

Python Question

All Cars Mechanical Enterprises (ACME) wishes to begin selling parts for vehicles made in the U.S. To do this, ACME engineers need to be able to search the NHTSA database for information about a vehicle when given the vehicle's VIN number. As the lead programmer for ACME, your job is to use Python to interface with NHTSA's API for decoding a VIN. You must query for a JSON result. Once you have the result you will then print all the fields and data that are returned from the query – except the fields for which the value is blank. For instance, "SeatRows" is a field that may be returned, but if "SeatRows" is returned with an empty value, do not show that field. The output should be as follows:

```
[CAR MAKE] - [CAR MODEL]
VIN# [VIN NUMBER]
-----
FIELD ONE: VALUE
FIELD TWO: VALUE
...
FINAL FIELD: VALUE
```

The system should accept the VIN number from the command-line, and the program must be called "vinLookup.py". Engineers must be able to run the program by entering either

```
python vinLookup.py [SOMEVINNUMBER|SOMEFILE]
```

or

```
./vinLookup.py [SOMEVINNUMBER|SOMEFILE]
```

If a VIN number is passed the system will perform a lookup via the API. If a filename is given, the system will read in a pickled car object from the file and print the information it contains. The system must be able to tell the difference between the two options by checking to see if the parameter passed ends in ".car". The system should be fault-tolerant – using try-except blocks for all I/O operations and should validate user input. The program should never crash; it will always let the user know if an error has occurred before exiting. If a VIN is passed that cannot possibly be valid, or any number of parameters besides one is passed, the system should alert the user and print appropriate usage information.

You must use object-oriented Python code. The main program (`vinLookup.py`) must have a main method but will then create a "Lookup" object that performs all the program's functions. You must use a "Car" object to store each VIN's information. A further requirement from the company management is that you must store the object representing each result in a file named with the convention "carmake-carmodel.car". So, if your query returned the vehicle

“Ford Pinto”, you would save the file as “ford-pinto.car”. This file will be the pickled car object, so that others in the company may make use of the object in their programs later.

Hints

- The NHTSA API is located at <https://vpic.nhtsa.dot.gov/api/Home>. It does not require authentication.
- Python has a wealth of libraries available. In particular you should probably use the JSON, requests, sys, and pickle modules, though there are others that may work as well.
- The following is a list of VIN numbers, with make and model provided so you can validate your programs.

Sample VINs

2G1WF52E559100812	Chevrolet Impala
2T1BU4EE9BC634845	Toyota Corolla
3FADP4BJ3FM162128	Ford Fiesta
1GKS2HKC2FR196167	GMC Yukon XL
1FT8W3BT7FEA87904	Ford F-350

Extra Features

Additional points will be awarded to submissions that

- Make FULL use of docstrings to comment all classes, fields, and functions.
- Create a module called cartools that includes all classes used. From this module all classes needed must be imported.
- Makes use of the getopt library to parse command-line arguments.