|  |  |
| --- | --- |
| EX:NO:4b | Implementing real-time/technical applications using Lists, Tuples.(Components of a car) |
| Date: |

**Aim:**

To Implement real-time/technical applications using Lists, Tuples.

(Components of a car)

**Algorithm:**

Step1: Start the program

Step2: Assign list of car parts

Step3: Perform all the list operation

Step4: Print the Result

Step5: End the program

**Program:**

class Car:

def \_\_init\_\_(self):

self.components = []

def add\_component(self, component\_name, component\_type):

component = (component\_name, component\_type)

self.components.append(component)

print(f"Added {component\_type} component: {component\_name}")

def display\_components(self):

if not self.components:

print("No components added to the car.")

else:

print("Car Components:")

for component in self.components:

component\_name, component\_type = component

print(f"Type: {component\_type}, Name: {component\_name}")

def main():

car = Car()

while True:

print("\nCar Component Management System")

print("1. Add Component")

print("2. Display Components")

print("3. Exit")

choice = input("Enter your choice: ")

if choice == '1':

component\_name = input("Enter component name: ")

component\_type = input("Enter component type: ")

car.add\_component(component\_name, component\_type)

elif choice == '2':

car.display\_components()

elif choice == '3':

print("Goodbye!")

break

else:

print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Result:**