# Article information:

用undetected\_chromedriver代替selenium解决浏览器打不开网页\_Scott0902的博客-CSDN博客\_undetected\_chromedriver  
<https://blog.csdn.net/Scott0902/article/details/127024380>

# Article summary:

1. The article discusses various methods of opening webpages using Python crawler code, such as requests and selenium.

2. It explains how many websites have anti-crawling mechanisms that make it difficult to open the page with the traditional methods.

3. The article suggests using undetected\_chromedriver instead of selenium to solve the issue of browsers not being able to open webpages.

# Article rating:

May be slightly imbalanced: The article presents the information in a generally reliable way, but there are minor points of consideration that could be explored further or claims that are not fully backed by appropriate evidence. Some perspectives may also be omitted, and you are encouraged to use the research topics section to explore the topic further.

# Article analysis:

The article is generally reliable and trustworthy in its discussion of various methods for opening webpages using Python crawler code, such as requests and selenium. The author provides detailed explanations of each method, including code examples, which makes it easier for readers to understand the concepts discussed in the article.

However, there are some potential biases in the article that should be noted. For example, while the author does mention Firefox as an alternative to Chrome, they do not provide any details or code examples for this method, which could lead readers to believe that Firefox is not a viable option for solving this problem. Additionally, while the author does mention possible risks associated with using undetected\_chromedriver (such as potential security issues), they do not provide any further information or advice on how to mitigate these risks.

In conclusion, while this article is generally reliable and trustworthy in its discussion of various methods for opening webpages using Python crawler code, there are some potential biases that should be noted when reading it.

# Topics for further research:

* Firefox web scraping Python
* Selenium Python web scraping
* Security risks of undetected\_chromedriver
* Python requests library
* Python web crawler code
* Python web scraping best practices

# Report location:

<https://www.fullpicture.app/item/51239776e6d92bfb962465bec7907d46>