



El futuro digital
es de todos

MinTIC

«Misión
TIC 2022»

Reto



Universidad de Caldas



Reto Módulo 5: PARQUEADERO

Objetivo

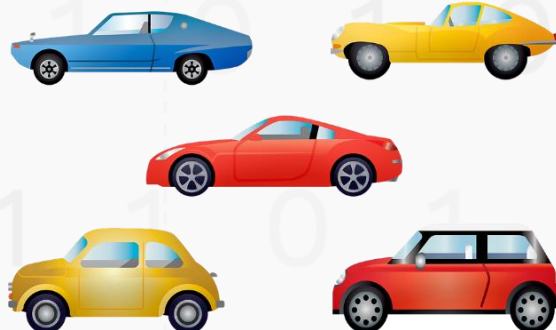
Desarrollar un programa en Python utilizando Replit que se le suministren dos listas: La primera con productos y la segunda con las cantidades.

Descripción del Reto

En el centro de la ciudad abrieron un pequeño parqueadero con capacidad para 9 automóviles.

El parqueo es muy simple, pues si hay un espacio vacío se caracteriza como Libre, Sino se caracteriza como ocupado.

De esa manera se requiere implementar un programa que permita mostrar los espacios ocupados o libres según datos suministrados.



Una matriz de 3 x 3 muestra los puestos entre 1 y 9, en donde se actualiza la liberación u ocupación según entren o salgan automóviles del parqueadero, desde una caseta situada en la entrada del parqueadero.



1	2	3		
←→			↑	
4	5	6		
←→			↓	
7	8	9		Caseta

¿Entonces qué debe hacer el programa en Python?

- Contener una matriz llamada ocupación con estados False y True, False si está desocupado True si está ocupado. Se debe inicializar en False, todo vacío.

```
ocupacion = [[False,False,False],  
             [False,False,False],  
             [False,False,False]]
```

- Contener un directorio con el número del parqueadero y las coordenadas de la matriz ocupación

```
puestos = {1:[0,0],2:[0,1],3:[0,2],  
           4:[1,0],5:[1,1],6:[1,2],  
           7:[2,0],8:[2,1],9:[2,2]  
           }
```

- Debe llamar una función validar para verificar la libertad u ocupación del parqueadero, el formato de ella es → **def validar(puesto, ocupacion):**, ella recibe la matriz de ocupación y el puesto a validar, ella retorna los Strings 'Libre' o 'Ocupado', según sea el caso.
- Luego debe llamar la función ocupar → **def ocupar(puesto, ocupacion, validacion):**, ella recibe el puesto, la matriz de ocupación y la previa validación para ocupar el parqueadero. Debe retornar la nueva matriz de Ocupación, obviamente verificando si realmente se puede ocupar ese parqueadero.
- Finalmente, se debe llamar una Función → **def liberar(puesto, ocupacion, validacion):**, ella recibe el puesto, la matriz de ocupación y la previa validación para ocupar el parqueadero. Debe retornar la nueva matriz de Ocupación, obviamente verificando si realmente se puede Liberar ese parqueadero.



- Implemente Listas y Directorios para solucionar el problema.
- Implemente IF o IF Else, o las condiciones que requiera. También implemente Ciclos WHILE o FOR según lo considere.
- El código final se debe cargar en Moodle, el cual se autocalificará automáticamente.
- *Cuando Cargue las funciones **def** a Moodle, previamente también debe cargar las **Lista Ocupación y el Directorio**.*
- Finalmente, es importante que las funciones Retornen los valores solicitados.

Link: Estructura de Apoyo en Colab

https://colab.research.google.com/drive/1sHO_tiCOvXXKJA_k3lI14_jn1n8iZivD?usp=sharing



Test

TEST	Salida Esperada
<pre>validacion = validar(2,ocupacion) #Ocupar print(validacion) ocupacion = ocupar(2,ocupacion,validacion) print(ocupacion[0][1]) validacion = validar(2,ocupacion) #Intento Ocupar si está lleno print(validacion) ocupacion = ocupar(2,ocupacion,validacion) print(ocupacion[0][1]) validacion = validar(2,ocupacion)#Liberar print(validacion) ocupacion = liberar(2,ocupacion,validacion) print(ocupacion[0][1])</pre>	Libre True Ocupado True Ocupado False
<pre>validacion = validar(9,ocupacion) print(validacion) ocupacion = ocupar(9,ocupacion,validacion) print(ocupacion[2][2]) validacion = validar(9,ocupacion) print(validacion) ocupacion = ocupar(9,ocupacion,validacion) print(ocupacion[2][2]) validacion = validar(9,ocupacion) print(validacion) ocupacion = liberar(9,ocupacion,validacion) print(ocupacion[2][2])</pre>	Libre True Ocupado True Ocupado False



Fecha Modo de entrega

- Fecha máxima: Miércoles 9 de Junio 2021 hasta las 11:59 pm
- Ruta de entrega: Botón PRESENTAR RETO DE LA SEMANA módulo 5

