L1-MPCE2I L1-BBTE L1-ECO-MATHS	Informatique - Travaux Pratiques	TP3

Objectifs du TP

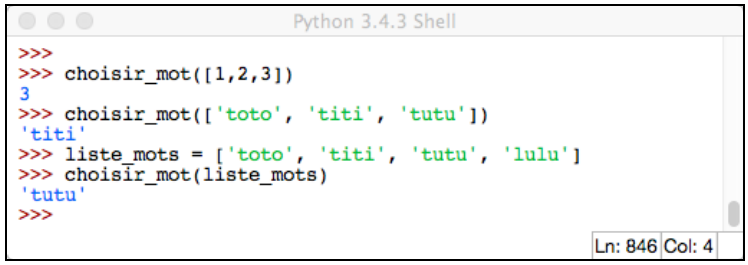
- Les fonctions (oui encore !)
- La boucle *while*
- La boucle *for*
- Les types liste et chaine

Problème : le jeu du pendu

Le but du jeu du Pendu est de trouver un mot tiré au hasard par l'ordinateur parmi une liste de mots. Le joueur doit ensuite dévoiler les lettres cachées en proposant une lettre. Si le joueur propose une lettre existante dans le mot, l'ordinateur dévoile tous les emplacements du mot où se trouve cette lettre. Le joueur a gagné s'il trouve le mot complet en x tentatives au plus, x étant égal à deux fois la longueur du mot à trouver.

Partie 1 : tirage au sort d'un mot

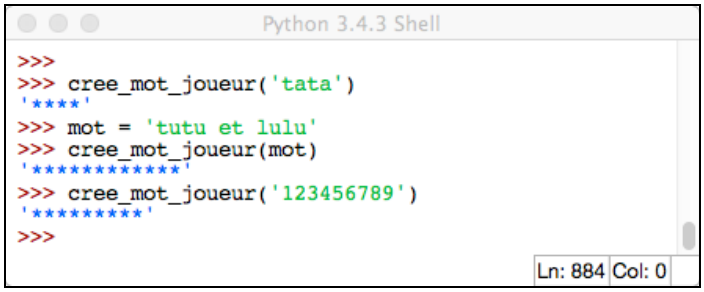
Ecrivez une fonction *choisir_mot(liste)* qui tire au hasard un élément de la liste et le retourne comme dans l'exemple de l'image suivante :



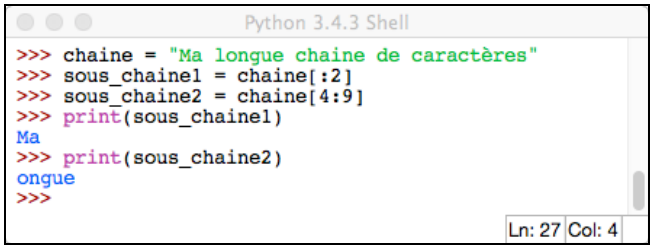
Vous utiliserez la fonction *choice(liste)* du module *random* qui permet de choisir au hasard un élément d'une liste. La fonction *choice* retourne l'élément choisi au hasard.

Partie 2 : générer le mot du joueur

Ecrivez une fonction *cree_mot_joueur(chaine)* qui retourne une nouvelle chaine de même longueur que l'argument *chaine* mais dont les lettres sont remplacés par le caractère '*' comme dans l'exemple de l'image suivante :

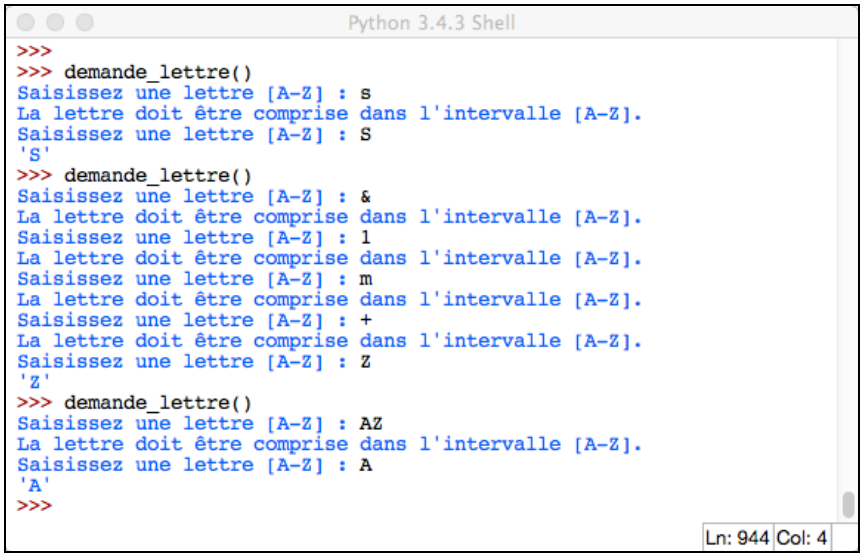


Remarque : vous pourrez utiliser la concaténation de chaîne et l'accès à des sous-chaînes :



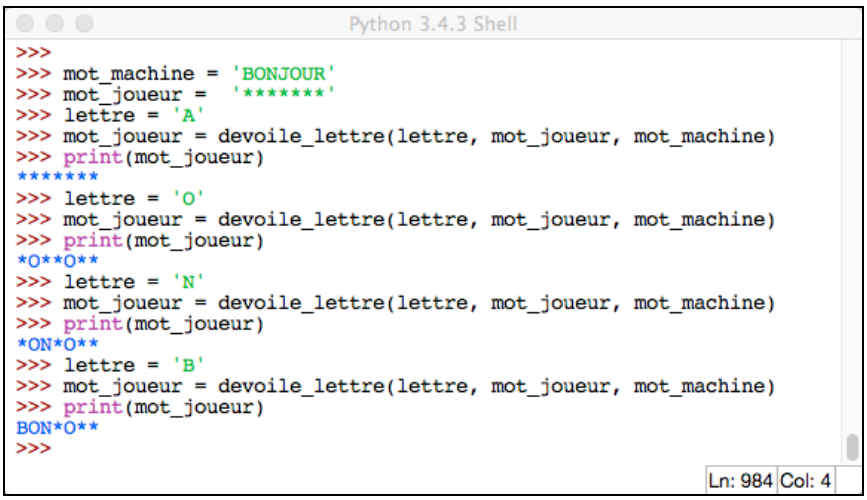
Partie 3 : saisir la lettre proposée par le joueur

Ecrivez une fonction *demande_lettre()* qui retourne la lettre choisi par l'utilisateur parmi l'ensemble des lettres possibles. La fonction continue tant que la saisie n'est pas correcte : lettre appartenant à l'intervalle et de longueur une comme dans l'exemple de l'image suivante :



Partie 4 : dévoiler les lettres trouvées

Ecrivez une fonction *devoile_lettre(lettre, mot_joueur, mot_machine)* qui retourne une chaine dans laquelle la *lettre* donnée en argument est dévoilée si elle existe dans *mot_machine*. Bien sûr, les autres lettres déjà dévoilées dans *mot_joueur* restent dévoilées. Voyez son fonctionnement dans l'exemple de l'image suivante :

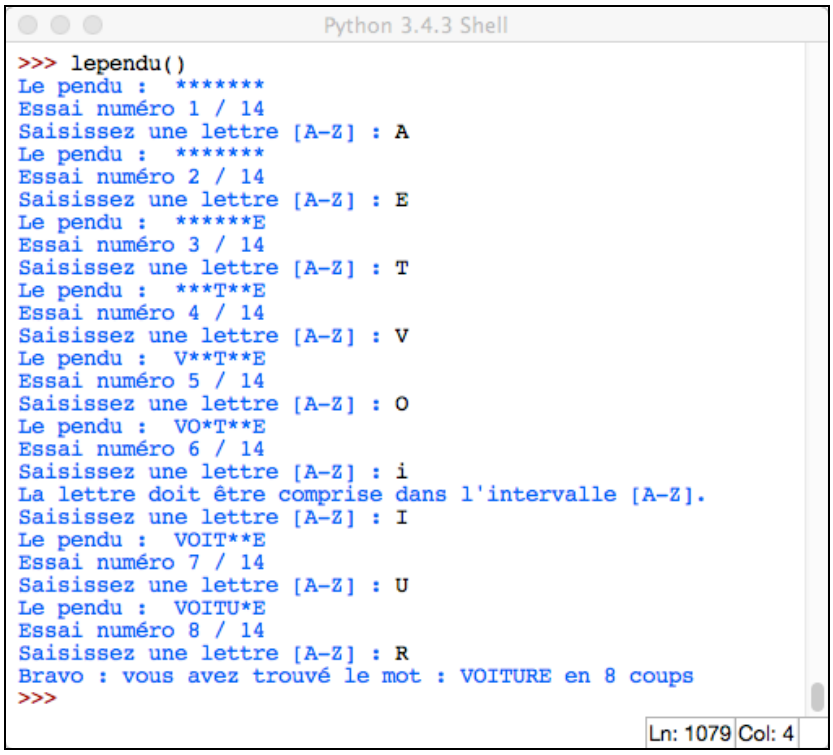


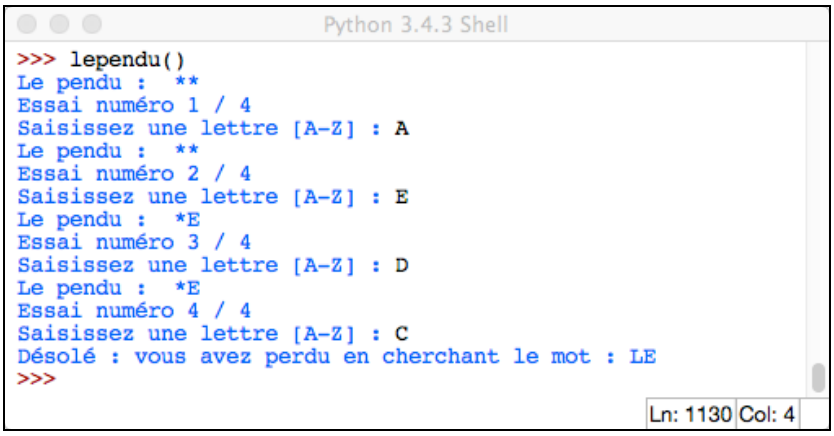
Partie 5 : la fonction principale

Ecrivez la fonction *lependu()* : son rôle est de lancer une partie du jeu du Pendu avec l'utilisateur. Son fonctionnement sera le suivant :

- la fonction affiche l'état du pendu (les étoiles sont les lettres par encore dévoilées) ;
- la fonction affiche ensuite le numéro de la prochaine tentative sur le nombre maximal de tentatives possibles ;
- la fonction demande à l'utilisateur de choisir une lettre (appel de la fonction *demande_lettre*) ;
- la fonction revient en début tant que le mot n'est pas complètement dévoilé ou tant que le nombre de tentatives maximal n'est pas atteint ;
- la fonction affiche finalement un message différent selon que l'utilisateur a gagné ou que l'utilisateur a perdu.

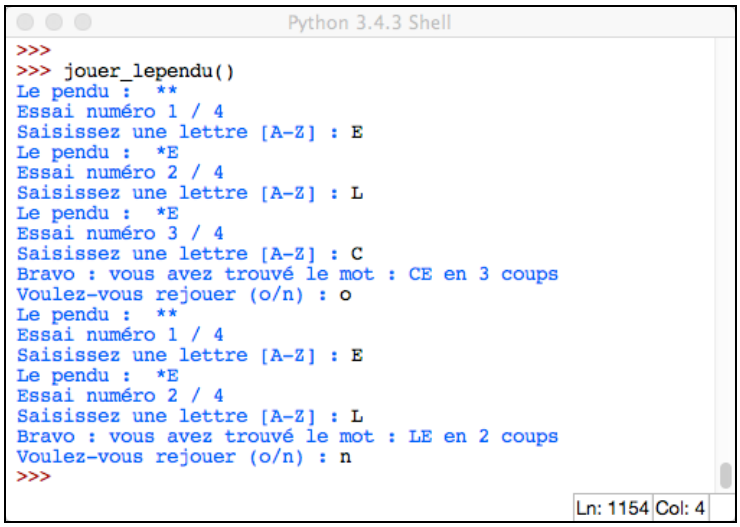
Regardez les deux exemples de fonctionnement suivants :





Partie 6 : jouer tant que désiré

Ecrivez une fonction *jouer_lependu()* qui permet de jouer au jeu du Pendu tant que l'utilisateur le souhaite comme dans l'exemple de l'image suivante :

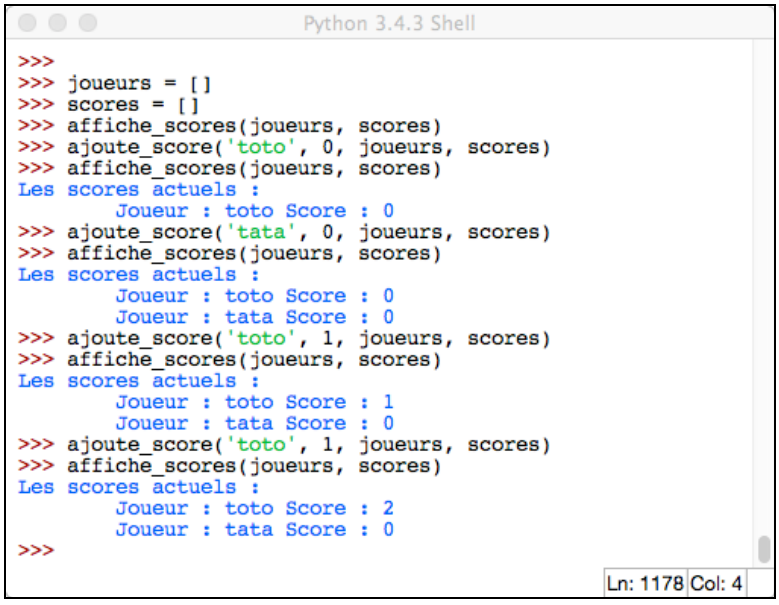


Partie 7 : gestion des joueurs et des scores

Ecrivez une fonction *ajoute_scores(joueur, score, joueurs, scores)* qui permet de mettre à jour la liste des joueurs et la liste des scores et une fonction *affiche_scores(joueurs, scores)* qui affiche les scores des joueurs.

Dans les deux fonctions, *joueurs* et *scores* sont des listes : *joueurs* est une liste contenant les noms des joueurs et *scores* et une liste contenant les scores (nombre entier) des joueurs.

Les deux listes doivent être cohérentes entre elles : elles doivent avoir la même taille et pour un même joueur, l'index dans la liste des joueurs doit correspondre à l'index dans la liste des scores.



Partie 8 : intégration de la gestion des joueurs

Modifiez la fonction *jouer_lependu()* en intégrant la gestion des joueurs et des scores. Vous devrez également modifier la fonction *lependu()* pour qu'elle retourne le résultat du joueur (perdu = 0 et gagné = 1).