '*Faux. Vous êtes à 10 années de la réponse correcte.*'.

Pour vérifier votre solution nommez votre script python `exercice3.py` et suivez les instructions à la fin de ce document.

Exercice 4 – Quiz avec plusieurs réponses possibles

Cette fois la question à poser est '*Quel encodage de texte est le plus courant de nos jours?*'. Et cette fois la réponse n'est pas un nombre mais une chaîne de caractères. On aimerait compter comme juste 4 versions différentes de la même réponse correcte : '*UTF8*', '*UTF-8*', '*utf8*' et '*utf-8*'.

Chacune de ces réponses devra afficher '*Juste!*', toute autre réponse devra afficher '*Faux*'.

Pour vérifier votre solution nommez votre script python `exercice4.py` et suivez les instructions à la fin de ce document.

Exercice 5 – Login

On veut écrire le code d'un login administrateur pour un site Web. Votre programme doit demander à l'utilisateur **son nom**, puis **son mot de passe**. Le nom d'utilisateur correct est « admin » et le mot de passe correct « 9501 ». Si l'utilisateur entre correctement ces deux éléments, votre programme devra afficher « *Vous êtes connecté en tant qu'admin.* ». Toute autre combinaison devra afficher « *Nom d'utilisateur ou mot de passe incorrect.* ».

Exemple d'exécution :

> Nom d'utilisateur : blabla@eduvaud.ch

> Mot de passe : 9501

> Nom d'utilisateur ou mot de passe incorrect.

Pour vérifier votre solution nommez votre script python `exercice5.py` et suivez les instructions à la fin de ce document.

Exercice 6 – calculateur de prix de parking

On veut écrire le code d'un automate de paiement de stationnement de parking. Le programme prendra le nombre de minutes que l'utilisateur a passé sur le parking en demandant '*Nombre de minutes de stationnement :*'.

Puis il affichera le coût en francs sous la forme '*x CHF*'. En dessous de 15 minutes, le parking est gratuit. A partir de 15 minutes mais moins d'une heure, le prix est de 5 centimes par minute. A partir d'une heure, le prix est de 5 francs par heure. On ne compte 1 heure que si elle est pleine, donc 2h50 passée sur le parking compte comme 2 heures, pas 3 heures.

Pour vérifier votre solution nommez votre script python `exercice6.py` et suivez les instructions à la fin de ce document.

Exercice 7 (Bonus) – autorisation et paiement en ligne

On veut coder un formulaire en ligne qui donne un service payant seulement à des utilisateurs majeurs ou munis d'une autorisation parentale. Il demandera d'abord son âge à l'utilisateur qui répondra par un nombre d'années. Puis s'il est mineur, le programme demandera si l'utilisateur possède une autorisation parentale. Les réponses '*oui*' et '*Oui*' seront considérées comme les seules valides. Toute autre réponse devra afficher '*Pas autorisé. Doit être majeur ou avoir une autorisation parentale.*' et terminer le programme. Si l'utilisateur est majeur, cette question ne sera pas posée.

Après cette vérification, le programme demandera combien d'argent l'utilisateur possède. Le coût du service étant de 30 CHF, si le montant entré par l'utilisateur dépasse 30 CHF on affichera le montant restant ainsi : '*Montant restant: 3*' dans le cas où l'utilisateur avait 33 CHF. Si le montant est inférieur à 30 CHF, le programme affichera '*Montant insuffisant: 17*' si l'utilisateur possède 17 CHF.

Pour vérifier votre solution nommez votre script python `exercice7.py` et suivez les instructions à la fin de ce document.

Vérification

Vérifiez vos fichiers en les plaçant sur le **Bureau** et en téléchargeant le fichier `python_checker_serie3.sh` depuis Moodle et en le copiant sur le **Bureau**. Ouvrez ensuite le **terminal** (touches espace + commande et tapez « ter » pour voir apparaître l'application terminal), entrez les 2 commandes suivantes :

```
cd ~/Desktop
```

```
sh python_checker_serie3.sh
```

Pour chaque exercice, le terminal devrait afficher « *exercice est correct !* ». Toute erreur sera affichée avec un message en *rouge*.