

**Analytix SOP**

**Interview Technical Test for**

**Sr. Developer – Automation**

**Document Control**

|  |  |
| --- | --- |
| **Document Name:** | Interview Technical Test for Senior Developer - Automation |
| **Brief Description:** | Technical test for Webscraping |
| **Classification:** | Automation |
| **Policy/Document Approver:** | Anish Pillai |
| **Policy/Document Owner:** | Deepak Soni |
| **Current Version:** | 1.0 |
| **First Document Release Date:** | 24th January 2024 |

**Version Control**

|  |  |  |
| --- | --- | --- |
| Version | Date | Brief Description of Change |
| 1.0 | 1/24/2024 | First Release |
|  |  |  |
|  |  |  |
|  |  |  |

# Contents

[Contents 3](#_Toc157008703)

[Test Description 4](#_Toc157008704)

[Time Limit 4](#_Toc157008705)

[Process 4](#_Toc157008706)

[Output 5](#_Toc157008707)

[Pre-Requisites 5](#_Toc157008708)

[User Interface: 5](#_Toc157008709)

[Important Notes: 5](#_Toc157008710)

[Results: 6](#_Toc157008711)

# Test Description

A client wants to scrape car auction details from <https://bringatrailer.com/convertible/>. The data should be saved in a csv file daily so the client can analyze it later.

For the purpose of this test, limit the total number of cars to the first 10 in the auction list.

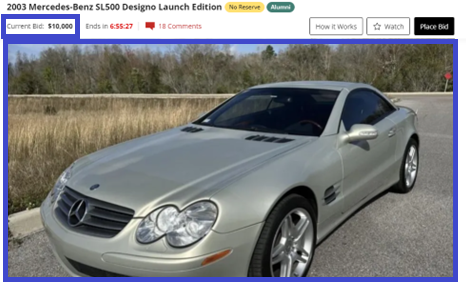
# Time Limit

Time limit for this test is 2 hours.

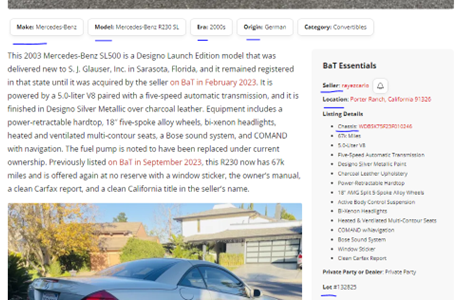
# Process

Below are steps to automate:

1. Go to <https://bringatrailer.com/convertible/>. Login is not required for this page.
2. Capture the Current Bid price and save the picture for each car. More details about picture mentioned later.



1. Navigate to each car auction page and get the following details of car mentioned below:



* Car Name
* Make
* Model
* Era
* Origin
* Seller
* Location
* Chassis number
* Lot number

Save the above details to a csv file. (Sample output format is provided along with this file **BringATrailer\_CarAuctionData\_2024-01-24\_Sample\_Output.csv**)

1. Save the main picture of the car in “Downloads” folder of the current directory. Filename should be the car’s auction “lot number”.

A sample output is provided for your convenience.

# Output

1. Data should be saved to a csv file with the following naming convention:

**“BringATrailer\_CarAuctionData\_yyyy-mm-dd.csv”**

1. Pictures should be saved a folder named “Pictures” in the same path as the project path.

# Pre-Requisites

1. Please go through the style guidelines given in the link [PEP 8 – Style Guide for Python Code | peps.python.org](https://peps.python.org/pep-0008/)
2. You may use selenium, requests, scrapy or any other library as required to scrape the data.

# User Interface:

User interface is not required for this test.

# Important Notes:

Below are the criteria for judging this test:

|  |  |
| --- | --- |
| **Test Points** | **Description** |
| Coding Standards | Follow coding guidelines for Class, functions, variables, module names etc. as mentioned in <https://peps.python.org/pep-0008/> |
| Comments | Provide appropriate comments in your code wherever required. |
| Correct Results | Code provides correct results as per client’s requirements. |
| Code Readability | Classes, functions, variables and modules names should be easily identifiable and understandable. |
| Enhancements | Bonus points for creating separate classes or functions as required. |

# Results:

Commit your code and output file in repository and share the link.

*---End of File---*