Project Specification

# Description

Design a computer program, written in Python 3, that reads world happiness index data from the provided csv file, and allows exploration of that data by the user. I’m using **PyCharm Community Edition 2023.2.3**

# Requirements

To be a successful project, your program must:

* Read sales data from a CSV (comma-separated values) file called “world\_happiness\_index\_2013\_2023.csv” and load it into an appropriate data structure(s):

**–** About Dataset

Originally Taken from <http://worldhappiness.report/> (Collaborator: Joakim Arvidsson). This work is licensed under the Community Data License Agreement - Permissive - Version 1.0.

https://www.kaggle.com/datasets/joebeachcapital/world-happiness-report-2013-2023

* The user is presented with a menu that has the following options to perform his/her query:
  1. The top 3 to 10 happiest countries for a certain year. (eg. 2015), as well as an option for the 3 least happiest countries for that year. The latter is optional but should be provided to the user as a choice.
  2. The top 3 countries that have gotten 1st in the happiness index for all the data provided the most number of times, as well as the 3 unhappiest countries based on happiness index the most number of times.
  3. Choose a country and show for a period (based on user’s choice, last 3, 5 or 10 years) whether the country has been increasing its rank or decreasing its rank.
  4. List the countries by name sorted in alphabetical order in ascending order/descending order. The default is ascending order, the descending order is a choice.
  5. Choose countries with at least a certain index. The user can opt for a higher end, but it is optional. List these countries in descending order according to index.
  6. Group countries by rank ranges in multiples of 10 (1 to 10, 11 to 20..) for the last 5 years in ascending order by rank.
  7. List the countries that have at least 3 consecutive years of lower ranks for any frame of time. (Think about 2014, how do you handle that?)
  8. For all options stated above, the user can choose a specific country from the filtered list and drill into the details such as:
     + the average rank, the range of its rank,
     + the range of its indexes,
     + the standard deviation of its indexes.
     + the years which the country has gotten the lowest rank and the highest rank.
* Be easy to use, verbose.
* Use modular program design.
* Employ sensible naming of variables, functions, modules, etc;
* Employ extensive (but clear) use of comments.
* Use only the following packages (and only if required):
* math;
* sys;
* os.
* Use only data types seen in the lectures (no dictionaries, classes, etc).

**Cannot use AI tools- like ChatGPT, no pandas, no lambda function.**

## Extra Functionality

To increase the quality of your project, several of the following extras can be attempted. This list is not definite, and any extra features (not listed here) that will add value to the program, and thus will improve the final evaluation of your work.

The program can also:

* + - Allow choice of input data file name;
    - Be robust (should not crash with unexpected input data):
      * type detection;
      * wrong menu options;
      * wrong filename;
      * . . .
    - Provide other statistical measures;
    - Provide a way to save a data analysis overview as a separate text file;
    - Provide a pleasant interface.

# Deliverables

The following deliverables are required (2 PDF files and all Python code):

* A User Manual (PDF file), containing instructions on how to use the software;
* a Code Manual (PDF file), containing:
  + An upper level UML Activity Diagram, describing the main functionality and flow control of the program;
  + A full print-out of the commented code.
* Source code (one or more Python files).