Nobel Prize

Laureates

By: Steve Bonillas, Sequoia Boubion-Mckay, Rania Shaker,
Garima Chaudhary , Angel Alcala Ruiz

OBJECTIVES

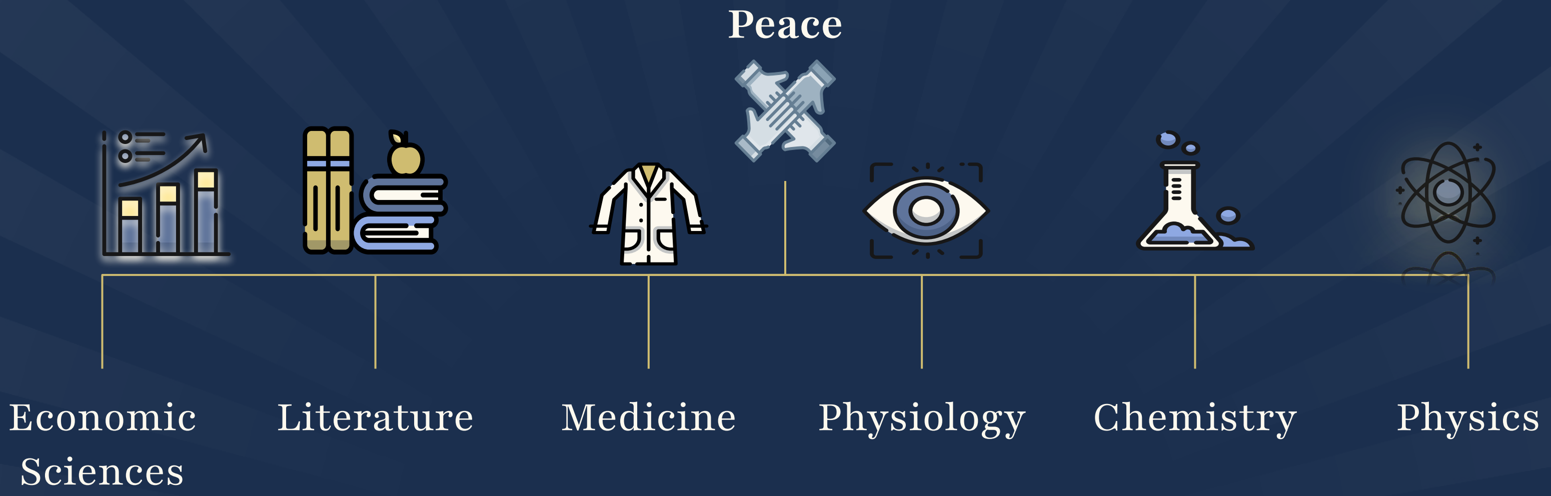
- Background information
- Data graphs
- Obstacles

HISTORICAL OVERVIEW

The Nobel Prizes are five separate prizes that, according to Alfred Nobel's will of 1895, are awarded to "those who, during the preceding year, have conferred the greatest benefit to humankind."

Alfred Nobel

Nobel Prizes



Peace

Economic Sciences

Literature

Medicine

Physiology

Chemistry

Physics

Where are the Nobel Prizes awarded?

Stockholm, Sweden

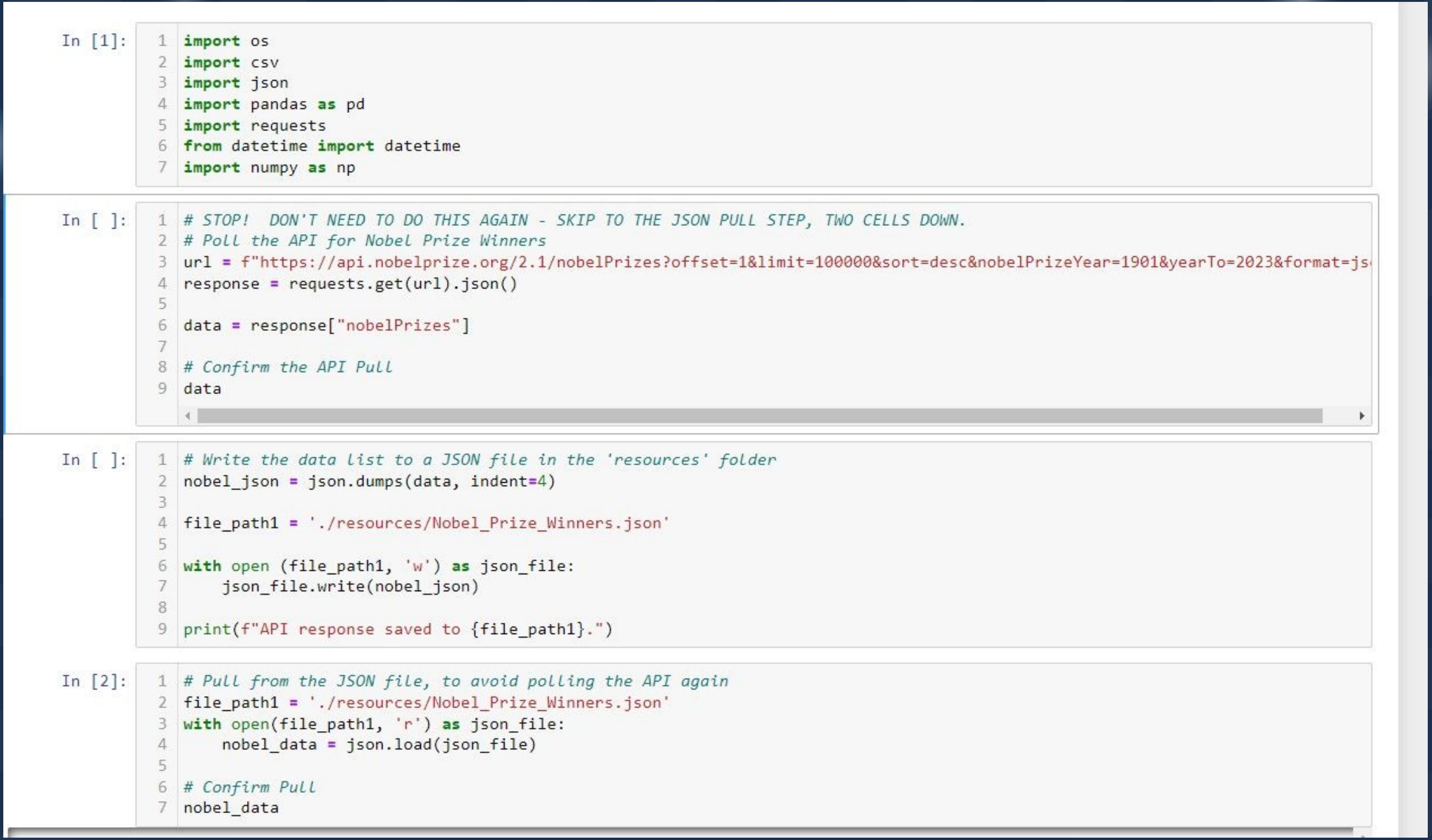
Where Nobel Prizes in Physics, Chemistry, Medicine, Literature and Economic Sciences are awarded

Oslo, Norway

Where the Nobel Peace Prize is awarded

Process

BEGINNING:



In [1]:

```python
import os
import csv
import json
import pandas as pd
import requests
from datetime import datetime
import numpy as np
```

In [ ]:

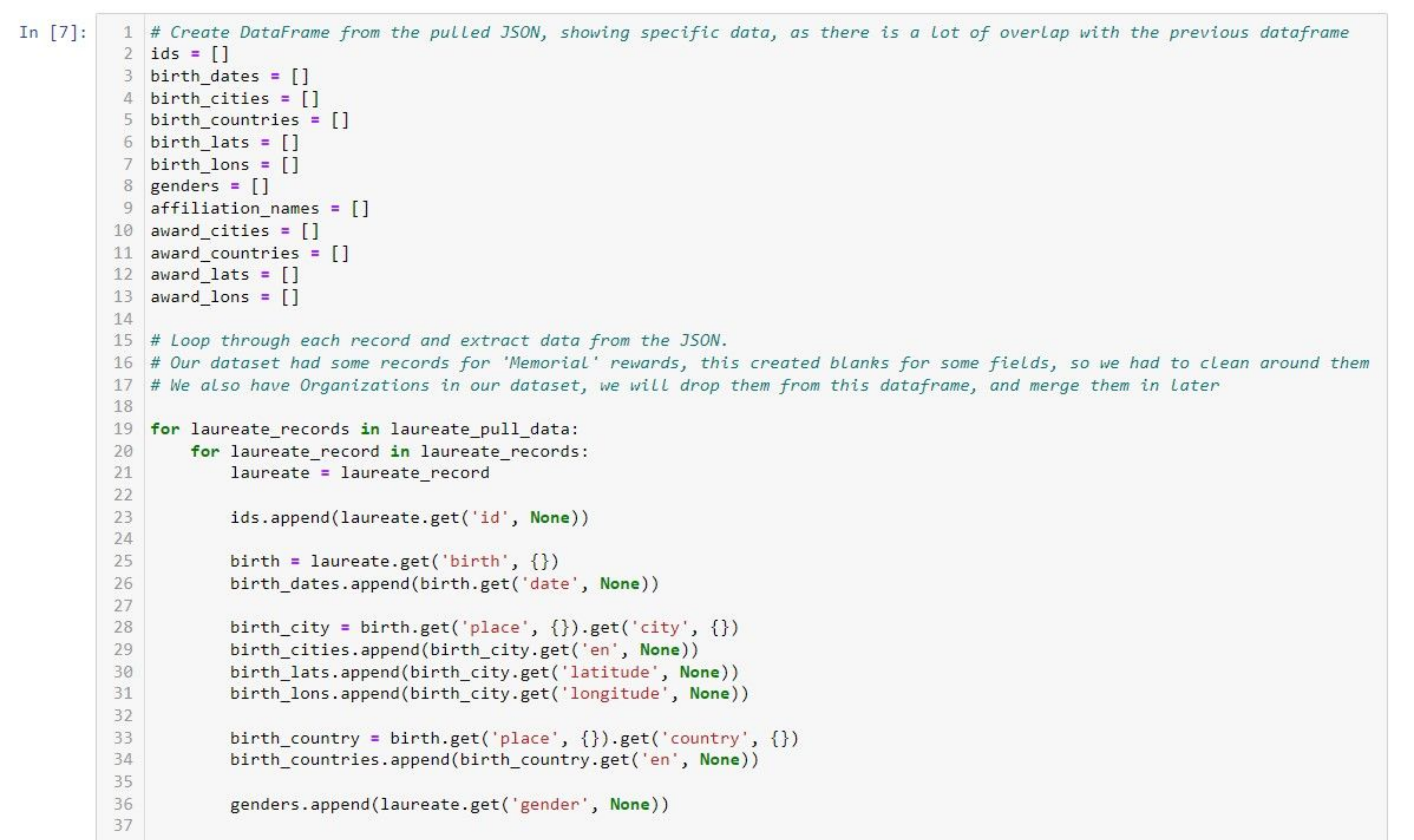
```python
# STOP!  DON'T NEED TO DO THIS AGAIN - SKIP TO THE JSON PULL STEP, TWO CELLS DOWN.
# Poll the API for Nobel Prize Winners
url = f"https://api.nobelprize.org/2.1/nobelPrizes?offset=1&limit=100000&sort=desc&nobelPrizeYear=1901&yearTo=2023&format=js
response = requests.get(url).json()

data = response["nobelPrizes"]

# Confirm the API Pull
data
```

In [ ]:

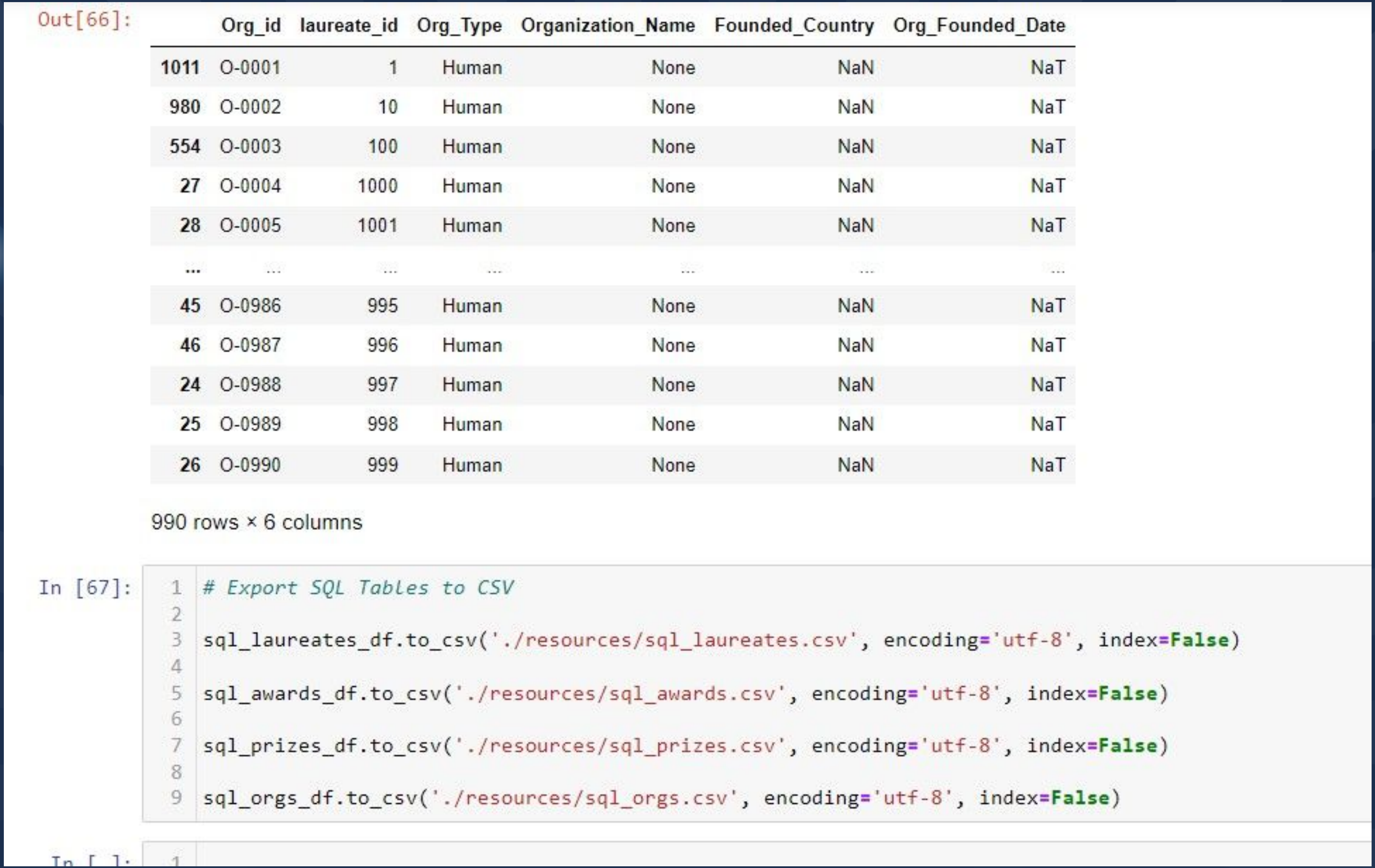
```python
# Write the data list to a JSON file in the 'resources' folder
nobel_json = json.dumps(data, indent=4)

file_path1 = './resources/Nobel_Prize_Winners.json'

with open (file_path1, 'w') as json_file:
    json_file.write(nobel_json)

print(f"API response saved to {file_path1}.")
```

In [2]:

```python
# Pull from the JSON file, to avoid polling the API again
file_path1 = './resources/Nobel_Prize_Winners.json'
with open(file_path1, 'r') as json_file:
    nobel_data = json.load(json_file)

# Confirm Pull
nobel_data
```



```python
# Create DataFrame from the pulled JSON, showing specific data, as there is a lot of overlap with the previous dataframe
ids = []
birth_dates = []
birth_cities = []
birth_countries = []
birth_lats = []
birth_lons = []
genders = []
affiliation_names = []
award_cities = []
award_countries = []
award_lats = []
award_lons = []

# Loop through each record and extract data from the JSON.
# Our dataset had some records for 'Memorial' rewards, this created blanks for some fields, so we had to clean around them
# We also have Organizations in our dataset, we will drop them from this dataframe, and merge them in later

for laureate_records in laureate_pull_data:
    for laureate_record in laureate_records:
        laureate = laureate_record

        ids.append(laureate.get('id', None))

        birth = laureate.get('birth', {})
        birth_dates.append(birth.get('date', None))

        birth_city = birth.get('place', {}).get('city', {})
        birth_cities.append(birth_city.get('en', None))
        birth_lats.append(birth_city.get('latitude', None))
        birth_lons.append(birth_city.get('longitude', None))

        birth_country = birth.get('place', {}).get('country', {})
        birth_countries.append(birth_country.get('en', None))

        genders.append(laureate.get('gender', None))

```



Out[66]:

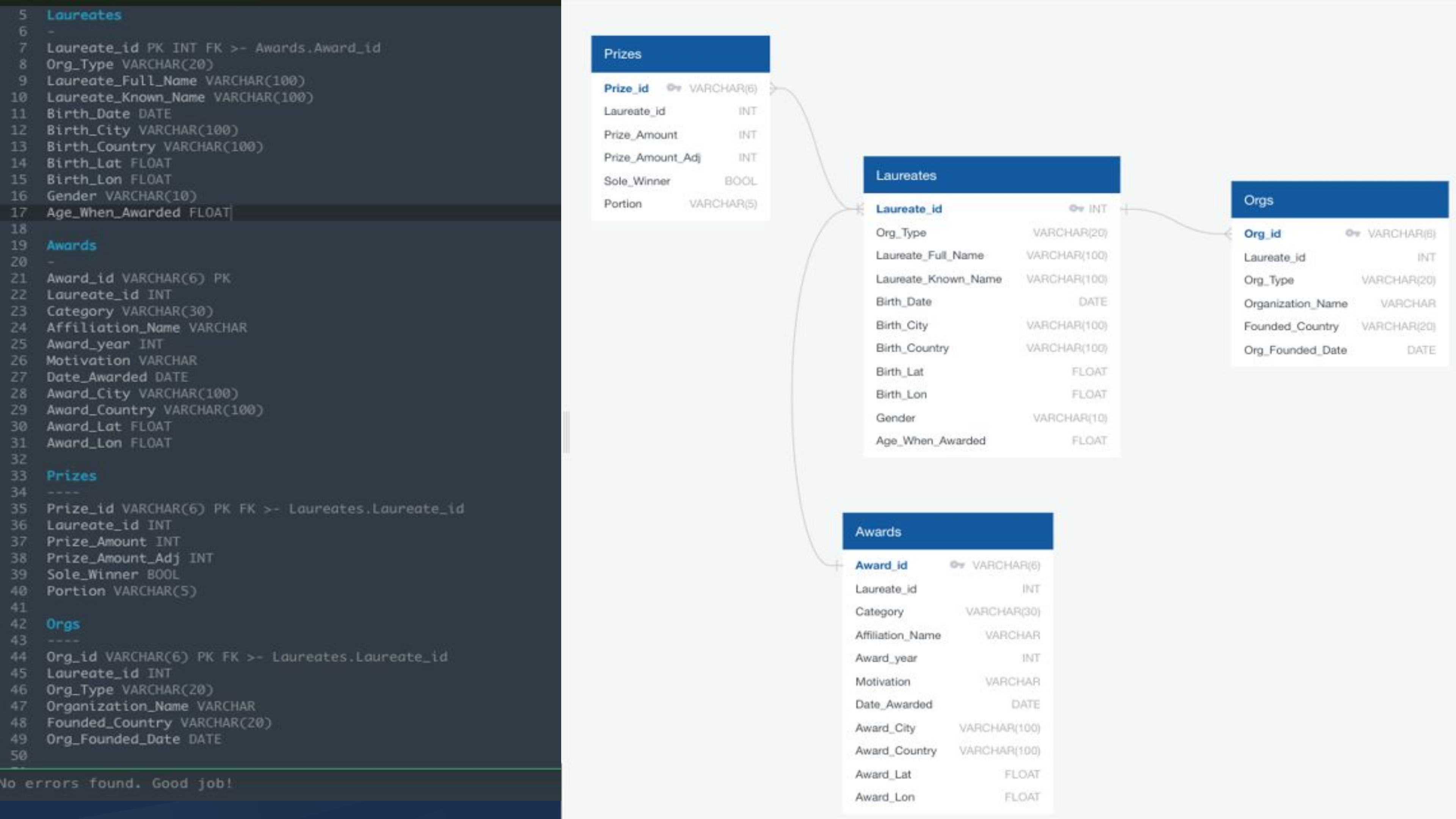
	Org_id	laureate_id	Org_Type	Organization_Name	Founded_Country	Org_Founded_Date
1011	O-0001	1	Human	None	NaN	NaT
980	O-0002	10	Human	None	NaN	NaT
554	O-0003	100	Human	None	NaN	NaT
27	O-0004	1000	Human	None	NaN	NaT
28	O-0005	1001	Human	None	NaN	NaT
...	...	...	...	...	...	...
45	O-0986	995	Human	None	NaN	NaT
46	O-0987	996	Human	None	NaN	NaT
24	O-0988	997	Human	None	NaN	NaT
25	O-0989	998	Human	None	NaN	NaT
26	O-0990	999	Human	None	NaN	NaT

990 rows × 6 columns

In [67]:

```
# Export SQL Tables to CSV

sql_laureates_df.to_csv('./resources/sql_laureates.csv', encoding='utf-8', index=False)

sql_awards_df.to_csv('./resources/sql_awards.csv', encoding='utf-8', index=False)

sql_prizes_df.to_csv('./resources/sql_prizes.csv', encoding='utf-8', index=False)

sql_orgs_df.to_csv('./resources/sql_orgs.csv', encoding='utf-8', index=False)
```

In [ ]:



```
Laureates
-
Laureate_id PK INT FK >- Awards.Award_id
Org_Type VARCHAR(20)
Laureate_Full_Name VARCHAR(100)
Laureate_Known_Name VARCHAR(100)
Birth_Date DATE
Birth_City VARCHAR(100)
Birth_Country VARCHAR(100)
Birth_Lat FLOAT
Birth_Lon FLOAT
Gender VARCHAR(10)
Age_When_Awarded FLOAT

Awards
-
Award_id VARCHAR(6) PK
Laureate_id INT
Category VARCHAR(30)
Affiliation_Name VARCHAR
Award_year INT
Motivation VARCHAR
Date_Awarded DATE
Award_City VARCHAR(100)
Award_Country VARCHAR(100)
Award_Lat FLOAT
Award_Lon FLOAT

Prizes
----
Prize_id VARCHAR(6) PK FK >- Laureates.Laureate_id
Laureate_id INT
Prize_Amount INT
Prize_Amount_Adj INT
Sole_Winner BOOL
Portion VARCHAR(5)

Orgs
----
Org_id VARCHAR(6) PK FK >- Laureates.Laureate_id
Laureate_id INT
Org_Type VARCHAR(20)
Organization_Name VARCHAR
Founded_Country VARCHAR(20)
Org_Founded_Date DATE

No errors found. Good job!
```

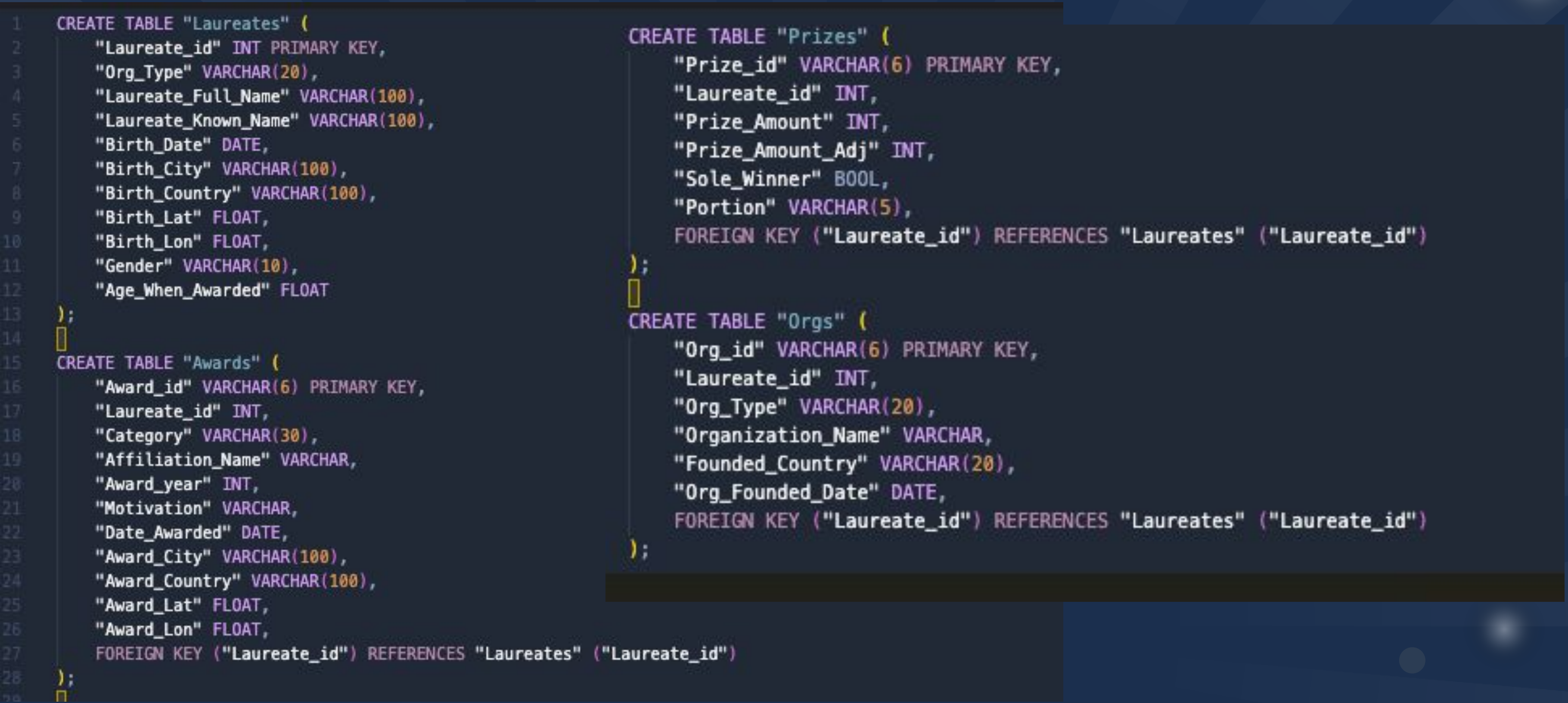
Prizes	
Prize_id	VARCHAR(6)
Laureate_id	INT
Prize_Amount	INT
Prize_Amount_Adj	INT
Sole_Winner	BOOL
Portion	VARCHAR(5)

Laureates	
Laureate_id	INT
Org_Type	VARCHAR(20)
Laureate_Full_Name	VARCHAR(100)
Laureate_Known_Name	VARCHAR(100)
Birth_Date	DATE
Birth_City	VARCHAR(100)
Birth_Country	VARCHAR(100)
Birth_Lat	FLOAT
Birth_Lon	FLOAT
Gender	VARCHAR(10)
Age_When_Awarded	FLOAT

Orgs	
Org_id	VARCHAR(6)
Laureate_id	INT
Org_Type	VARCHAR(20)
Organization_Name	VARCHAR
Founded_Country	VARCHAR(20)
Org_Founded_Date	DATE

Awards	
Award_id	VARCHAR(6)
Laureate_id	INT
Category	VARCHAR(30)
Affiliation_Name	VARCHAR
Award_year	INT
Motivation	VARCHAR
Date_Awarded	DATE
Award_City	VARCHAR(100)
Award_Country	VARCHAR(100)
Award_Lat	FLOAT
Award_Lon	FLOAT

SQL

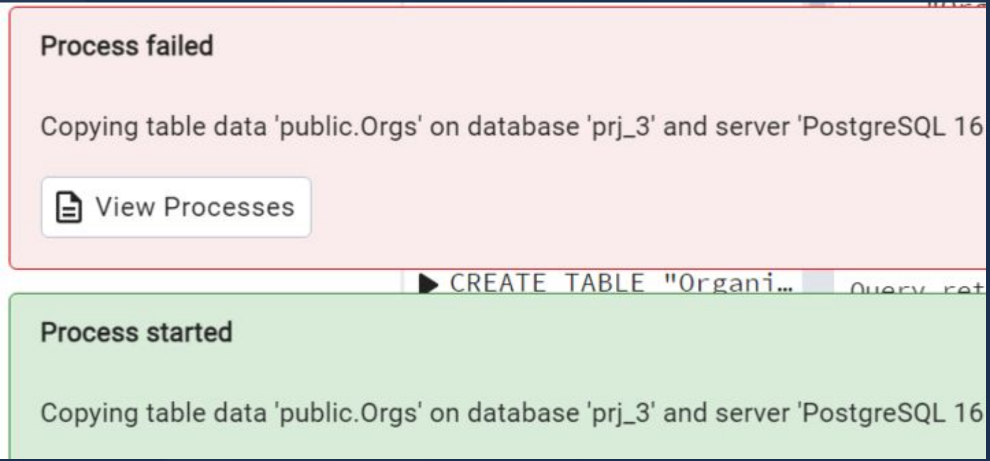


```
CREATE TABLE "Laureates" (
    "Laureate_id" INT PRIMARY KEY,
    "Org_Type" VARCHAR(20),
    "Laureate_Full_Name" VARCHAR(100),
    "Laureate_Known_Name" VARCHAR(100),
    "Birth_Date" DATE,
    "Birth_City" VARCHAR(100),
    "Birth_Country" VARCHAR(100),
    "Birth_Lat" FLOAT,
    "Birth_Lon" FLOAT,
    "Gender" VARCHAR(10),
    "Age_When_Awarded" FLOAT
);

CREATE TABLE "Awards" (
    "Award_id" VARCHAR(6) PRIMARY KEY,
    "Laureate_id" INT,
    "Category" VARCHAR(30),
    "Affiliation_Name" VARCHAR,
    "Award_year" INT,
    "Motivation" VARCHAR,
    "Date_Awarded" DATE,
    "Award_City" VARCHAR(100),
    "Award_Country" VARCHAR(100),
    "Award_Lat" FLOAT,
    "Award_Lon" FLOAT,
    FOREIGN KEY ("Laureate_id") REFERENCES "Laureates" ("Laureate_id")
);
```

```
CREATE TABLE "Prizes" (
    "Prize_id" VARCHAR(6) PRIMARY KEY,
    "Laureate_id" INT,
    "Prize_Amount" INT,
    "Prize_Amount_Adj" INT,
    "Sole_Winner" BOOL,
    "Portion" VARCHAR(5),
    FOREIGN KEY ("Laureate_id") REFERENCES "Laureates" ("Laureate_id")
);

CREATE TABLE "Orgs" (
    "Org_id" VARCHAR(6) PRIMARY KEY,
    "Laureate_id" INT,
    "Org_Type" VARCHAR(20),
    "Organization_Name" VARCHAR,
    "Founded_Country" VARCHAR(20),
    "Org_Founded_Date" DATE,
    FOREIGN KEY ("Laureate_id") REFERENCES "Laureates" ("Laureate_id")
);
```

Obstacles

- A crucial obstacle we faced was trying to connect our databases.
- ★ The issue was resolved by connecting the tables in the order they were structured and fixing some code errors.



Process failed

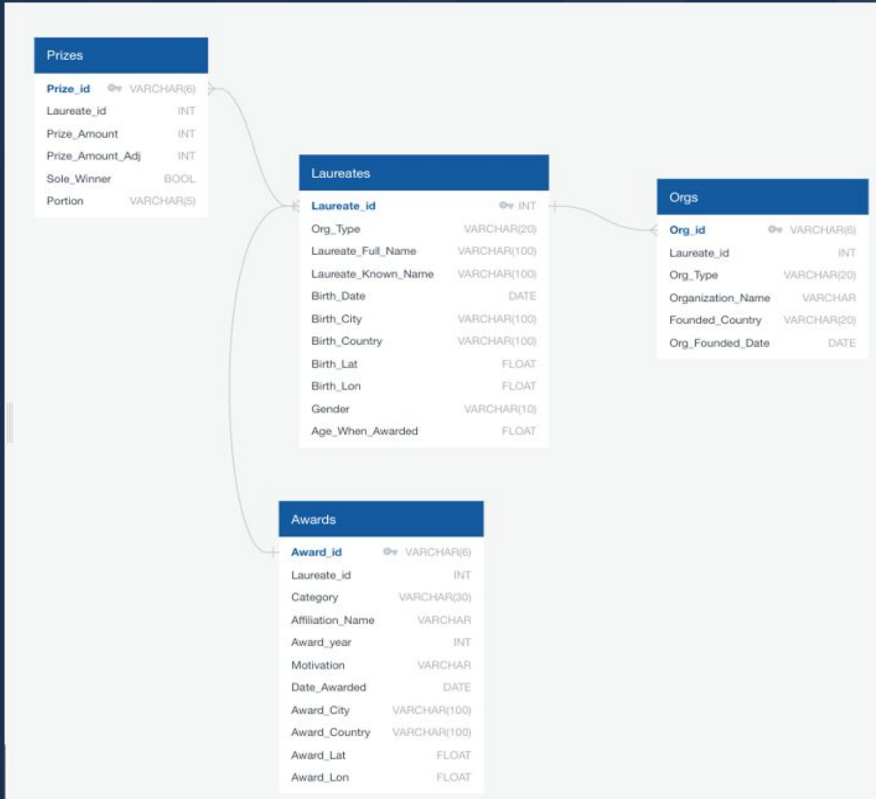
Copying table data 'public.Orgs' on database 'prj_3' and server 'PostgreSQL 16

▶ CREATE TABLE "Organi…

Query ret

Process started

Copying table data 'public.Orgs' on database 'prj_3' and server 'PostgreSQL 16



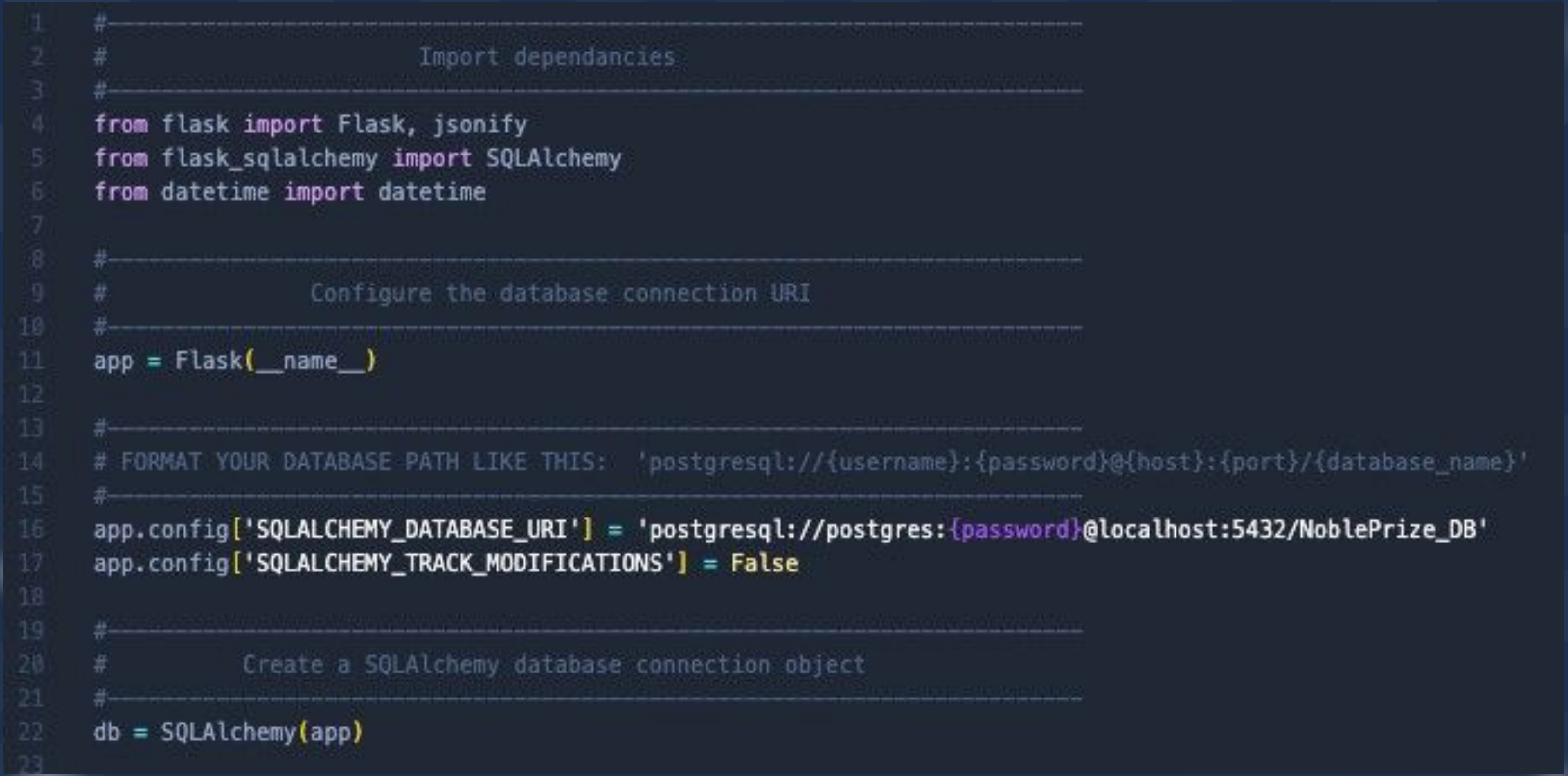
Prizes	
Prize_id	VARCHAR(6)
Laureate_id	INT
Prize_Amount	INT
Prize_Amount_Adj	INT
Sole_Winner	BOOL
Portion	VARCHAR(5)

Laureates	
Laureate_id	INT
Org_Type	VARCHAR(20)
Laureate_Full_Name	VARCHAR(100)
Laureate_Known_Name	VARCHAR(100)
Birth_Date	DATE
Birth_City	VARCHAR(100)
Birth_Country	VARCHAR(100)
Birth_Lat	FLOAT
Birth_Lon	FLOAT
Gender	VARCHAR(10)
Age_When_Awarded	FLOAT

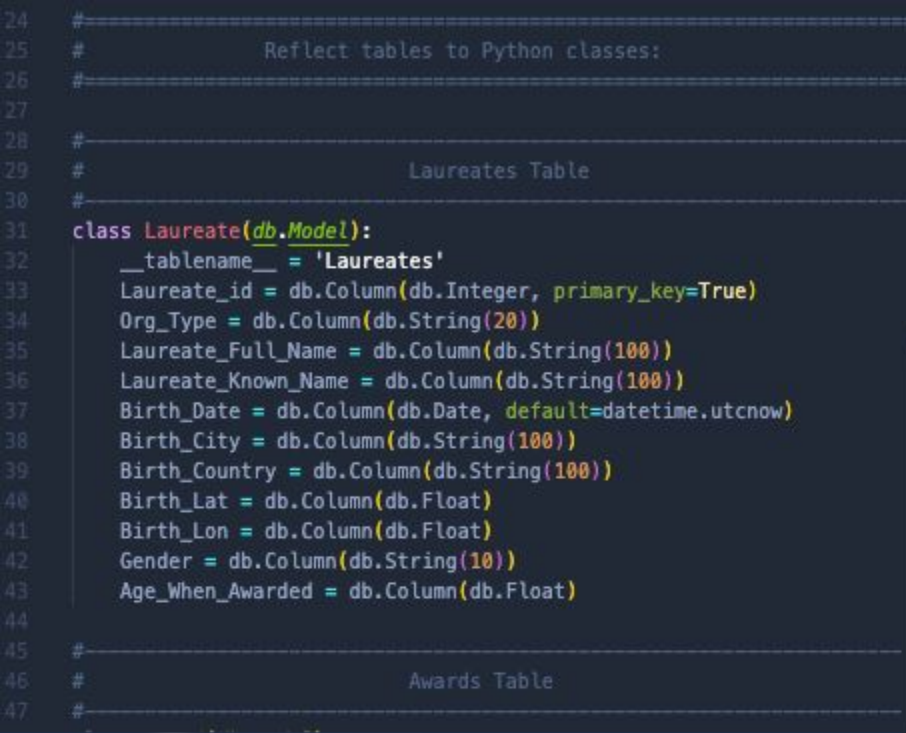
Orgs	
Org_id	VARCHAR(6)
Laureate_id	INT
Org_Type	VARCHAR(20)
Organization_Name	VARCHAR
Founded_Country	VARCHAR(20)
Org_Founded_Date	DATE

Awards	
Award_id	VARCHAR(6)
Laureate_id	INT
Category	VARCHAR(30)
Affiliation_Name	VARCHAR
Award_year	INT
Motivation	VARCHAR
Date_Awarded	DATE
Award_City	VARCHAR(100)
Award_Country	VARCHAR(100)
Award_Lat	FLOAT
Award_Lon	FLOAT

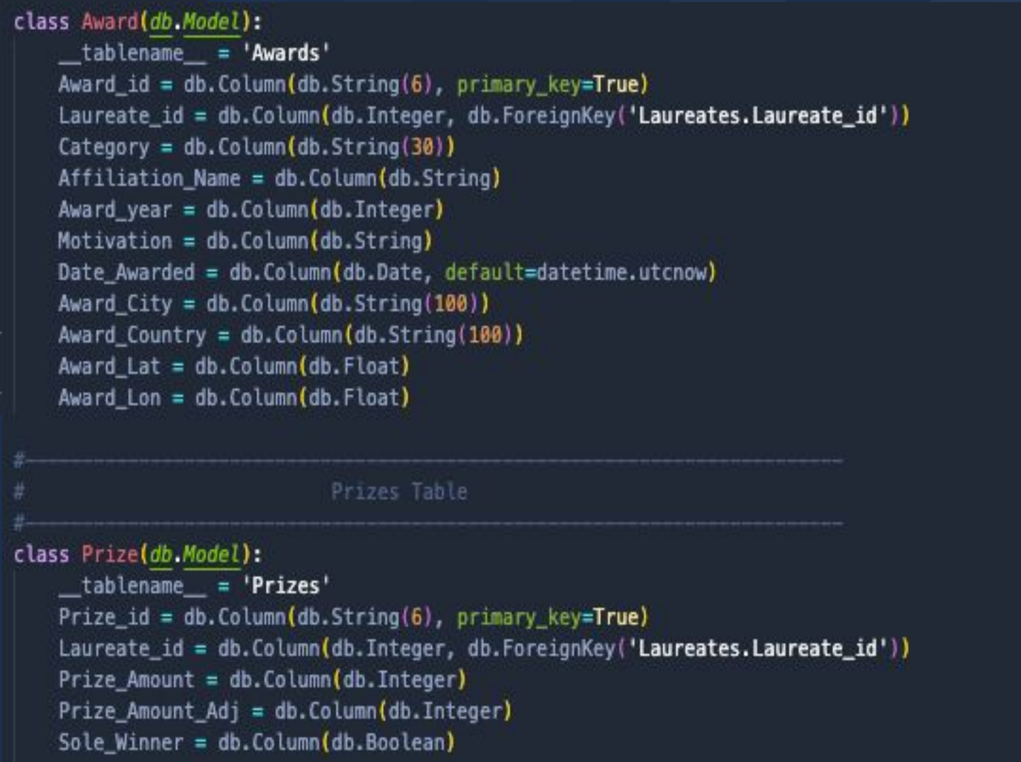
Python/Flask



```
#------------------------------------------------------------------------
#                        Import dependancies
#------------------------------------------------------------------------
from flask import Flask, jsonify
from flask_sqlalchemy import SQLAlchemy
from datetime import datetime

#------------------------------------------------------------------------
#                Configure the database connection URI
#------------------------------------------------------------------------
app = Flask(__name__)

#------------------------------------------------------------------------
# FORMAT YOUR DATABASE PATH LIKE THIS:  'postgresql://{username}:{password}@{host}:{port}/{database_name}'
#------------------------------------------------------------------------
app.config['SQLALCHEMY_DATABASE_URI'] = 'postgresql://postgres:{password}@localhost:5432/NoblePrize_DB'
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False

#------------------------------------------------------------------------
#           Create a SQLAlchemy database connection object
#------------------------------------------------------------------------
db = SQLAlchemy(app)
```



```python
#==============================================================================
#              Reflect tables to Python classes:
#==============================================================================

#------------------------------------------------------------------------------
#                             Laureates Table
#------------------------------------------------------------------------------
class Laureate(db.Model):
    __tablename__ = 'Laureates'
    Laureate_id = db.Column(db.Integer, primary_key=True)
    Org_Type = db.Column(db.String(20))
    Laureate_Full_Name = db.Column(db.String(100))
    Laureate_Known_Name = db.Column(db.String(100))
    Birth_Date = db.Column(db.Date, default=datetime.utcnow)
    Birth_City = db.Column(db.String(100))
    Birth_Country = db.Column(db.String(100))
    Birth_Lat = db.Column(db.Float)
    Birth_Lon = db.Column(db.Float)
    Gender = db.Column(db.String(10))
    Age_When_Awarded = db.Column(db.Float)

#------------------------------------------------------------------------------
#                             Awards Table
#------------------------------------------------------------------------------
```



```python
class Award(db.Model):
    __tablename__ = 'Awards'
    Award_id = db.Column(db.String(6), primary_key=True)
    Laureate_id = db.Column(db.Integer, db.ForeignKey('Laureates.Laureate_id'))
    Category = db.Column(db.String(30))
    Affiliation_Name = db.Column(db.String)
    Award_year = db.Column(db.Integer)
    Motivation = db.Column(db.String)
    Date_Awarded = db.Column(db.Date, default=datetime.utcnow)
    Award_City = db.Column(db.String(100))
    Award_Country = db.Column(db.String(100))
    Award_Lat = db.Column(db.Float)
    Award_Lon = db.Column(db.Float)

#------------------------------------------------------------------------------
#                             Prizes Table
#------------------------------------------------------------------------------
class Prize(db.Model):
    __tablename__ = 'Prizes'
    Prize_id = db.Column(db.String(6), primary_key=True)
    Laureate_id = db.Column(db.Integer, db.ForeignKey('Laureates.Laureate_id'))
    Prize_Amount = db.Column(db.Integer)
    Prize_Amount_Adj = db.Column(db.Integer)
    Sole_Winner = db.Column(db.Boolean)
```

Data Time

It's

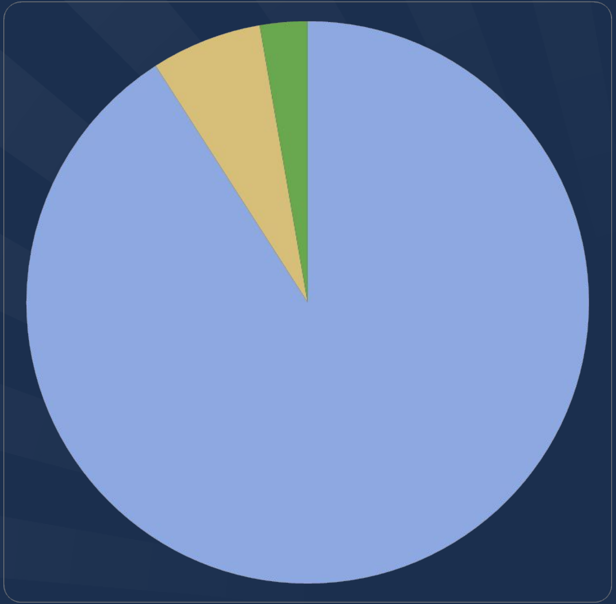
Show

Time

In total there are…….

990 Laureates

The Nobel Prize gender gap



● Men

Organizations

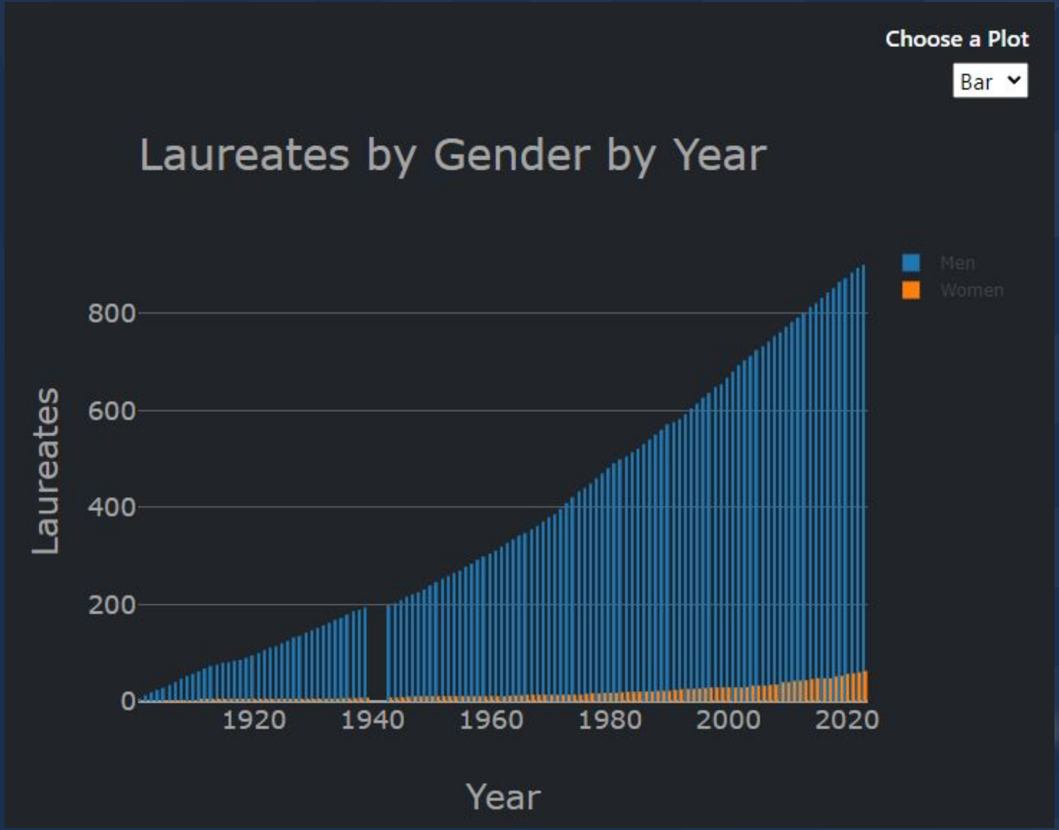
● Women

- While the majority of Laureates are human, there are 27 organizations that were also gifted a Nobel Prize .

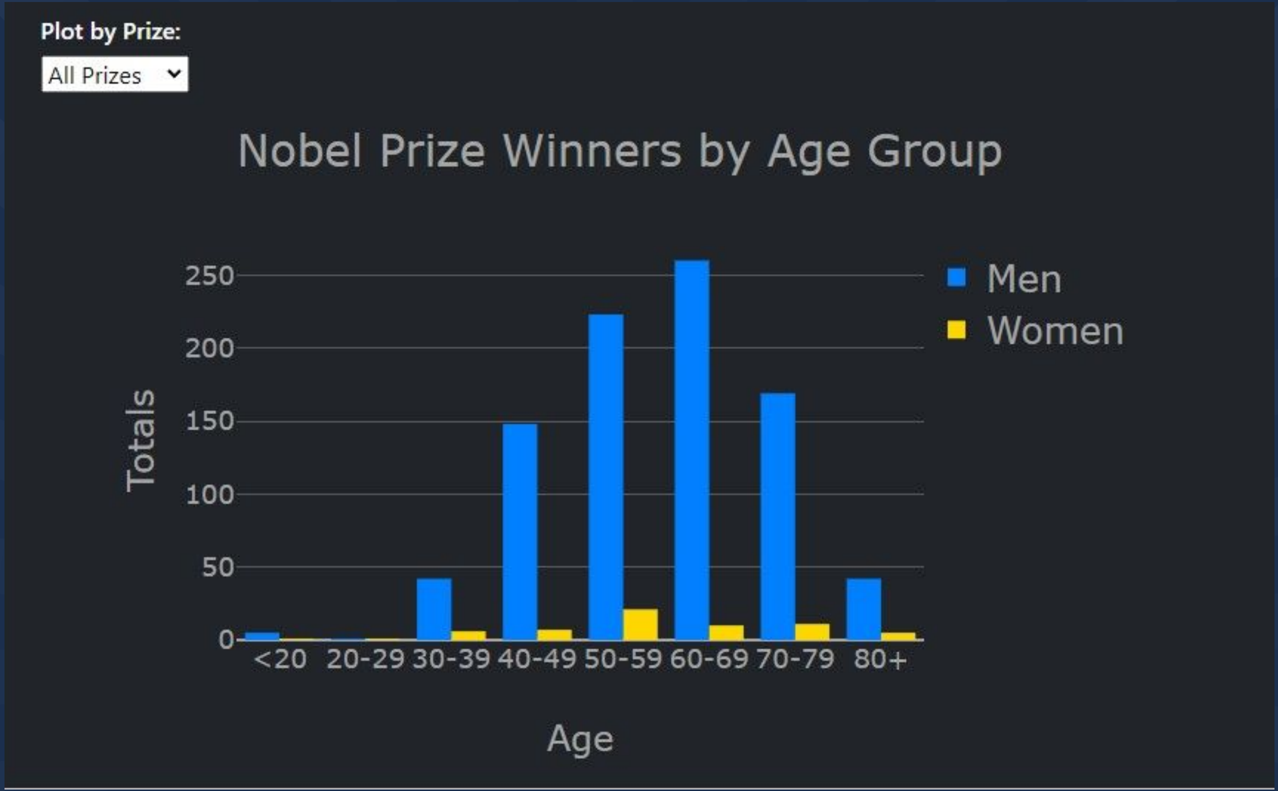
Interactive visualizations

Men & Women Laureate Winners by Year

- Represents cumulative numbers for Nobel Prizes Laureates (both men and women) over the years from 1901 to 2023



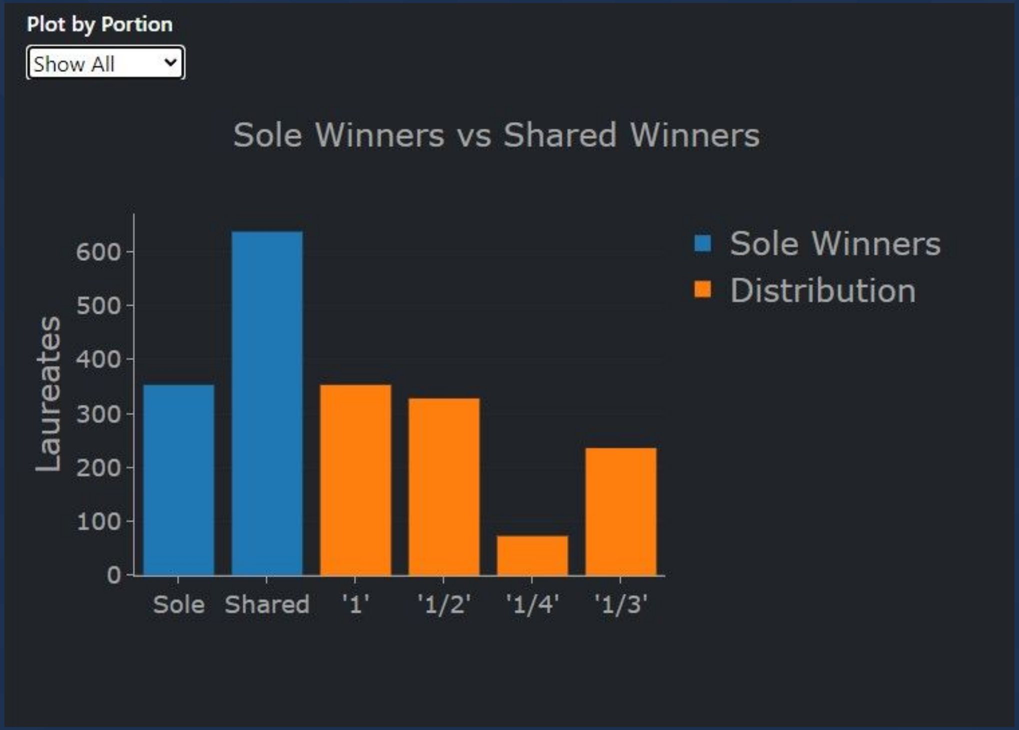
LAUREATES BY AGE GROUP



Plot by Prize:

All Prizes

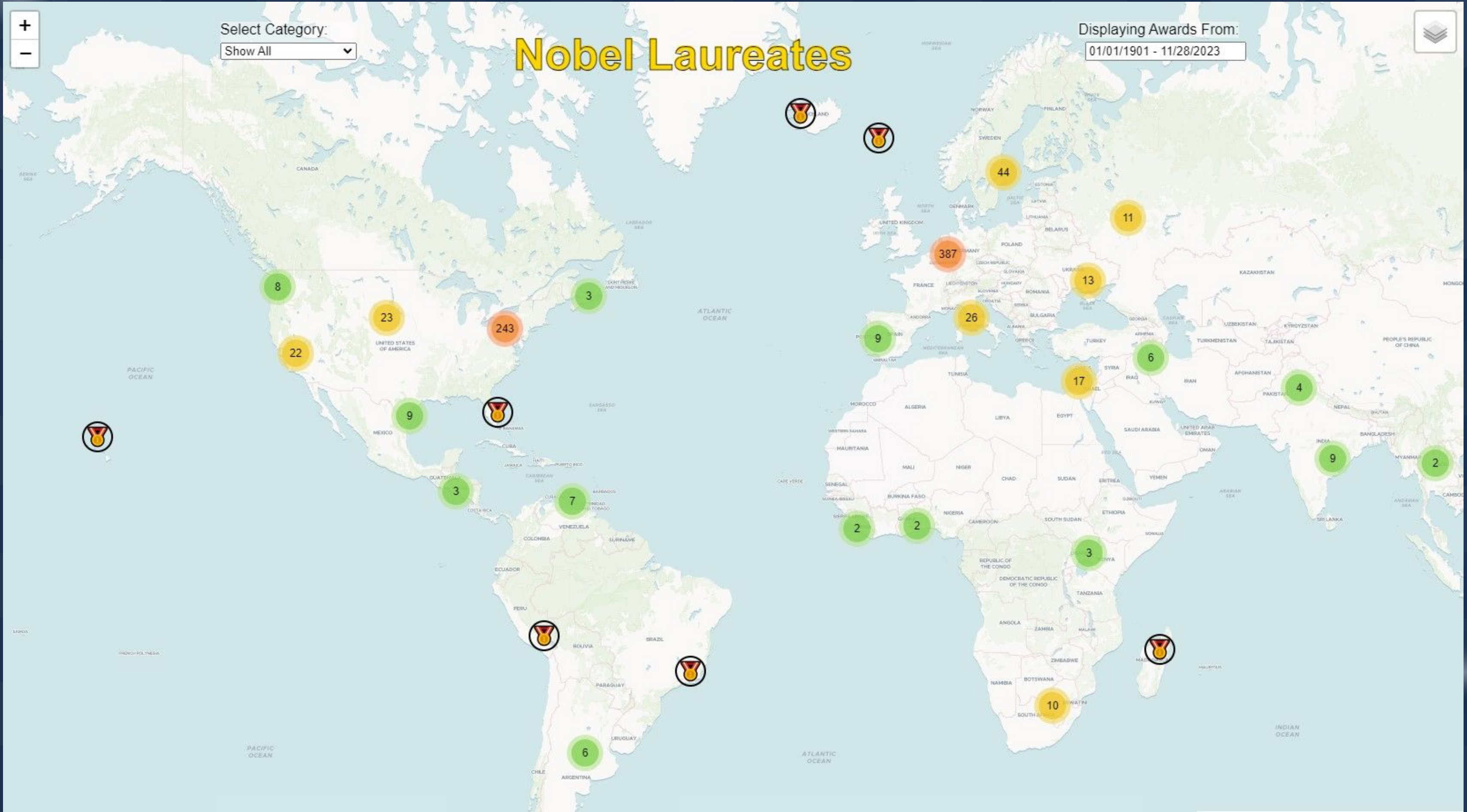
Number of sole & Shared Winners



Plot by Portion

Show All

Laureate location

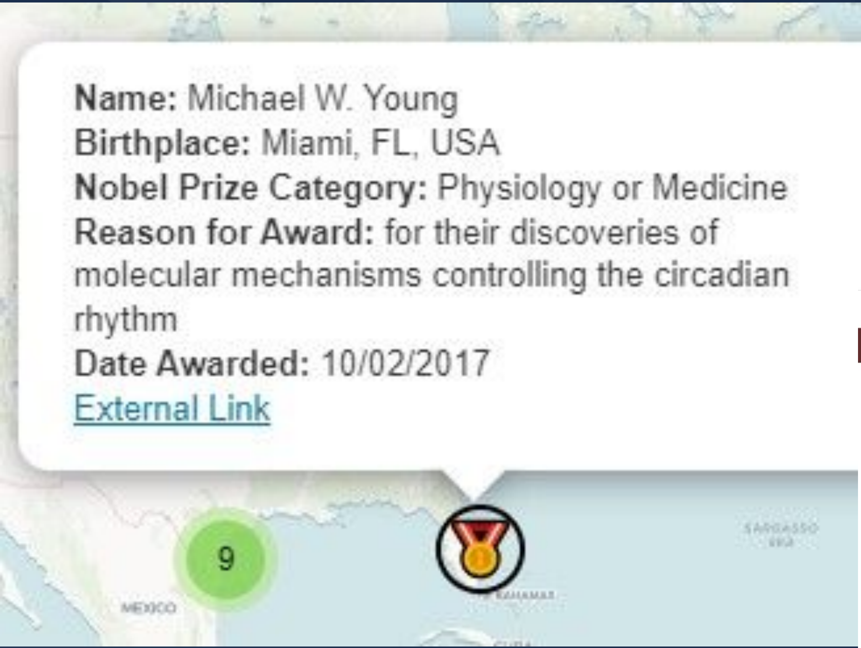


Nobel Laureates

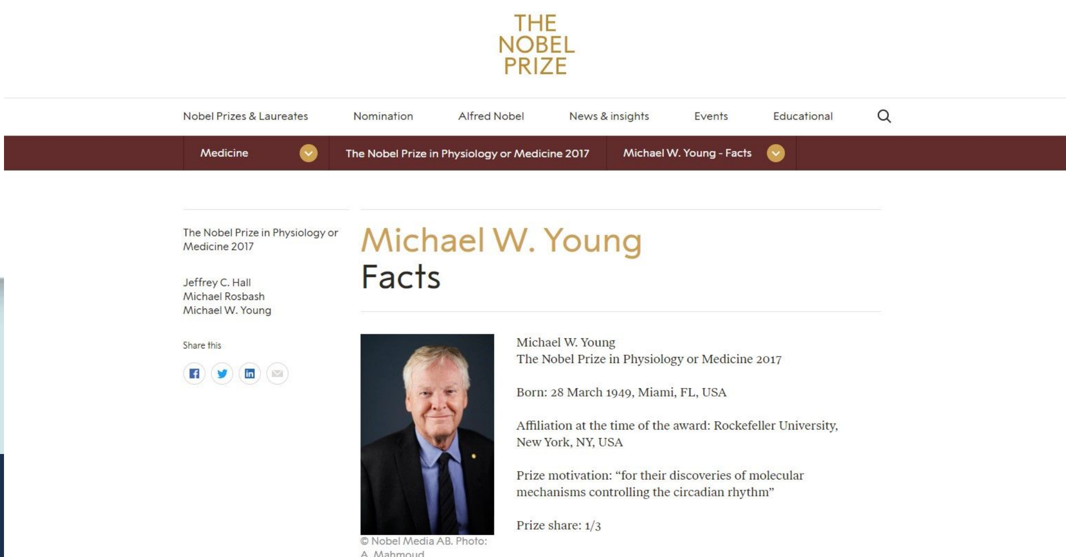
Select Category: Show All

Displaying Awards From: 01/01/1901 - 11/28/2023

LAUREATE LOCATION MAP: interactive link



Name: Michael W. Young
Birthplace: Miami, FL, USA
Nobel Prize Category: Physiology or Medicine
Reason for Award: for their discoveries of molecular mechanisms controlling the circadian rhythm
Date Awarded: 10/02/2017
External Link



THE NOBEL PRIZE

The Nobel Prize in Physiology or Medicine 2017

Jeffrey C. Hall
Michael Rosbash
Michael W. Young

Share this

Michael W. Young
Facts

Michael W. Young
The Nobel Prize in Physiology or Medicine 2017

Born: 28 March 1949, Miami, FL, USA

Affiliation at the time of the award: Rockefeller University, New York, NY, USA

Prize motivation: "for their discoveries of molecular mechanisms controlling the circadian rhythm"

Prize share: 1/3

© Nobel Media AB. Photo: A. Mahmoud

THANK YOU

Do you have any questions?

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**

Please keep this slide for attribution