Vidzeme University of Applied Sciences

**Faculty of Engineering**

Introduction to Python and Data Exploration

group 08

practical work #5

Valmiera, 2023

CONTENT

1 Code Structure 4

2 Error handling 4

List of used resources 5

APPENDIX 1 6

|  |  |  |  |
| --- | --- | --- | --- |
| Document versions | | | |
| Version | Status / Changes | Date | Author |
| 1.0 | Made the code and documentation for Task 2 | 15.03.2023 | Ričards Didriksons |

|  |  |  |  |
| --- | --- | --- | --- |
| **Contacts and responsible (-s)** | | | |
| Name Surname | Department | Position | Contact information (e-mail) |
| Rainers Krūmiņš | Group #8 | Coordinator | rainers.krumins@va.lv |
| Ričards Didriksons | Group #8 | Member | ricards.didriksons@va.lv |
| Gatis Stūrītis | Group #8 | Member | gatis.sturitis@va.lv |

**GOAL.** Make a program that reads from imdb.com ‘Blizzard of Souls’ HTML document the director, top stars, writers, awards and full list of cast using package BeautifulSoup.

# Code Structure

* **Graphical user interface, text

  Description automatically generated with medium confidence**Imported required packages – BeautifulSoup, Regex, and urllib.request.
* Opening the selected URL and parsing the HTML document as ‘html.parser’ into beautifulsoup. Header ‘User-Agent’ was changed in order to tell the server that this is not a python script and to avoid 403 error.

Text

Description automatically generated

* 4 functions:
* Prints the title of the movie which is under <h1> tag within the HTML document.

Text

Description automatically generated with medium confidence

* Prints the ‘top credits’ of the movie (director, writers, stars). The extracted information is located within 3 <li> tags within a single <div> which contains keyphrase ‘principal-credits’.

Graphical user interface, text

Description automatically generated

* Prints the number of awards won and nominated. The text is located within a single <span> tag, which is under a <li> tag that contains keyphrase ‘awards\_information’.

Text, Word

Description automatically generated with medium confidence

* Prints all members of the cast and their adjacent roles in the movie. Information about every actor is located within individual <div> tags which are located under a single main <div> which contains the keyphrase ‘title-cast\_\_grid’. The individual <div>’s contain role and actor name information within their respective <a> tags.

Graphical user interface, text, application

Description automatically generated

# Error handling

Within ‘get\_cast()’ function it is possible that an actor does not have a role, which is handled by using a ‘try:, catch:’ block. If an actor does not have a role, then no role will be printed next to their name, iterator still has to be incremented regardless of the situation.

**Graphical user interface, text

Description automatically generated**

When get\_awards() is executed on a movie webpage that has not yet been nominated for any award the function throws an error. That is handled here, and information ‘! No nominations yet !’ is printed.

**Graphical user interface, text, application

Description automatically generated**

# List of used resources

Introduction to Python and Data Exploration - Dr.sc.ing., Kaspars Osis

# APPENDIX 1

\_pyt2023\_group08\_pw4.zip

\_pyt2023\_group08\_pw4\_rainers\_krumins.zip

\_pyt2023\_group08\_pw4\_ricards\_didriksons.zip