

Case 1 - Encapsulation

Build a class called as Car Create Car class – Encapsulation

- **Instance members:**
 1. VehicleId – This should be auto incremented for each object.
 2. RegistrationNumber: Is of type string.
 3. DateOfManufacture – Is of Type DateTime
 4. Color – Is of type Enum and should be value from White, Black, Red, Grey, Brown
 5. Make – Is of type string
 6. Model – Is of type string
 7. Price – Is of type Decimal.
 8. CurrentPrice – Readonly decimal property and should return value depreciated at 10% per annum. If actual price is 100 after One year its CurrentPrice would be 90(100-10%) and after two years it would be 81 (90-10%) and so on
 9. Status – A readonly string which shows the current speed of car.
- **Instance Constructor:**
 - Provide constructor with Parameters to initialize all the Instance Members
 - Set Status to "Congratulations on Purchasing a new car"
- **Static Members:**
 - MaxAccelerateSpeed – When we accelerate the car we cannot increase speed by this amount.
 - MinDecelerateSpeed – When we decelerate the car, we cannot decrease speed by this amount.
 - ServiceYears: After these many years, Car cannot Start and will throw error – "Car Expired"
- **Instance Methods:** Start(), Stop(), Accelerate(int offsetSpeed), Decelerate(int offsetSpeed)
 - I. **Start()** method
 - a. Set the car to initial speed of 20.
 - b. After Service Years, Car should throw exception with message "Car Expired" and not start.
 - c. Change the Status accordingly eg. "Car Started at a speed of 20"
 - II. **Stop()** method
 - a. Set the car speed to 0
 - b. Set the Status accordingly eg. "Car Stopped"
 - III. **Accelerate(int offsetSpeed):**
 - a. To increase Speed of Car by offsetSpeed
 - b. **offsetSpeed** should not be greater than MaxAccerateSpeed.
 - c. **Car Speed should not go beyond 140**
 - IV. **Decelerate(int offsetSpeed)**
 - a. To decrease Speed of Car by offsetSpeed
 - b. offsetSpeed speed should not be less than MinDecelarateSpeed.

- c. Car Speed cannot be less than 0.

Build the GUI as below and program each button using the above Car class.

The screenshot shows a Windows application window titled "Vehicle Form". The window has a standard Windows title bar with minimize, maximize, and close buttons. The main content area is divided into two sections. On the left, there are seven input fields with labels: "VehicleId", "RegNo.", "Date of Manufacture" (which includes a calendar icon), "Color", "Make", "Model", and "Price". On the right, there are several buttons: "Create", "Get", "Clear", "Start", "Stop", "Accelerate", and "Deaccelerate". Below the "Accelerate" button, there is a label "Current Status Of Vehicle" and a small, empty rectangular box. A large, diagonal watermark reading "Deccansoft" is visible across the entire image.