Practical – 16

CODE:

```
#include <iostream>
#include <fstream>
#include <sstream>
#include <string>

using namespace std;

class Car {
    public:
    string name;
    string fuelType;
    double mileage;
    double price;
    string transmission;
    int tankCapacity;
    int seatingCapacity;
    string isAutomatic;
};

Car readCarFromFile(ifstream& file) {
    Car car;
    string line;
```

```
    if (getline(file, line, ';')) {
        stringstream ss(line);
        getline(ss, car.name, ',');
        getline(ss, car.fuelType, ',');
        ss >> car.mileage;
        ss.ignore(); // Ignore the comma
        ss >> car.price;
        ss.ignore(); // Ignore the comma
        getline(ss, car.transmission, ',');
        ss >> car.tankCapacity;
        ss.ignore(); // Ignore the comma
        ss >> car.seatingCapacity;
        ss.ignore(); // Ignore the comma
        getline(ss, car.isAutomatic, ',');
    }
    return car;
}

void displayCarDetails(const Car& car) {
    cout << "Name: " << car.name << endl;
    cout << "Fuel Type: " << car.fuelType << endl;
    cout << "Mileage: " << car.mileage << " kmpl" << endl;
    cout << "Price: $" << car.price << endl;
    cout << "Transmission: " << car.transmission << endl;
    cout << "Tank Capacity: " << car.tankCapacity << " liters" << endl;
    cout << "Seating Capacity: " << car.seatingCapacity << " persons" << endl;
```

```
    cout << "Automatic: " << car.isAutomatic << endl;
}

void retrieveCarByModel(ifstream& file, const string& modelName) {
    file.clear(); // Clear eof flag
    file.seekg(0, ios::beg); // Rewind file
    Car car;
    while ( file) {
        car = readCarFromFile(file);
        if (car.name == modelName) {
            cout << "Details of " << modelName << ":\n";
            displayCarDetails(car);
        }
    }
    }


void retrieveCarByFuelType(ifstream& file, const string& fuelType) {
    file.clear(); // Clear eof flag
    file.seekg(0, ios::beg); // Rewind file
    Car car;
    cout << "Cars with fuel type " << fuelType << ":\n";
    while (file) {
        car = readCarFromFile(file);
        if (car.fuelType == fuelType) {
            displayCarDetails(car);
```

```
            cout << "----------------\n";
        }
    }
}

void retrieveCarByPriceRange(ifstream& file, double minPrice, double maxPrice)
{
    file.clear(); // Clear eof flag
    file.seekg(0, ios::beg); // Rewind file
    Car car;
    cout << "Cars within price range $" << minPrice << " - $" << maxPrice << ":\n";
    while (file) {
        car = readCarFromFile(file);
        if (car.price >= minPrice && car.price <= maxPrice) {
            displayCarDetails(car);
            cout << "----------------\n";
        }
    }
}

int main() {
    ifstream file("cardata.txt");
    if (!file.is_open()) {
        cout << "Error opening file cardata.txt" << endl;
        return 1;
    }
```

```
    cout << "Data loaded successfully.\n";

    int option;
    cout << "Welcome To Tata Motors\n";
    cout << "Choose an option:\n";
    cout << "1. Retrieve car details by model\n";
    cout << "2. Retrieve car details by fuel type\n";
    cout << "3. Retrieve car details by price range\n";
    cin >> option;

    switch (option) {
        case 1: {
            cout << "1.TIAGO   2.PUNCH   3.ALTROZ  4.TIGOR   5.NEXON\n6.HARRIER
7.SAFARI\n";
            string modelName;
            cout << "Enter the model name: ";
            cin >> modelName;
            retrieveCarByModel(file, modelName);
            break;
        }
        case 2: {
            cout << "1.PETROL   2.DIESEL   3.CNG      4.ELECTRIC\n";
            string fuelType;
            cout << "Enter the fuel type: ";
            cin >> fuelType;
```

```
            retrieveCarByFuelType(file, fuelType);
            break;
        }
        case 3: {
            double minPrice=0, maxPrice;

            cout << "Enter the maximum price: ";
            cin >> maxPrice;
            retrieveCarByPriceRange(file, minPrice, maxPrice);
            break;
        }
        default:
            cout << "Invalid option!\n";
    }

    file.close();
    return 0;
}
```

OUTPUT:

```
Data loaded successfully.
Welcome To Tata Motors
Choose an option:
1. Retrieve car details by model
2. Retrieve car details by fuel type
3. Retrieve car details by price range
1
1.TIAGO    2.PUNCH    3.ALTROZ   4.TIGOR    5.NEXON
6.HARRIER  7.SAFARI
Enter the model name: TIAGO
Details of TIAGO:
Name: TIAGO
Fuel Type: Petrol
Mileage: 8.2 kmpl
Price: $19
Transmission: Manual
Tank Capacity: 60 liters
Seating Capacity: 5 persons
Automatic: yes

Process returned 0 (0x0)   execution time : 67.073 s
Press any key to continue.
```

```
Data loaded successfully.
Welcome To Tata Motors
Choose an option:
1. Retrieve car details by model
2. Retrieve car details by fuel type
3. Retrieve car details by price range
2
1.PETROL   2.DIESEL   3.CNG      4.ELECTRIC
Enter the fuel type: CNG
Cars with fuel type CNG:
Name: TIGOR
Fuel Type: CNG
Mileage: 9.55 kmpl
Price: $19.28
Transmission: AMT
Tank Capacity: 35 liters
Seating Capacity: 5 persons
Automatic: No
----------------------

Process returned 0 (0x0)   execution time : 11.499 s
Press any key to continue.
```

```
Welcome To Tata Motors
Choose an option:
1. Retrieve car details by model
2. Retrieve car details by fuel type
3. Retrieve car details by price range
3
Enter the maximum price: 15
Cars within price range $0 - $15:
Name: SAFARI
Fuel Type: Diesel
Mileage: 27.34 kmpl
Price: $14.5
Transmission: AMT
Tank Capacity: 50 liters
Seating Capacity: 7 persons
Automatic: Yes
----------------------
Name:
Fuel Type:
Mileage: 27.34 kmpl
Price: $14.5
Transmission:
Tank Capacity: 50 liters
Seating Capacity: 7 persons
Automatic:
----------------------

Process returned 0 (0x0)   execution time : 10.288 s
Press any key to continue.
```