

# Sistema de Reserva de Vehículos

## Introducción

El objetivo de este proyecto es desarrollar un sistema de reserva de vehículos utilizando Python. Este sistema permitirá a los usuarios gestionar clientes, vehículos y reservas, además de guardar la información de las reservas en un archivo Excel para su posterior análisis y consulta.

## Objetivos

1. Desarrollar Clases en Python (TAMBIEN FUNCIONA CON PARADIGMA FUNCIONAL) :
  - Implementar clases para gestionar clientes, vehículos y reservas.
  - Crear métodos dentro de las clases para realizar operaciones como alquilar y devolver vehículos.
2. Utilizar Estructura de Paquetes en Python:
  - Organizar el código utilizando la estructura de paquetes para mantener el código modular y fácil de mantener.
3. Guardar Datos en Excel:
  - Utilizar la biblioteca **pandas** para guardar la información de las reservas en un archivo Excel.

## Estructura del Proyecto

El proyecto debe seguir la siguiente estructura de archivos:

```
vehiculos/  
|  
├─ __init__.py  
├─ reservas.py  
├─ vehiculos.py  
├─ clientes.py  
example.py
```

## Descripción de los Archivos

### **vehiculos/init.py:**

- Archivo de inicialización del paquete. Puede estar vacío.

### **vehiculos/clientes.py:**

- Contiene la clase `Cliente` que almacena la información de los clientes.
- Métodos:
  - `__init__(self, nombre, apellido, licencia_conducir)`: Inicializa un cliente con su nombre, apellido y licencia de conducir.
  - `__str__(self)`: Devuelve una representación en cadena del cliente.

### **vehiculos/vehiculos.py:**

- Contiene la clase `Vehiculo` que almacena la información de los vehículos.
- Métodos:
  - `__init__(self, matricula, modelo, precio_por_dia)`: Inicializa un vehículo con su matrícula, modelo y precio por día.
  - `__str__(self)`: Devuelve una representación en cadena del vehículo.
  - `alquilar(self)`: Marca el vehículo como no disponible.
  - `devolver(self)`: Marca el vehículo como disponible.

### **vehiculos/reservas.py:**

- Contiene la clase `Reserva` que almacena la información de las reservas.
- Métodos:
  - `__init__(self, cliente, vehiculo, fecha_inicio, fecha_fin)`: Inicializa una reserva con el cliente, vehículo, fecha de inicio y fecha de fin.
  - `__str__(self)`: Devuelve una representación en cadena de la reserva.
- Función:
  - `guardar_reservas_en_excel(reservas, archivo)`: Guarda una lista de reservas en un archivo Excel.

### **example.py:**

- Script de ejemplo que demuestra cómo utilizar las clases y funciones definidas en el paquete `vehiculos`.
- Pasos:
  - Crear instancias de `Cliente` y `Vehiculo`.
  - Crear instancias de `Reserva` y marcar vehículos como alquilados.
  - Guardar las reservas en un archivo Excel.
  - Devolver los vehículos y mostrar su estado.

## **Requisitos**

- Python 3.x
- Biblioteca `pandas` (se puede instalar usando `pip install pandas openpyxl`)

## **Instrucciones de Ejecución**

1. Asegúrate de que todos los archivos están en la estructura de carpetas mencionada.

Ejecuta el script `example.py` desde el directorio que contiene la carpeta `vehiculos`:

```
python example.py
```

2. El script generará un archivo `reservas_vehiculos.xlsx` con la información de las reservas.

## **Resultados Esperados**

- El sistema debe crear y gestionar reservas de vehículos.
- La información de las reservas debe guardarse correctamente en un archivo Excel.
- Al ejecutar el script de ejemplo, se deben mostrar en consola las reservas realizadas y el estado de los vehículos después de devolverlos.

## **Conclusión**

Este proyecto está realizado para las alumnas de TIPTI, para la asesoría opcional de Python con el Tutor Santino Ursino.

Integra la funcionalidad de guardar datos en un archivo Excel, lo que es útil para el análisis y la gestión de datos.