CAR PARKING SYSTEM IN C++

by

Aditya Bal Gupta
Indian Institute of Technology, Ropar, Punjab

Working of the Program:

When we run the code, these choices come up. We can the choose any of the available options by entering the choice number.

```
_____________________________________________________
                VEHICLE PARKING SYSTEM
Available choices:
1. To add a new vehicle
2. To get total number of vehicles parked
3. For departure of vehicle
4. To get total amount of money collected
5. Display vehicles parked currently
6. Exit
_____________________________________________________
Enter your choice:
```

1. **To add a new Vehicle:**
   This option is used to add a new vehicle into our parking system.

   Implementation:

```
void vehicle::add_vehicle() {
    vehicle *v = new vehicle;
    char c1, c2;
    cout << "Enter type of vehicle (1 for Car, 2 for Bike) : ";
    cin >> v->type;
    cout << "Enter vehicle registration number : ";
    cin >> v->reg_number;
    cout << "Enter arrival time (HH.MM.SS) : ";
    cin >> v->arrival.hour >> c1 >> v->arrival.minute >> c2 >> v->arrival.second;
    cout << "Enter Date (DD/MM/YYYY): ";
    cin >> v->dt.day >> c1 >> v->dt.month >> c2 >> v->dt.year;

    veh.at(i).reg_number = v->reg_number;
    veh.at(i).type = v->type;
    veh.at(i).arrival.hour = v->arrival.hour;
    veh.at(i).arrival.minute = v->arrival.minute;
    veh.at(i).arrival.second = v->arrival.second;
    veh.at(i).dt.day = v->dt.day;
    veh.at(i).dt.month = v->dt.month;
    veh.at(i).dt.year = v->dt.year;

    i++;
    vehicle_count++;

    if (v->type == 1)
        car_count++;
    else
        bike_count++;

    cout << "\nVehicle added" << endl;
}
```

- User is asked to enter the type of vehicle. (1 for car and 2 for bike)
- Then the user is asked to enter Vehicle registration number.
- Then arrival time and arrival date are added.
- Count of vehicles currently parked is increased by 1.

2. **To get total number of vehicles parked**
   Used to get the total number of bikes and cars currently parked.

   Implementation:

```
void disp_tot_veh(){
    cout << "Number of vehicles parked currently : " << vehicle_count << endl;
    cout << "Number of cars parked currently : " << car_count << endl;
    cout << "Number of bikes parked currently : " <<  bike_count << endl;
}
```

3. **For departure of vehicle**

   Implementation:

```
void vehicle:: vehicle_departure() {
    string reg_num;
    int typ;
    Time depart;
    int Time_diff;
    int charge = 0;
    char c1,c2;

    cout << "Enter vehicle type(1 for Car/2 for Bike) : ";
    cin >> typ;
    cout << "Enter vehicle number : ";
    cin >> reg_num;
    cout << "Enter departure Time (HH.MM.SS) : ";
    cin >> depart.hour >> c1 >> depart.minute >> c2 >> depart.second;


    for (int j = 0; j <= i; j++){
        if ((veh.at(j).reg_number == reg_num) && (veh.at(j).type == typ)){
            veh.at(j).departure.hour = depart.hour;
            veh.at(j).departure.minute = depart.minute;
            veh.at(j).departure.second = depart.second;

            Time_diff = TimeDifference(veh.at(j).arrival, depart);

            if (veh.at(j).type == 1){
                car_count--;
                if (Time_diff < 2)
                    charge = 30;
                else{
                    if ((Time_diff > 2) && ((Time_diff < 5)))
                        charge = 40;
                    else
                        charge = 60;
                }
            }
```

```
            else{
                bike_count--;

                if ( Time_diff < 2)
                    charge = 10;
                else{
                    if ((Time_diff > 2) && ((Time_diff < 5)))
                        charge = 15;
                    else
                        charge = 20;
                }

            }

            cout << "\nVehicle having vehicle number  : " << veh.at(j).reg_number
            << " has left the parking after paying Rs. " << charge << endl;

            veh.erase(veh.begin() + j);
            i--;
            vehicle_count--;
            money_collected = money_collected + charge;
            break;
        }

        if (j == i){
            cout << "\nWrong Entry , Try Again " << endl << endl;
            cout << "Departure : " << endl;
            vehicle_departure();
        }
    }
}
```

- User is asked to enter the type of vehicle. (1 for car and 2 for bike)
- Then the user is asked to enter Vehicle registration number.
- Then user is asked to enter departure time.
- Count of vehicles currently parked is decreased by 1.
- According to the duration of time the vehicle is parked, and the type of the vehicle, amount of fee is also calculated.

4. **To get total amount of money collected**

   To get the total amount of money collected, calculated in the departure section of the code.

   Implementation:

```
void disp_tot_amount(){
    cout << "Total amount of money collected : " << money_collected << endl;
}
```

5. **Display vehicles parked currently**

   To display all the vehicles in the parking system at any moment.

Implementation:

```
void vehicle::disp_vehicles(){
    cout << "Vehicle Type\t\tVehicle Reg. Number\t\t\tArrival Date\t\t\tArrival Time"<< endl;
    for (int j = 0; j < i; j++){
        cout << veh[j].type << "\t\t\t" << veh[j].reg_number << "\t\t\t\t" << veh[j].dt.day << "/"
        << veh[j].dt.month << "/" << veh[j].dt.year << "\t\t\t" << veh[j].arrival.hour << ":"
        << veh[j].arrival.minute << ":" << veh[j].arrival.second << endl;
    }
}
```

6. **Exit**

   To exit the program.