# Take Home Test

This test is for us to gauge your data wrangling and visualization skills in a context similar to a typical project executed within our team.

The tables needed to complete the tasks are located in the data/ folder and their columns definitions are in appendix of this document. Please note that the data is completely synthetic, and so the numerical values have no insights to analyze.

Please attach any script/notebook that you used to complete the tasks, in the language of your choice (R, python, SQL, ..).

You can present your ideas or proposition in whatever format you choose (Word, Google Docs, Google Slides, PPT, PDF, etc.).

You can use any external tools and resources (Google, …) that you need to answer.

You may also use a chatbot assistant, whether it is to generate code or query information. If you do use one, we are interested to know the process you took when working with it. Out of fairness for every candidate, please restrict yourself to free options available on the market.

Different LLMs can be accessed for free here : <https://labs.perplexity.ai/>

Task 1

In our data pipeline, the tables companies\_activities and companies\_emissions are processed together in order to obtain the assets table. The companies\_\* tables are the raw input that we receive from a third-party data provider, and the assets table is its processed version that is the input to our stress test model. Another table (not part of this test) contains the production forecasts for different industries for different regions of the world, or for the entire world (“Global” geography in the tables). Those 2 tables are used in input of the model, to produce stress test results.

Please list the transformations you would apply to tables **companies\_activities** and **companies\_emissions**, in order to obtain a table equivalent to the **assets** table.

You can share any script/notebook that you used for an exploratory analysis. You do not need to implement the cleaning, but rather describe the steps (and their order) you would take to create the **assets** table.

* Missing values (NAs) in the equity ownership column, need to fill them in
  + Some rows have always empty production
* Power sector has production in MW, and emissions in MWh/tCO2, meaning that emissions and productions datasets need to be aligned on the units before merging
* Need to pivot the equity ownership columns to have the years as rows
* Need to aggregate the countries at the global level
* Years as columns must be pivoted to rows
* Removes years prior to 2024
* Since only 1 geography appears here (Global), all productions of a given company are aggregated over all countries where this company produces
* Plan tech prod : aggregate on all columns except ald\_sector and ald\_location, and sum the equity column
* plan\_emission\_factor : Aggregate the same, but on 1 dataframe sum the company emission, and in another sum the company activities. Then, merge the two dataframes and divide activities by emissions.

Task 2

The **stress\_test\_results** table contains the output of a run of the stress test model for the companies listed in the assets table. Using this and the **portfolio** table, please draw a plot showing the Expected Loss. It should be possible to change the x-axis variable, to display the EL per ald\_sector, or the EL per ald\_business\_unit.

Tip:

* If you’re using R for this task, a way to do the parametrization is to use symbols. Strings can be interpreted as symbols using the syntax: *!!rlang::sym(“variable\_name”)*
* The Expected loss : lgd \* pd\_shock \* exposure
* A barplot
* The results are computed at company level, then aggregated by sum at the ald\_sector level
* A parametrization of the x-axis that allows to switching between ald\_sector and ald\_business\_unit
* Parametrization is thought in advance to be able to add other granularities with minimal change to the function

Task 3 (optional)

The file **dashboard.R** contains a template for a basic shiny app. Incorporate the EL plot from before into the template, using a mechanism of your choice (such as a button or dropdown menu) to allow for changing the variable that determines the x-axis ticks.

Tip:

* Learn about shiny’s UI&Server framework : <https://info201.github.io/shiny.html>
* Learn about shiny’s reactivity concept : <https://mastering-shiny.org/basic-reactivity.html>
* A shiny app displaying the plot with a UI element that switches the x-axis

# Appendix

* Table companies\_activities

Production of different carbon-intensive industries for a set of companies, located in different countries.

|  |  |
| --- | --- |
| company\_id | Id of the company |
| company\_name | Name of the company |
| ald\_sector | Sector of production |
| ald\_business\_unit | Sub-sector of production |
| activity\_unit | Unit of production |
| ald\_location | Country where the company operates its production. In iso2c format. |
| Equity Ownership 2022 | Production in 2022 |
| Equity Ownership 2023 | Production in 2023 |
| Equity Ownership 2024 | Production in 2024 |
| Equity Ownership 2025 | Production in 2025 |
| Equity Ownership 2026 | Production in 2026 |

* Table companies\_emissions

Greenhouse gas emissions, resulting of the production presented in the companies\_production table.

|  |  |
| --- | --- |
| company\_id | Id of the company |
| company\_name | Name of the company |
| ald\_sector | Sector producing the greenhouse emissions |
| ald\_business\_unit | Sub-sector producing the greenhouse emissions |
| activity\_unit | Unit of emissions |
| ald\_location | Country where the emissions emanate from. |
| Equity Ownership 2022 | Emissions from production in 2022 |
| Equity Ownership 2023 | Emissions from production in 2023 |
| Equity Ownership 2024 | Emissions from production in 2024 |
| Equity Ownership 2025 | Emissions from production in 2025 |
| Equity Ownership 2026 | Emissions from production in 2026 |

* Assets

|  |  |
| --- | --- |
| year | Year of production |
| company\_id | Id of the company |
| company\_name | Name of the company |
| ald\_sector | Sector of emissions |
| ald\_business\_unit | Sub-sector of emissions |
| ald\_production\_unit | Unit of production |
| emissions\_factor\_unit | Unit of emissions |
| scenario\_geography | Region of the world where the company produces. Matches a climate scenario’s geography. |
| plan\_tech\_prod | Planned production |
| plan\_emission\_factor | Planned emission factor. Computed as production/emissions |

* stress\_test\_results

|  |  |
| --- | --- |
| company\_id | Id of the company |
| company\_name | Name of the company |
| ald\_sector | Sector of production |
| ald\_business\_unit | Sub-sector of production |
| scenario\_geography | Geography for which the results have been computed |
| baseline\_scenario | Baseline production scenario |
| shock\_scenario | Shock production scenario |
| net\_present\_value\_baseline | Net Present Value in the baseline scenario |
| net\_present\_value\_shock | Net Present Value in the shock scenario |
| net\_present\_value\_difference | Difference in npv\_shock – npv\_baseline |
| shock\_year | Year at which the transition shock between the baseline and shock scenario is applied |
| term | Year at which the PD has been computed, that corresponds to the expiration date of a Loan. |
| pd\_baseline | Probability of Default in the baseline scenario |
| pd\_shock | Probability of Default in the shock scenario |
| pd\_difference | Difference in pd\_shock – pd\_baseline |
| run\_id | ID of the stress test run |