מגישים:

אופיר כהן (ת.ז 208711234) ושגיא בן נון (ת.ז 206446288)

**רשימת טיפוסים:**

**class UserInterface-** מחלקה השייכת ל ConsoleUI המחברת בין הלוגיקה של המוסך לבין המשתמש.

**class Garage-** מחלקה שמכילה את המוסך כולו. ע"י שימוש במחלקה זו, ניתן לבצע את כל הפעולות הנדרשות במסמך התרגיל (למשל תדלוק רכב או הוספת רכב למוסך).

**enum eVehicleTypes-** enum השייך למחלקה Garage ומייצג את כל סוגי הרכבים הנתמכים במוסך בשלב זה.

**class VehicleGenerator-** מחלקה שמייצרת רכב לפי סוג רכב וה properties המתאימים (הסבר על ה properties בהמשך הדף).

**class CustomerInfo-** מחלקה המאחסנת מידע עבור הלקוחות (דף אפיון לכל רכב הכולל את שם ומספר הטלפון של הבעלים וסטטוס).

**enum eVehicleStatus-** enum שמייצג את הסטטוס של הרכב שנמצא במוסך (בטיפול, טופל או שולם).

**abstract class EnergySource-** מחלקה אבסטרקטית שמייצגת מקור אנרגיה של המנוע הנמצא ברכב.

**class FuelEngine,ElectricEngine-** מחלקות היורשות מ EnergySource, המכילות מידע אודות סוג המנוע (חשמלי או דלק).

**struct Wheel-** מבנה המייצג גלגל של רכב. מכיל מידע אודות שם היצרן ולחץ האוויר.

**abstract class Vehicle-** מחלקה אבסטרקטית המכילה מידע הקיים בכל סוג רכב שהוא (כגון רשימת גלגלים, מנוע, שם מודל ולוחית רישוי). בנוסף, קיימת בה מטודה וירטואלית המחזירה מידע אודות הרכב.

**class Truck,Car,Motorcycle-** מחלקות היורשות מ Vehicle, המכילות מידע ספציפי נוסף אודות סוג הרכב כולל enums ספציפיים (למשל eCarColor השייך ל Car המייצג את צבע המכונית ו eLicenseType השייך ל Motorcycle המייצג את סוג הרישיון).

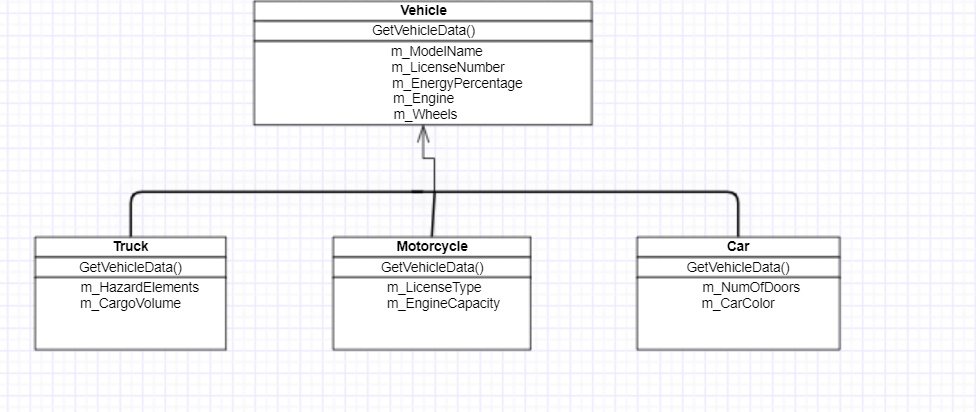
**class VehicleProperties-** מחלקה המכילה מידע הדרוש על מנת לייצר רכב.

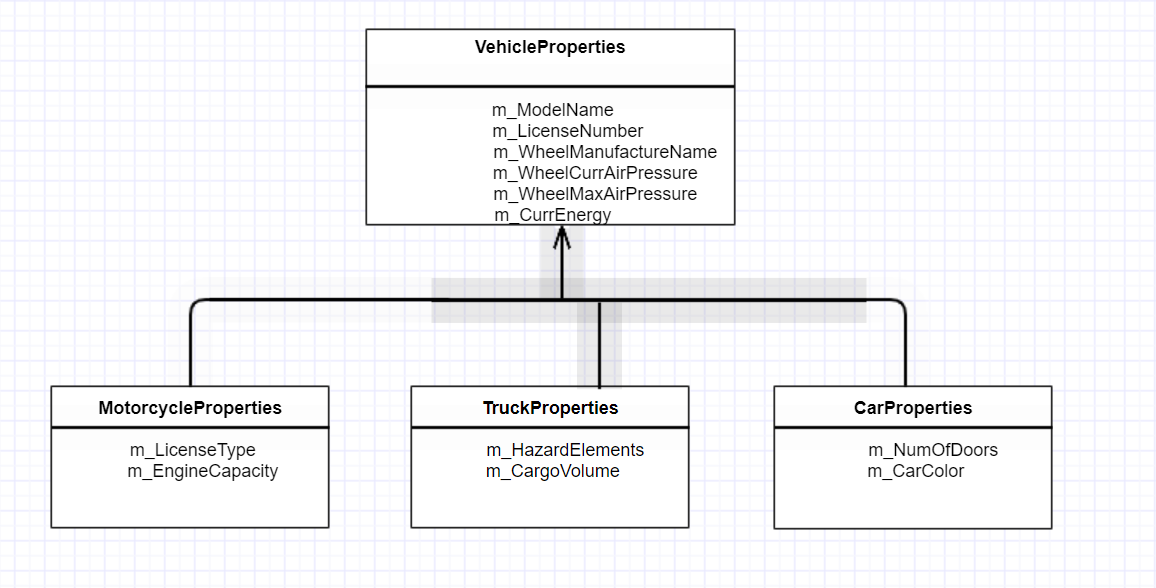
**class TruckProperties,CarProperties,MotorcycleProperties-** מחלקות היורשות מ VehicleProperties המכילות את שאר המידע הדרוש על מנת לייצר סוג רכב ספציפי.

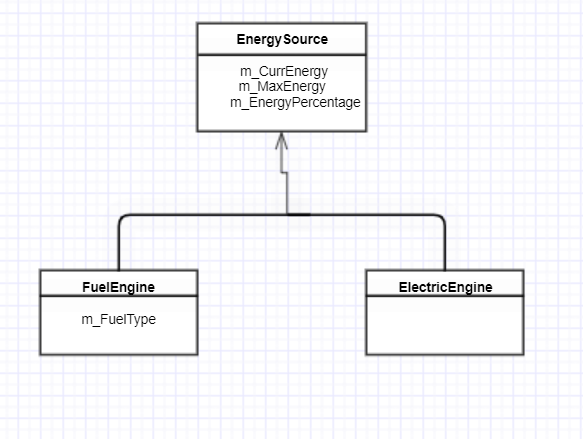
\*מחלקות ה Properties נועדו על מנת למנוע שכפול קוד או הוספת מטודות מיותרות, בעקבות כך יש צורך רק במטודה אחת היכולה לייצר את כל סוגי הרכבים הנתמכים במוסך. כדי לתמוך בסוג רכב חדש במוסך, יש צורך לשנות את הקוד במקומות הבאים בלבד: הוספה של שדה נוסף המייצג את סוג הרכב החדש ב **eVehicleTypes** והוספת שורת else if(…) במטודה GenerateVehicle() במחלקה **VehicleGenerator**. בנוסף, יש צורך לייצר מחלקה מתאימה לסוג הרכב החדש היורשת מ Vehicle ו Properties מתאים היורש מ VehicleProperties.

בנוסף, יצרנו שלוש מחלקות Exceptions חדשות היורשות מ Exception, הכוללות את המחלקה ValueOutOfRangeException שנדרשנו לממש במסגרת התרגיל.

**היררכיות ההורשה:**





תמונה שמכילה טקסט, תרשים, קו, מלבן

התיאור נוצר באופן אוטומטי

**הקשר בין הרכיבים:**

