

## HodgBarr Ski buyers guide

A web scraping tool helping you purchase skis

CS021: Computer Programming I

Fall 2020

Group Members:

Owein Labarr & Colin Hodge

(olabarr & cehodge)

College of Engineering and Mathematical Sciences

The University of Vermont

December 4, 2020

1

---

<sup>1</sup>Owein and Colin would like to extend our thanks to the people at evo.com, without them this program would not work.

# 1 Front-End & The Quiz

The front-end of the program consists of three buyers guides. When the program is run, the user will be prompted to enter a number corresponding to the buyers guide they would like to use.

The first guide is a set of functions designed to ask the user questions on their gender, skill level, desired ski type, and desired ski profile. Information on these specifications is also included. The program then provides the user with the top three skis that meet their specifications using the scraped and formatted dictionary copied from the back-end program.

Since ski boots size does not correlate to shoe size, the second guide converts the users shoe size to the recommended boot size. In the industry, it is common to see a comfort and performance fit size for any given shoe size, therefore both sizes are returned to the user with a brief description regarding the difference between the two.

The third guide assists the user in determining the recommended ski length based on their height. The user will be prompted to enter their height in return for predetermined range of ski lengths in centimeters. The recommended lengths fit within industry standards, and will correlate to the lengths displayed on evo.com or any other ski website.

# 2 Back-End & Selenium

The back-end program was used to create a dictionary of ski names and their respective links organized in the order they were asked about in the front end quiz. To access the internet a collection of open sourced APIs were used to open and scrape webpages, this is known as Selenium WebDriver.

The back-end utilized the predictability of the links on evo.com to concatenate strings of the URLs for each individual option in the quiz. Each link started and ended with the same text, and as the filters were applied to the webpage, text was added in so that the link would take you to that appropriate page. This was done instead of manually entering the URLs in the dictionary because there were 96 possible answers to the quiz, and for each answer the top 3 most compatible skis were shown, adding up to 288 specific links. Doing this manually would lead to ease of making a mistake within the syntax of the code.

As the program creates the links to the pages selenium opens a chrome browser using those links. Selenium then searches for the title of the skis and the respective link by looking on the page for their class names and creates 2 lists, the two individual lists are then zipped together. A dynamic dictionary is then created using embedded four loops to organize the dictionary based on the order the questions are asked in the quizzes.

### 3 Why this Program?

Owein and Colin are avid skiers and understand the complexity of the ski industry as well as how overwhelming purchasing a new pair of skis can be. There are so many things to consider such as correct sizing as well as getting the correct pair for the type of skiing you will be doing. The creators wanted to build something that was a "One Stop Shop" that new and old skiers alike could use to help them purchase their next pair of "planks".

### 4 Bibliography

#### Bibliography

B. Muthukadan, "Python zip()," *Programiz*, 2018. [Online]. Available:

<https://www.programiz.com/python-programming/methods/built-in/zip>. [Accessed: 04-Dec-2020].

Evo, "Mountain, Street & Water: Community, Culture & Giving Back," *evo*, 2020. [Online].

Available: <https://www.evo.com/>. [Accessed: 04-Dec-2020].

Parewa Labs, "Python zip()," *Programiz*. [Online]. Available:

<https://www.programiz.com/python-programming/methods/built-in/zip>. [Accessed: 04-Dec-2020].