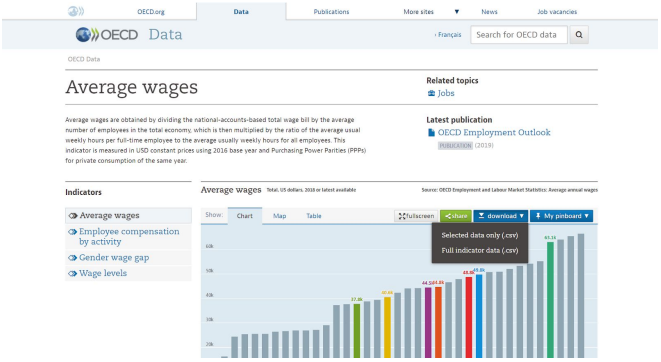
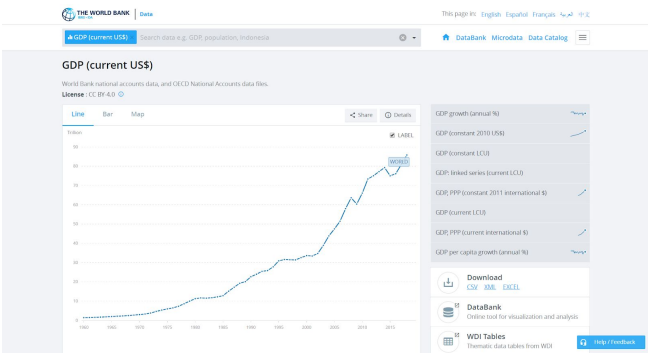
ETL Project (Steve Kinsella, David Lin, Neil Mudjer)

- **Extract: your original data sources and how the data was formatted (CSV, JSON, pgAdmin 4, etc).**
    - Original data source:
        - Source 1:
            - https://data.oecd.org/earnwage/average-wages.htm
            - 

            - Use the “Full indicator data” (“**avg_wages**”)
        - Source 2:
            - https://data.worldbank.org/indicator/ny.gdp.mktp.cd
            - 

            - Use the “CSV” file under “Download” (“**gdp**”)

- **Transform: what data cleaning or transformation was required.**
  - **The type of transformation needed for this data (cleaning, joining, filtering, aggregating, etc).**
    - Cleaning: we renamed columns to allow easier join later in SQL
    - Create a table of code and country name as the **country_code** table and set index as **c_id**

	c_id	country	code
0	1	Aruba	ABW
1	2	Afghanistan	AFG
2	3	Angola	AGO
3	4	Albania	ALB
4	5	Andorra	AND

    - Melt the data frame:
      - The **wage** dataframe was structured such that each year / country combination was a row.

```
wage_df.head()
```

	LOCATION	INDICATOR	SUBJECT	MEASURE	FREQUENCY	TIME	Value	Flag Codes
0	AUS	AVWAGE	TOT	USD	A	1990	40102.173279	NaN
1	AUS	AVWAGE	TOT	USD	A	1991	39911.210288	NaN
2	AUS	AVWAGE	TOT	USD	A	1992	40712.373962	NaN
3	AUS	AVWAGE	TOT	USD	A	1993	41087.155064	NaN
4	AUS	AVWAGE	TOT	USD	A	1994	41365.722620	NaN

      - The **gdp** dataframe was structured such that each year was a column within the country row

```
gdp_df.head()
```

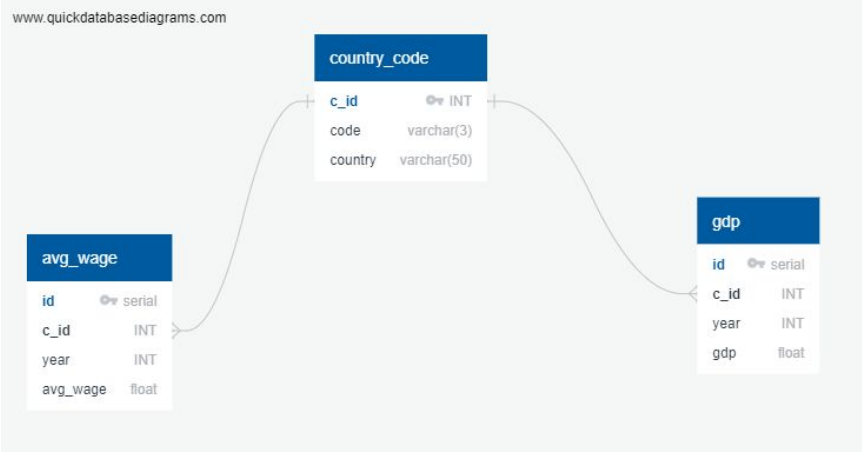
	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	
0	Aruba	ABW	GDP (current US$)	NY.GDP.MKTP.CD	NaN	NaN	
1	Afghanistan	AFG	GDP (current US$)	NY.GDP.MKTP.CD	537777811.1	548888895.6	54
2	Angola	AGO	GDP (current US$)	NY.GDP.MKTP.CD	NaN	NaN	
3	Albania	ALB	GDP (current US$)	NY.GDP.MKTP.CD	NaN	NaN	
4	Andorra	AND	GDP (current US$)	NY.GDP.MKTP.CD	NaN	NaN	

- We used Pandas.melt to rearrange the **gdp** dataframe to match the wages dataframe so that each country / year combination is a it's own row

```
gdp_df.head()
```

	code	country	year	gdp
0	ABW	Aruba	1960	NaN
264	ABW	Aruba	1961	NaN
528	ABW	Aruba	1962	NaN
792	ABW	Aruba	1963	NaN
1056	ABW	Aruba	1964	NaN

- **Load: the final database, tables/collections, and why this was chosen.**
  - We loaded the **country_code**, **avg_wage**, and **gdp** tables separately into SQL
    - We chose the SQL data structure over Mongo because the data was already in a tabular format, and we did not feel a need for Mongo's flexibility
  - Setting things up in SQL
    - The **avg_wage** and the **gdp** tables were transformed to replace the code and country name with **c_id**. The tables were linked via key **c_id**.



  - **Joining**: we did not join in python. this was done as part of SQL query,
    - e.g.,

```
SELECT *
FROM country_code
JOIN avg_wage ON country_code.c_id = avg_wage.c_id
JOIN gdp ON country_code.c_id = gdp.c_id
```

- **Filtering**: this was also in SQL query, e.g., year,
  - e.g.,

```
SELECT country_code.country, country_code.code, avg_wage.year,
gdp, avg_wage, ((avg_wage/gdp)*100) AS div
FROM country_code
JOIN avg_wage ON country_code.c_id = avg_wage.c_id
JOIN gdp ON country_code.c_id = gdp.c_id
WHERE
gdp.year = avg_wage.year
and
gdp IS NOT NULL
and
avg_wage <= 45000
and
gdp.year = 1990
ORDER BY div desc;
```

- We chose these data because we wanted to see if there's a relationship between the success of the country (as indicated by GDP) and success of the average person (as indicated by average wage)