



Ciclo 1 Fundamentos de programación

Reto 2

Descripción del problema: Un estacionamiento desea conocer el valor que debe cobrar a sus clientes dependiendo del tiempo (hora y fracción en minutos) que duró el vehículo del cliente estacionado en el parqueadero así:

Al ingresar al parqueadero se cobra una hora si no tiene membresía e independiente de la fracción consumida, por ejemplo, si el cliente ocupa el estacionamiento por 0 horas y 35 minutos se cobra la hora completa, pero si tiene una membresía vigente solo se le cobra media hora. Para tiempos superiores a 1 hora se sigue la siguiente regla: si la fracción es menor o igual a 10 minutos solo se cobra las horas consumidas si tiene una membresía vigente en caso contrario se le cobra media hora adicional, por ejemplo, si el cliente ocupa el estacionamiento por 2 horas y 10 minutos y tiene una membresía vigente se cobran solo las dos horas; si la fracción es superior a 10 minutos e inferior o igual 30 minutos se cobran las horas consumidas más el 30% del valor de una hora si tiene una membresía vigente en caso contrario se le cobra el 80% de una hora adicional, por ejemplo, si el cliente ocupa el estacionamiento por 3 horas y 23 minutos y no tiene una membresía vigente se le cobran el costo de cuatro horas; si la fracción es superior a 30 minutos se cobra el 50% del valor de la hora si tiene una membresía vigente en caso contrario se le cobra una hora completa, por ejemplo, si el cliente ocupa el estacionamiento por 1 hora y 46 minutos y no tiene membresía se cobran el costo de dos horas. En todos los casos se deberá cobrar un IVA que será indicado por el estacionamiento. Escriba una función que reciba como parámetro un diccionario en el cuál las llaves son la placa del vehículo, cuatro enteros correspondientes al valor de la hora de estacionamiento, la cantidad de horas que duró estacionado el vehículo, la fracción en minutos consumidas por el cliente y el IVA a cobrar y una llave booleana membresía (True= si posee membresía, False = no posee membresía). La función deberá crear un nuevo diccionario de salida con las llaves 'placa' y 'valor'. La función deberá retornar una cadena que le proporcionará al estacionamiento la información del valor a pagar por parte del cliente así: "El valor a pagar por el vehículo {diccionarioSalida['placa']} es {diccionarioSalida['valorPagar']}".



Ejemplo:

Placa	Valor Hora	Horas	Fracción	Membresía	IVA	return
EUQ587	4000	4	17	True	19	El valor a pagar por el vehículo EUQ587 es 20468.0

Diccionario de Entrada:

Llaves	Tipo	Descripción
placaVehiculo	str	Placa del vehículo estacionado por el cliente.
valorHora	int	Valor de la hora a cobrar
horas	int	Cantidad de horas que duró estacionado el vehículo.
fraccion	int	Fracción en minutos que duró estacionado el vehículo.
membresia	bool	True= si posee membresía, False = no posee membresía
iva	int	Iva a aplicar.

Salida:

Tipo del retorno	Descripción
str	"El valor a pagar por el vehículo {diccionarioSalida['placa']} es {diccionarioSalida['valorPagar']}", dónde, el valor a pagar por el cliente debe ser calculado acorde a las condiciones indicadas en el enunciado.

Esqueleto:

```
def cobroEstacionamiento (datos: dict) -> dict:
```

```
    pass
```