A2019 CarBuyerBot

Readme

Version 2.0
07/16/2020
# Table of Contents

1. **Introduction** .......................................................................................................................... 3  
   1.1 Overview ............................................................................................................................. 3  
   1.2 Common Use cases ............................................................................................................. 5  

2. **Requirements & Prerequisites** ............................................................................................ 6  
   2.1 System Requirements ......................................................................................................... 6  
   2.2 Prerequisites ....................................................................................................................... 6  
   2.3 Security Measures ............................................................................................................. 6  
   2.4 Disclaimers ........................................................................................................................ 6  

3. **Getting Started** ................................................................................................................... 7  
   3.1 Skill Matrix ......................................................................................................................... 7  
   3.2 Installation Hierarchy ......................................................................................................... 7  
   3.3 Quick Start ........................................................................................................................ 7  
   3.3.1 Setup ............................................................................................................................. 7  
   3.3.2 Configuration ............................................................................................................... 7  

4. **Reports** .................................................................................................................................. 9  

5. **Logs** ..................................................................................................................................... 10  

6. **Troubleshooting & Support** ................................................................................................ 10  
   6.1 Support ............................................................................................................................. 11  
   6.2 FAQs .................................................................................................................................. 11  

Appendix A: Record of Changes ................................................................................................. 12  
Appendix B: Acronyms ................................................................................................................ 13  
Appendix C: References .............................................................................................................. 14
1. Introduction

This document contains all essential information for users to make full use of the CarBuyerBot. This manual includes a description of the functions and capabilities as well as step-by-step procedures for the setup & configuration of the Bot.

1.1 Overview

The CarBuyerBot was created for users who are in the market for a used car for the purpose of allowing them to compare vehicle listings across model years by using a formula to determine the price per mile of the expected remaining life of a car. The output of the bot is the creation of a make/model specific csv which includes all relevant information about each matching vehicle listing.

**Price Per Remaining Mile** = Vehicle Asking Price/Expected Life of Car (in mi) – Current Mileage

Example:

I am interested in buying a Honda Civic within 500 miles of my zip code. I expect any car to last me 150,000 miles. I would fill those parameters into the variables of the CarBuyerBot and start the bot. The CarBuyerBot will fill in those details into CarGurus.com, pull back all vehicle listings matching my make/model/search radius criteria, and create a spreadsheet applying the formula for determining the price per remaining mile of the car’s useful life.

Obviously, this doesn’t account for upgrades between models (leather seats, 4 door vs 2 door, technology packages, etc) – but it gives another datapoint to support or refute a vehicle being a good deal at a given price-point/mileage.

Lets take the following listings as an example:

Car 1:

<table>
<thead>
<tr>
<th>2015 Honda Civic EX</th>
<th>GREAT DEAL</th>
<th>$1,920 BELOW</th>
<th>CarGurus IMV of $12,915</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Price: $10,995</td>
<td>Mileage: 74,690 mi</td>
<td>Location: Monroe, NC (36 mi)</td>
</tr>
<tr>
<td></td>
<td>Dealer rating:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Car 1 Price Per Remaining Mile = $10,995/(150000-74690)

Car 1 Price Per Remaining Mile = $0.1459/mile
Car 2:

![2014 Honda Civic Si](image)

**GREAT DEAL**

$1,931 BELOW CarGurus IMV of $17,926

Price: $15,995

Mileage: 36,603 mi

Location: Charlotte, NC (1.5 mi)

Dealer rating: ⭐⭐⭐⭐⭐

Car 2 Price Per Remaining Mile = $15,995/(150000-36603)

Car 2 Price Per Remaining Mile = $0.1410/mile

In our scenario, car 2 is cheaper per mile given an expected remaining life of the car. In this case car 1 happens to be an EX while Car 2 happens to be an SI, so that would have to be taken into consideration, but this bot can help quickly narrow the list of “good deals” and give data points for comparison in the form of a quickly sorted spreadsheet.
1.2 Common Use cases

The CarBuyerBot can be used to search for any make and model combination supported by CarGurus.com – it also supports all “search within distance” criteria that CarGurus.com supports – from 10mi to Nationwide.

This is especially useful because available tools for identifying a “good deal” aren’t terribly accurate. Take a look at the result below comparing 2 cars in the same geographic area of the same model year. CarGuru’s analysis of the car on line 18 was that this is a “Fair Deal”. Compare that to the car from line 45, which CarGuru’s analysis says is a Good deal. The listing on line 45 has over twice the mileage as the car from row 18 and is only $800 less expensive – assuming the $800 difference wasn’t an absolute deal breaker, it would make the most sense to buy the listing on row 18 all other things held equal.

<table>
<thead>
<tr>
<th>Year</th>
<th>Model</th>
<th>Price</th>
<th>Mileage</th>
<th>PricePerMile</th>
<th>CarGurusAnalysis</th>
<th>City</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>2016 Toyota Prius</td>
<td>17494</td>
<td>29404</td>
<td>0.145062854</td>
<td>GOOD DEAL</td>
<td>Charlotte NC</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>2016 Toyota Prius</td>
<td>19598</td>
<td>17501</td>
<td>0.148022266</td>
<td>FAIR DEAL</td>
<td>Statesville NC</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2016 Toyota Prius</td>
<td>18998</td>
<td>22178</td>
<td>0.14868562</td>
<td>FAIR DEAL</td>
<td>Gastonia NC</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>2016 Toyota Prius</td>
<td>20959</td>
<td>20736</td>
<td>0.162141043</td>
<td>HIGH PRICE</td>
<td>Los Angeles CA</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>2016 Toyota Prius</td>
<td>19299</td>
<td>31723</td>
<td>0.163167818</td>
<td>HIGH PRICE</td>
<td>Los Angeles CA</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>2016 Toyota Prius</td>
<td>22300</td>
<td>16128</td>
<td>0.16577029</td>
<td>HIGH PRICE</td>
<td>Chicago IL</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>2016 Toyota Prius</td>
<td>22300</td>
<td>18778</td>
<td>0.169941016</td>
<td>OVERPRICED</td>
<td>Temple Hill MD</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>2016 Toyota Prius</td>
<td>22259</td>
<td>22208</td>
<td>0.174181482</td>
<td>FAIR DEAL</td>
<td>Stafford TX</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>2016 Toyota Prius</td>
<td>18188</td>
<td>46571</td>
<td>0.1758501</td>
<td>GOOD DEAL</td>
<td>Tarboro NC</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>2016 Toyota Prius</td>
<td>21000</td>
<td>31710</td>
<td>0.1775298</td>
<td>HIGH PRICE</td>
<td>Saint Louis MO</td>
<td></td>
</tr>
</tbody>
</table>
2. Requirements & Prerequisites

2.1 System Requirements
Windows 64-bit and 32-bit versions are supported. No specific hardware requirements outside of those requirements needed for the installation of the A2019 Bot Agent

2.2 Prerequisites

2.3 Security Measures
No specific Security Measures required as this bot utilizes a publicly available website and writes to a locally hosted csv.

2.4 Disclaimers
Any decision on buying a car should be taken seriously based on many factors...vehicle history report, budget, reputation of seller, model reliability, etc – this bot is designed to give car buyers another interesting data point to consider with evaluating vehicle listings.
3. Getting Started

3.1 Skill Matrix

There is truly only a single bot file which is used for the processing. Fill in the required parameters on lines 3-7 of the bot and let it run. The bot will produce a .csv file which will automatically be saved to the C:\temp directory of your local machine – named after the make, model, and date that you executed the bot run.

Note that the bot uses the standard Bot Store template which is located on the Automation Anywhere GitHub page.

3.2 Installation Hierarchy

In Automation Anywhere Enterprise, the bot installs to the Bot Store parent directory into a subfolder named CarBuyerBot-Automation Anywhere. When installed in community edition, the bot installs into the Bots directory, creating a subfolder named CarBuyerBot-AutomationAnywhere with the CarBuyerBot in said folder.

3.3 Quick Start

The setup for the CarBuyerBot is very straightforward as there are only a few criteria which need to be set to enable the search to take place. Fill in the required parameters on lines 3-7 of the bot and verify that the account being used to execute the bot has permissions to write to the C:\temp directory of your local machine.

3.3.1 Setup

No external application setup required. All configurations handled through setting variables directly in the bot setup.

3.3.2 Configuration

The following section is a breakdown of the different variables required to be set on lines 3-7 of the bot:

**Make/Model** Can be any make/model supported by CarGurus.com. Attempt to use the site manually once to make note of how the make/model are displayed in their dropdowns. The values in Input.csv must match the CarGuru.com dropdown values exactly.

**Zip Search** Should be the zip code to search for. If you're searching nationwide this doesn't matter as much, but a zip code is still required.

**Search Radius** supports all search criteria supported by CarGurus.com. This includes the values: 10 mi, 25 mi, 50 mi, 75 mi, 100 mi, 150 mi, 200 mi, 500 mi, or Nationwide. Please make sure that the search radius value is formatted exactly as above. There should be a space between the number and the mi.

Example:
<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Setup Values</td>
</tr>
<tr>
<td>3</td>
<td>String: Assign &quot;Toyota&quot; to $sMake$</td>
</tr>
<tr>
<td>4</td>
<td>String: Assign &quot;Sienna&quot; to $sModel$</td>
</tr>
<tr>
<td>5</td>
<td>String: Assign &quot;28031&quot; to $sZip$</td>
</tr>
<tr>
<td>6</td>
<td>Comment: &quot;10 mi, 25 mi, 50 mi, 75 mi, 100 mi, 150 mi, 200 mi, 500 mi, or Nationwide&quot;</td>
</tr>
<tr>
<td>7</td>
<td>String: Assign &quot;50 mi&quot; to $sSearchRadius$</td>
</tr>
</tbody>
</table>
4. Reports

The output created by this bot is in the c:\temp directory of your local machine, named after the make_model_zip_date.

Example:

Toyota_Sienna_28031_2020-07-16.csv

The contents of the csv include the year, model, Trim, Color, price, mileage, calculated price per mile, CarGurus.com deal analysis, city, state, and URL. These fields should allow any user to make informed decisions on comparing, sorting, and filtering on different vehicle listings.
5. Logs

On error the bot will log to the C:\ProgramData\AutomationAnywhere\Bots\Logs\CarBuyerBot-Automation Anywhere directory. Here you can find logging for errors as well as Snapshots of the screen when the bot encountered an issue.
6. Troubleshooting & Support

6.1 Support

Should you run into any issues with your bot executing double check the following:

1. Make sure that the values match possible options in the CarGurus.com drop downs for make/model.
   
   a. Additionally, make sure that each of the fields are not prefixed with spaces.

2. Should the bot not run correctly, make sure to check the error folder, logs and screenshots subfolders to see where the bot may have had an issue.

3. If the bot is run too many times back to back – CarGurus.com seems to send bad data back to the calling REST call…wait a couple hours and try running it again, or hop on to a VPN so your traffic is coming from a different IP.

6.2 FAQs
# Appendix A: Record of Changes

<table>
<thead>
<tr>
<th>No.</th>
<th>Version Number</th>
<th>Date of Change</th>
<th>Author</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>1.0</td>
<td>07/16/2020</td>
<td>Micah Smith</td>
<td>Initial release of the CarBuyerBot. Currently only supports CarGurus.com</td>
</tr>
</tbody>
</table>
## Appendix B: Acronyms

<table>
<thead>
<tr>
<th>No.</th>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
</table>


## Appendix C: References

<table>
<thead>
<tr>
<th>No.</th>
<th>Topic</th>
<th>Reference Link</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>