

VITYARTHI – PROJECT



Project title : Car Management System

Course title : Introduction To Problem Solving And Programming

Course code : CSE 1021

Course type : Flipped Course

Course credits : 4

Professor : Dr. Sivabalan KR

Slot : B14+B23+D21

Submitted by : Dilip S

Registration number : 25BAI10590

Submission date : 22/11/2025

VITYARTHI – PROJECT

INTRODUCTION

A vehicle management system is a powerful tool, which can be implemented to improve fleet efficiency, manage drivers' performance, track vehicle data, and integrate it into a company's ERP. They are also designed to monitor and control fuel consumption within your fleet.

In this List-based car management system you find the suitable car for you based on your need.

VITYARTHI – PROJECT

PROBLEM STATEMENT

This project contains a list of cars which are economical and expensive and the users can search for a car based on their need using different factors which can be convenient for them.

FUNCTIONAL REQUIREMENTS

MODULES AND FILES USED:

Files:

- **List(L):**

This file contains the list of cars as a list

User defined functions and their purposes:

Search():

Searchs for a specific car

Search_price():

Searchs based on price

Search_brand():

Searchs based on brand

Search_year():

Searchs for cars of specific year

VITYARTHI – PROJECT

Search_year2():

Searchs based on year range

Display():

All the cars are displays

NON-FUNCTIONAL REQUIREMENTS

1. Performance:

- The system shall respond with the list of cars to the user's inputs within seconds.
- The system shall handle a large number of the inputs entered by the user.

2. Security:

- The system will not hold / store any sensitive data of the user.
- The system shall be secure against all the common web vulnerabilities.

3. Usability:

- The system is made to be simple and easy for the user to use and the set of instruction will guide the user throughout their car selection process.

4. Reliability:

- The system shall be available anytime whenever the user wants to search for a car and the system can be accessed anywhere around the world with a python code executer.

5. Maintainability:

VITYARTHI – PROJECT

- The system is written with a maintainable and understandable code with comment lines.
 - The system has proper documentation so that it can be updated in the coming future.
6. Resource efficiency:
- The system is designed to minimise power consumption and optimize battery life.
 - The system shall require less than 1MB of storage space for optimised running.
 - The system doesn't require any network bandwidth so it can be accessed even in places without proper internet connection.

SYSTEM ARCHITECTURE

- Presentation layer is built using jupyter notebook, where the inputs are accepted from the user.
- Application layer is where the input data is processed and the searching is performed then the outputs are stored and displayed later.
- Data layer is a storage space where the user inputs and the output values are stored.

DESIGN DIAGRAM

VITYARTHI – PROJECT

1. Workflow:

User input ➡ Data validation ➡ searching for suitable car(price,year,brand,model) ➡processes ➡Result display

2. Data flow Diagram:

- input:
The system allows the user to enter the input based on their convenient.
- Process:
After getting the input the system processes it.
- Output:
Displays the processed data(the list of cars)

3. Component diagram :

- Input form: gets (year,price,brand,modal) as input
- Validation module: Checks all input ranges and types
- process core: searches based on the input
- Output display: Presents the results

VITYARTHI – PROJECT

SCREENSHOTS

```
Python 3.10.8 Shell
C:\Users\Nitya\Downloads\Nityarthi project-1.py (3.10.8)
File Edit Format Run Options Windows Help

# Data provided by the user
L = [['lamborghini', 'urque s', 2024, 237848.0],
     ['lamborghini', 'revuelto', 2024, 604363.0],
     ['lamborghini', 'huracan', 2024, 285000.0],
     ['lamborghini', 'countach', 2024, 2640000.0],
     ['lamborghini', 'essenza', 2024, 3751988.0],
     ['mercedes benz', 'sennar', 2018, 7638461.0],
     ['mercedes benz', 'maybach', 2023, 398809.0],
     ['mercedes benz', 'cabriolet', 2022, 132530.0],
     ['mercedes benz', 'limousine', 2023, 55421.0],
     ['mercedes benz', 'cabriolet', 2021, 132530.0],
     ['bmw', 'x3', 2019, 87349.0],
     ['bmw', 'x5', 2021, 116867.0],
     ['bmw', 'x7', 2023, 156626.0],
     ['bmw', 'x7', 2022, 301204.0],
     ['bmw', 'm4', 2018, 227710.0],
     ['honda', 'amaze', 2018, 8674.0],
     ['honda', 'city', 1981, 13132.0],
     ['honda', 'accord hybrid', 2018, 59590.0],
     ['honda', 'civic', 2020, 26927.0],
     ['honda', 'elevate', 2023, 14084.0],
     ['tata', 'tiger', 2024, 7228.0],
     ['tata', 'curvv', 2019, 15662.0],
     ['tata', 'nexon', 2024, 10843.0],
     ['tata', 'punch', 2024, 9518.0],
     ['tata', 'harrier', 2024, 22891.0],
     ['toyota', 'innova', 2023, 24096.0],
     ['toyota', 'fortuner', 2024, 39759.0],
     ['toyota', 'vellfire', 2022, 146987.0],
     ['toyota', 'camry', 2023, 29342.0],
     ['toyota', 'land cruiser', 2024, 134000.0],
     ['rolls royce', 'ghost', 2020, 113841.0]]
```

```
Python 3.10.8 Shell
C:\Users\Nitya\Downloads\Nityarthi project-1.py (3.10.8)
File Edit Format Run Options Windows Help

['toyota', 'land cruiser', 2024, 134000.0],
['rolls royce', 'ghost', 2020, 113841.0],
['rolls royce', 'phantom', 2024, 135455.0],
['rolls royce', 'cullinan', 2021, 144578.0],
['rolls royce', 'spectro', 2019, 420000.0],
['rolls royce', 'boat tail', 2021, 2800000.0],
['maruti', 'swift', 2024, 10843.0],
['maruti', 'vitara breeze', 2024, 13843.0],
['maruti', 'ertiga', 2024, 15662.0],
['maruti', 'fronx', 2024, 14785.0],
['maruti', 's-cross', 2023, 15643.0]]

def search():
    b=input('enter the brand:')
    m=input('enter the model:')
    fl=0
    brand=b.lower()
    model=m.lower()
    for i in L:
        if i[0]==brand and i[1]==model:
            print(i)
            fl=1
            x=input('if you want to know more type yes to get the link or no:')
            k=x.lower()
            if k=='yes':
                if brand=='maruti' and model=='swift':
                    print('https://www.carwale.com/maruti-suzuki-cars/swift/')
                elif brand=='maruti' and model=='vitara breeze':
                    print('https://www.carwale.com/maruti-suzuki-cars/brezza/')
                elif brand=='maruti' and model=='ertiga':
                    print('https://www.carwale.com/maruti-suzuki-cars/ertiga/')
                elif brand=='maruti' and model=='fronx':
                    print('https://www.carwale.com/maruti-suzuki-cars/fronx/')
```

VITYARTHI – PROJECT

```
G:\Vityarthi project-1\py - C:\Users\Aksh\Desktop\Vityarthi project-1\py (3.10.8)
File Edit Format Run Options Window Help

print('https://www.carwale.com/toyota-cars/innova-crysta/')
elif brand=='toyota' and model=='fortuner':
    print('https://www.carwale.com/toyota-cars/fortuner-legend')
elif brand=='toyota' and model=='vellfire':
    print('https://www.carwale.com/toyota-cars/vellfire/')
elif brand=='toyota' and model=='camry':
    print('https://www.carwale.com/toyota-cars/camry/')
elif brand=='toyota' and model=='landeruiser':
    print('https://www.carwale.com/toyota-cars/land-cruiser/')
elif brand=='rollaroyce' and model=='ghost':
    print('https://www.cardekho.com/carmodels/Rolla-Royce/Rolla-Royce_Ghost')
elif brand=='rollaroyce' and model=='phantom':
    print('https://www.cardekho.com/carmodels/Rolla-Royce/Rolla-Royce_Phantom')
elif brand=='rollaroyce' and model=='cullinan':
    print('https://www.cardekho.com/rolls-royce/cullinan')
elif brand=='rollaroyce' and model=='spectro':
    print('https://www.cardekho.com/rolls-royce/spectro')
elif brand=='rollaroyce' and model=='dawn':
    print('https://www.cardekho.com/rolls-royce/dawn')
elif brand=='tata' and model=='tiger':
    print('https://cars.tatamotors.com/tiger/tee.html')
elif brand=='tata' and model=='curvy':
    print('https://www.carwale.com/tata-cars/curvy/')
elif brand=='tata' and model=='nexon':
    print('https://www.carwale.com/tata-cars/nexon/')
elif brand=='tata' and model=='punch':
    print('https://www.cardekho.com/tata/punch')
elif brand=='tata' and model=='harrier':
    print('https://www.carwale.com/tata-cars/harrier/')
elif brand=='lamborghini' and model=='urus':
    print('https://www.carwale.com/lamborghini-cars/urus-s')
elif brand=='lamborghini' and model=='revuelto':
```

```
G:\Vityarthi project-1\py - C:\Users\Aksh\Desktop\Vityarthi project-1\py (3.10.8)
File Edit Format Run Options Window Help

print('https://www.carwale.com/lamborghini-cars/urus-s/')
elif brand=='lamborghini' and model=='revuelto':
    print('https://www.cardekho.com/lamborghini/revuelto')
elif brand=='lamborghini' and model=='huracan':
    print('https://www.carwale.com/lamborghini-cars/huracan-evo')
elif brand=='lamborghini' and model=='countach':
    print('https://www.lamborghini.com/en-en/models/limited-series/countach-lpi-800-4')
elif brand=='lamborghini' and model=='essenza':
    print('https://www.carwale.com/news/lamborghini-essenza-scv12-limited-edition-trackonly-car-breaks-cover')
elif brand=='mercedes' and model=='senna':
    print('https://silodrome.com/mercedes-benz-190e-2-3-16/')
elif brand=='mercedes' and model=='maybach':
    print('https://www.carwale.com/mercedes-benz-cars/maybach-s-class/')
elif brand=='mercedes' and model=='cabriolet':
    print('https://www.carwale.com/mercedes-benz-cars/cle-cabriolet/')
elif brand=='mercedes' and model=='limousine':
    print('https://www.carwale.com/mercedes-benz-cars/a-class-limousine/')
elif brand=='mercedes' and model=='cabriolet':
    print('https://www.carwale.com/mercedes-benz-cars/cle-cabriolet/')
elif brand=='honda' and model=='amaze':
    print('https://www.hondacarindia.com/honda-amaze')
elif brand=='honda' and model=='city':
    print('https://www.hondacarindia.com/honda-city-5th-generation')
elif brand=='honda' and model=='accordhybrid':
    print('https://www.carwale.com/honda-cars/accord/')
elif brand=='honda' and model=='civic':
    print('https://www.carwale.com/honda-cars/civic/')
elif brand=='honda' and model=='elevate':
    print('https://www.hondacarindia.com/honda-elevate')
else:
    print('link not available')

if fl==0:
```


VITYARTHI – PROJECT

```

def search_price():
    price=float(input('Enter the price lower limit in dollars:'))
    upprice=float(input('enter the price upper limit in dollars:'))
    fl=0
    for i in L:
        if float(i[3])>=price and float(i[3])<=upprice:
            print(i)
            fl=1
    if fl==0:
        print('no cars available in this year range')
def search_brand():
    b=input('Enter the brand of the car:')
    brand=b.lower()
    fl=0
    for i in L:
        if i[0]==brand:
            print(i)
            fl=1
    if fl==0:
        print('No cars available in this brand')
def search_year():
    year=int(input('Enter the year:'))
    fl=0
    for i in L:
        if int(i[2])==year:
            print(i)
            fl=1
    if fl==0:
        print('no cars available in this year')

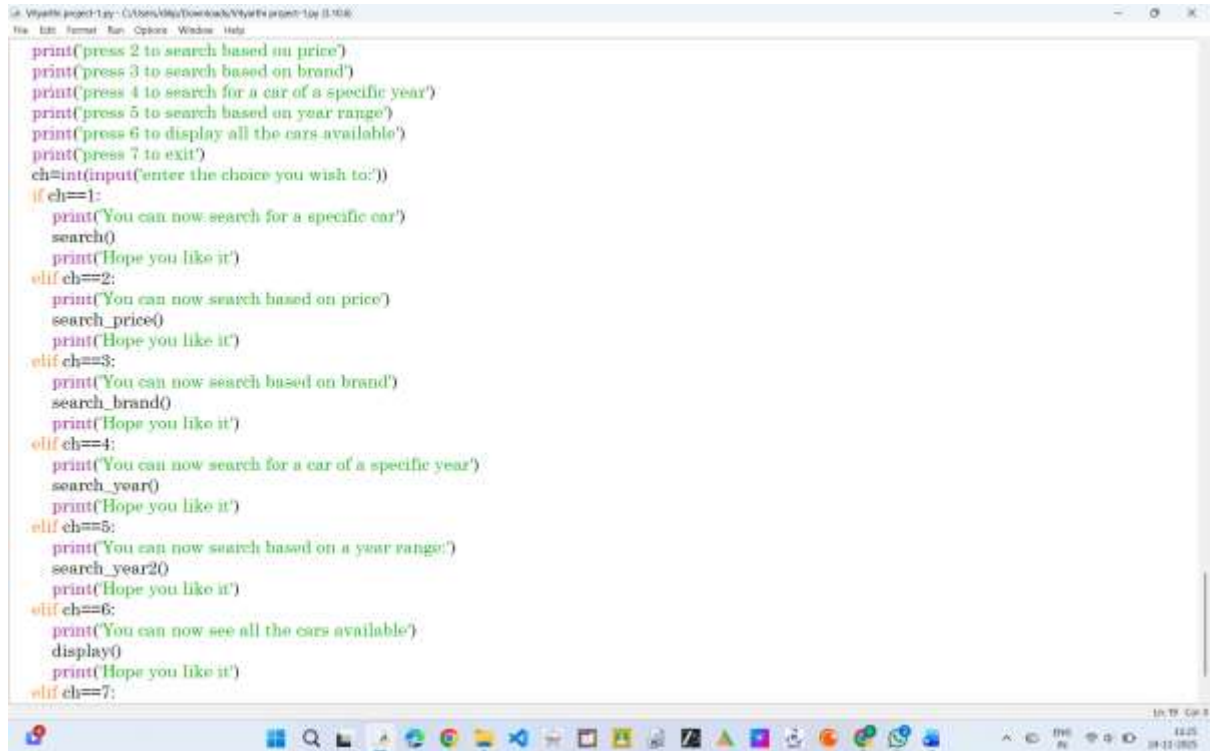
```

```

fl=0
for i in L:
    if int(i[2])>=year1 and int(i[2])<=year2:
        print(i)
        fl=1
if fl==0:
    print('no cars available in this year range')
def display():
    for i in L:
        print(i)
#main
while True:
    print('Choose any option from the menu to continue')
    print('menu')
    print('press 1 to search for a specific car')
    print('press 2 to search based on price')
    print('press 3 to search based on brand')
    print('press 4 to search for a car of a specific year')
    print('press 5 to search based on year range')
    print('press 6 to display all the cars available')
    print('press 7 to exit')
    ch=int(input('enter the choice you wish to:'))
    if ch==1:
        print('You can now search for a specific car')
        search()
        print('Hope you like it')
    elif ch==2:
        print('You can now search based on price')
        search_price()
        print('Hope you like it')
    elif ch==3:
        print('You can now search based on brand')

```

VITYARTHI – PROJECT



The screenshot shows a Python IDE window titled "Vityarthi project-1.py - C:\Users\Aksh\Desktop\Vityarthi project-1.py (3.10.8)". The code is a menu-driven program for searching cars. It uses a loop to repeatedly show a menu of options: searching by price, brand, year, year range, or displaying all cars. Each option is handled by a corresponding function (search_price, search_brand, search_year, search_year20, display) and a confirmation message "Hope you like it". The program ends when the user chooses to exit (option 7).

```
print('press 2 to search based on price')
print('press 3 to search based on brand')
print('press 4 to search for a car of a specific year')
print('press 5 to search based on year range')
print('press 6 to display all the cars available')
print('press 7 to exit')
ch=int(input('enter the choice you wish to:'))
if ch==1:
    print('You can now search for a specific car')
    search()
    print('Hope you like it')
elif ch==2:
    print('You can now search based on price')
    search_price()
    print('Hope you like it')
elif ch==3:
    print('You can now search based on brand')
    search_brand()
    print('Hope you like it')
elif ch==4:
    print('You can now search for a car of a specific year')
    search_year()
    print('Hope you like it')
elif ch==5:
    print('You can now search based on a year range')
    search_year20()
    print('Hope you like it')
elif ch==6:
    print('You can now see all the cars available')
    display()
    print('Hope you like it')
elif ch==7:
```

VITYARTHI – PROJECT

CHALLENGES FACED

Storing the details of the cars inside a List by searching for their price , year , brand , model. Thinking for the different methods to process.

LEARNING AND KEY TAKEAWAYS :

1. Implementation :

Learnt how to implement the acquired knowledge to solve the real-world problems, acquired a lot of practical skills in shape-shifting the academics into a key for locks of problem in today's world.

2. Automation value :

This would have took a lot of time and energy if it was done manually but now within seconds with the help of this calculator we can get the results automatically in a few seconds.

3. Testing :

Coding part is easy if we know the logic behind it but we can only know our code is correct when it test run it with multiple values, so I understood that testing a system is as important as designing a system.

VITYARTHI – PROJECT

4. Communication :

The system's ability of guidance has been improved and error messages are directly displayed in the output UI for optimal user support.

FUTURE ENHANCEMENTS

The system will search for the cars you wish to based on price,brand,model,year.But as of now it contains only the data of 50 cars in future the data of all cars can added and it can be used.in future by adding some more it features it can become a source for selecting the suitable car which your looking for.

As of now this code has limited features in future the data can be maximized and even many other user defined functions can be added so that it would be used for all the purposes regarding cars.

REFERENCE

1. VITYARTHI and CSE1021 course for Python coding skills.
2. Google for researching about the cars.
3. Peer and mentor feedbacks from,
 - Dr. Sivabalan KR
 - Gokul Prasad K : 25BAI10148
 - Baves P : 25BAI10093
 - Sudharshan S G : 25BAI10706

VITYARTHI – PROJECT

- Raghul S K : 25BAI10288

THANK YOU