

TP 12/30 – Places de Parking – Python exo n°4

Contexte.

Vous êtes le gérant d'un parking commercial comportant actuellement 27 emplacements pour accueillir les véhicules des clients du centre commercial.

Niveau 1.

Afficher « *Bienvenue au niveau -1, que souhaitez-vous faire ?* »

Créer une liste nommé parking contenant 27 emplacements :

- La lettre 'D' correspond à un emplacement **disponible**
- La lettre 'V' correspond à un emplacement **contenant un véhicule**
- La lettre 'H' correspond à un emplacement **pour les personnes à mobilité réduite**.

Afficher si l'emplacement n°3 est **disponible** ou **indisponible**

Placer une voiture au 1^{er} emplacement du parking (n°0)

Afficher tous les emplacements du parking en précisant leurs disponibilités

Niveau 2.

Proposer à l'utilisateur de d'entrer le chiffre 1 ou 2

1. Lui permet de garer une voiture en précisant le numéro de l'emplacement

2. Lui permet de recupérer une voiture en précisant le numéro de l'emplacement

Créer une fonction `afficher_parking()` affichant les emplacements

Niveau 3.

Après plusieurs mois de fonctionnement, vous avez décidé d'agrandir le parking en construisant l'étage -2 et l'étage -3 comportant respectivement 27 nouveaux emplacements.

Apporter les modifications par rapport à l'exercice précédant pour prendre compte ces deux nouveaux étages.

Créer un dictionnaire nommé `codes_debloquage`, il va permettre de définir un code secret lorsqu'une voiture est garée. Si le code est incorrect lors du déblocage, afficher « Stop ! »

Format du code de déblocage : AAAA-2222-XX

Niveau 4.

Définir au hasard deux emplacements sur chaque étage étant un accès prioritairement réservé aux **personnes à mobilité réduite**.

Ils ne pourront pas être sélectionné sauf en donnant un code contenant le format suivant HP-1111-X3 soit :

- HP au départ + un tiret + 4 chiffres + X3

Voir l'outil regex (module re) : https://www.w3schools.com/python/python_regex.asp