Analyze International Debt Statistics

The only prerequisite to complete this project is familiarity with the contents covered in DataCamp's **Introduction to SQL** course.

SQL DataCamp projects are completed in Jupyter Notebooks. If you're not familiar with Jupyter Notebooks, that's okay! All you need to know is that you can execute SQL commands in the code cells provided, as long as you have `%%sql` at the top of them. If you'd like more info on Jupyter Notebooks, go **here**.

If you experience odd behavior you can reset the project by clicking the circular arrow in the bottom-right corner of the screen. Resetting the project will discard all code you have written so be sure to save it offline first.

The Notebook has been uploaded for this project. It is noted that the project is completed in mySQL, you can fork and clone the project and make a PR for notebook in any other language.

List of tasks for the Project:

Task 1: Instructions

Inspect the international debt data.

- Read the line of code provided for you, which connects you to the `international_debt` database.
- Select all of the columns from the `international_debt` table and limit the output to the first 10 rows.

Task 2: Instructions

Find the number of distinct countries.

- Use the `DISTINCT` clause and the `COUNT()` function in pair on the `country_name` column.
- Alias the resulting column as `total_distinct_countries`.

Task 3: Instructions

Extract the unique debt indicators in the table.

- Use the `DISTINCT` clause on the `indicator_code` column.
- Alias the resulting column as `distinct_debt_indicators`.

- Order the results by `distinct_debt_indicators`.

Task 4: Instructions

Find out the total amount of debt as reflected in the table.

- Use the built-in `SUM` function on the `debt` column, then divide it by 1000000 and round the result to 2 decimal places so that the output is fathomable.
- Alias the resulting column as `total_debt`.

Task 5: Instructions

Find out the country owing to the highest debt.

- Select the `country_name` and `debt` columns, then apply the `SUM` function on the `debt` column.
- Alias the column resulted from the summation as `total_debt`.
- `GROUP` the results `BY` `country_name` and `ORDER` them `BY` the new alias `total_debt` in a *descending* manner.
- `LIMIT` the number of rows to be *one*.

Task 6: Instructions

Determine the average amount of debt owed across the categories.

- Select `indicator_code` aliased as `debt_indicator`, then select `indicator_name` and `debt`.
- Apply an aggregate function on the `debt` column to average out its values and alias it as `average_debt`.
- Group the results by the newly created `debt_indicator` and already present `indicator_name` columns.
- Sort the output with respect to the `average_debt` column in a descending manner and limit the results to *ten*.

Task 7: Instructions

Find out the country with the highest amount of principal repayments.

- Select the `country_name` and `indicator_name` columns.
- Add a `WHERE` clause to filter out the maximum `debt` in `DT.AMT.DLXF.CD` category.

Task 8: Instructions

Find out the debt indicator that appears most frequently.

- Select the `indicator_code` column, then separately apply an aggregate function to count its values. Alias the column resulting from the counting as `indicator_count`.
- Group the results by `indicator_code` and order them first by the newly created `indicator_count` column then the `indicator_code` column, both in a descending manner.
- Limit the resulting number of rows to 20.

Task 9: Instructions

Get the maximum amount of debt that each country owes.

- Select the `country_name` and apply an aggregate function to take the maximum of `debt`. Alias the aggregate column as `maximum_debt`.
- Group the results by `country_name`.
- Order the results by `maximum_debt` in descending order.
- Limit the output to 10 rows.