Data Science Salaries in 2023

CS673 Assignment #1

Aayushi Verma

M.S. Data Science

Pace University

INTRODUCTION

As a data scientist, I am always curious about the scope of this career, which is very broad. I decided to do an analysis on the Data Science Salaries 2023 dataset to glean insights about real salary information, not only from data scientists, but also other adjacent job roles. I used SQL to perform this analysis by importing the dataset and related data into tables, and writing relevant views and queries that yielded interesting insights.

DATA

I used the Data Science Salaries 2023 dataset from Kaggle, which describes data science-related job titles and salaries. In some of the columns (employee_residence, company_location, salary_currency) of the main dataset, the dataset metadata indicates that these columns use ISO 3166 codes for (employee_residence, company_location) and ISO 4217 codes for (salary_currency). So I found a dataset for ISO 3166 codes and for ISO 4217 codes. I imported each of these tables into SQL and created tables for each of them. I also manually created tables for other columns in the dataset (experience_level, employment_type, company_size). I used these secondary tables and joined with the main dataset to make the query outputs more readable.

Before reading the raw CSV files into the database, I had to clean the data. I wrote a script in Python that cleans the data in a format that will be digestible in SQL. The cleaning tasks included replacing spaces with underscores in the column headers, replacing dashes with underscores in the column headers, converting objects to numeric data types when appropriate, and exploring the data columns so we can set the appropriate data type in SQL when defining our table.

DATABASE SCHEMA

I created a database in PGAdmin using PostgreSQL to store the tables and views for this project. Below is the schema for this database, showing the relationships between the tables. I created this schema using DB Designer. The main data source is the table ds_salaries. There are 5 other tables, which have primary keys corresponding to foreign keys in ds_salaries. These tables are: iso3166, iso4217, experience_levels, employment_types, and company_size.

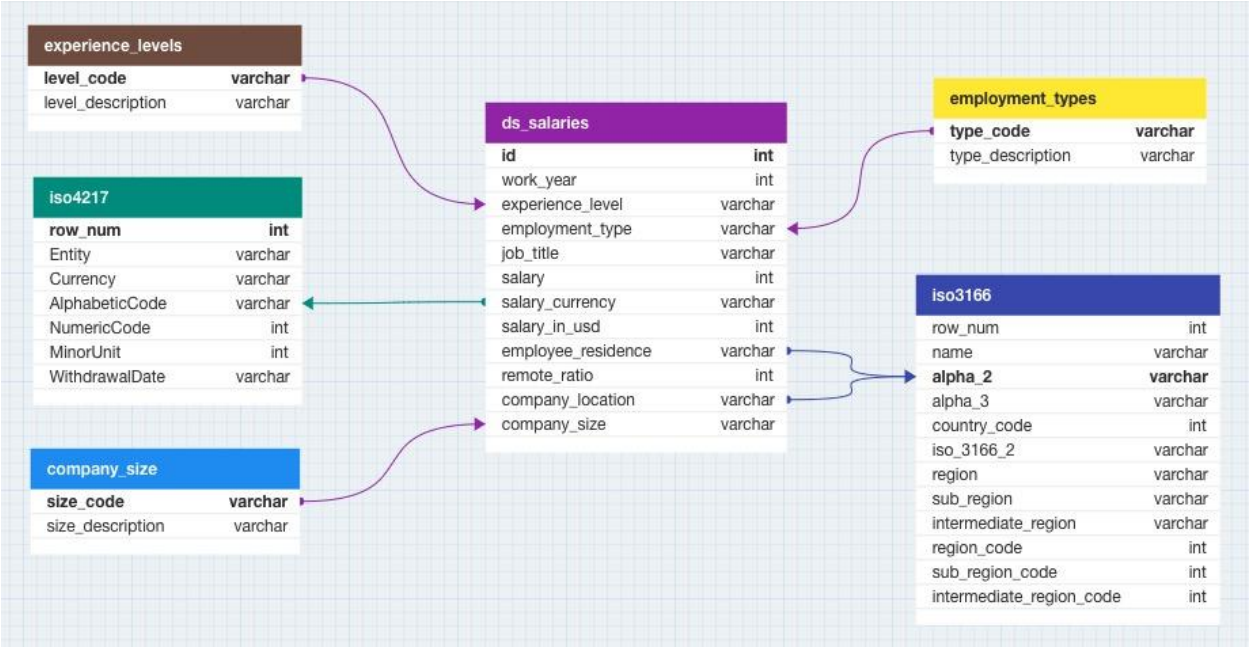


TABLE CREATION

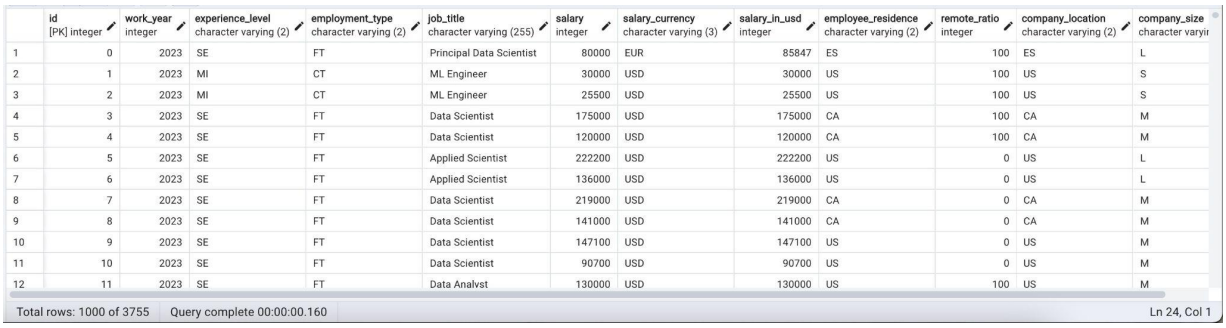
I created a total of 6 tables. Three tables (ds_salaries, iso3166, iso4217) ingest data from the CSV files, and three tables (experience_level, employment_type, company_size) contain lookup data that I inserted manually.

This SQL code creates a table with the columns of the ds_salaries dataset, and inserts that data into the SQL table from the CSV file.

```
-- creating main table
CREATE TABLE ds_salaries (
    id INT PRIMARY KEY,
    work_year INT,
    experience_level VARCHAR(2),
    employment_type VARCHAR(2),
    job_title VARCHAR(255),
    salary INT,
    salary_currency VARCHAR(3),
    salary_in_usd INT,
    employee_residence VARCHAR(2),
    remote_ratio INT,
    company_location VARCHAR(2),
    company_size VARCHAR(2)
);
```

```
-- copying CSV data into table
COPY ds_salaries
FROM
'/Users/av15397n/Documents/GitHub/CS673-Scalable-Databases/Project
1/data/ds_salaries_fixed.csv'
DELIMITER ','
CSV HEADER;

-- checking values copied over
SELECT * FROM ds_salaries;
```



	id [PK] integer	work_year integer	experience_level character varying (2)	employment_type character varying (2)	job_title character varying (255)	salary integer	salary_currency character varying (3)	salary_in_usd integer	employee_residence character varying (2)	remote_ratio integer	company_location character varying (2)	company_size character varyir
1	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
2	1	2023	MI	CT	ML Engineer	30000	USD	30000	US	100	US	S
3	2	2023	MI	CT	ML Engineer	25500	USD	25500	US	100	US	S
4	3	2023	SE	FT	Data Scientist	175000	USD	175000	CA	100	CA	M
5	4	2023	SE	FT	Data Scientist	120000	USD	120000	CA	100	CA	M
6	5	2023	SE	FT	Applied Scientist	222200	USD	222200	US	0	US	L
7	6	2023	SE	FT	Applied Scientist	136000	USD	136000	US	0	US	L
8	7	2023	SE	FT	Data Scientist	219000	USD	219000	CA	0	CA	M
9	8	2023	SE	FT	Data Scientist	141000	USD	141000	CA	0	CA	M
10	9	2023	SE	FT	Data Scientist	147100	USD	147100	US	0	US	M
11	10	2023	SE	FT	Data Scientist	90700	USD	90700	US	0	US	M
12	11	2023	SE	FT	Data Analyst	130000	USD	130000	US	100	US	M

Total rows: 1000 of 3755 | Query complete 00:00:00.160 | Ln 24, Col 1

This SQL code creates a table with the columns of the ISO 3166 country codes dataset, and inserts that data into the SQL table from the CSV file.

```
-- creating country code table
CREATE TABLE iso3166 (
    row_num INT,
    name VARCHAR(255),
    alpha_2 VARCHAR(2) PRIMARY KEY,
    alpha_3 VARCHAR(3),
    country_code INT,
    iso_3166_2 VARCHAR(255),
    region VARCHAR(255),
    sub_region VARCHAR(255),
    intermediate_region VARCHAR(255),
    region_code INT,
    sub_region_code INT,
    intermediate_region_code INT
);

-- copying CSV data into table
COPY iso3166
FROM
'/Users/av15397n/Documents/GitHub/CS673-Scalable-Databases/Project
1/data/iso_3166_fixed.csv'
DELIMITER ','
CSV HEADER;
```

```
-- checking values copied over
SELECT * FROM iso3166;
```

	row_num integer	name character varying (255)	alpha_2 [PK] character varying (2)	alpha_3 character varying (3)	country_code integer	iso_3166_2 character varying (255)	region character varying (255)	sub_region character varying (255)	intermediate_region character varying (255)	region_code integer	sub_reg integer
1	0	Afghanistan	AF	AFG	4	ISO 3166-2:AF	Asia	Southern Asia	[null]	142	
2	1	Åland Islands	AX	ALA	248	ISO 3166-2:AX	Europe	Northern Europe	[null]	150	
3	2	Albania	AL	ALB	8	ISO 3166-2:AL	Europe	Southern Europe	[null]	150	
4	3	Algeria	DZ	DZA	12	ISO 3166-2:DZ	Africa	Northern Africa	[null]	2	
5	4	American Samoa	AS	ASM	16	ISO 3166-2:AS	Oceania	Polynesia	[null]	9	
6	5	Andorra	AD	AND	20	ISO 3166-2:AD	Europe	Southern Europe	[null]	150	
7	6	Angola	AO	AGO	24	ISO 3166-2:AO	Africa	Sub-Saharan Africa	Middle Africa	2	
8	7	Anguilla	AI	AIA	660	ISO 3166-2:AI	Americas	Latin America and the ...	Caribbean	19	
9	8	Antarctica	AQ	ATA	10	ISO 3166-2:AQ	[null]	[null]	[null]	0	
10	9	Antigua and Barbuda	AG	ATG	28	ISO 3166-2:AG	Americas	Latin America and the ...	Caribbean	19	
11	10	Argentina	AR	ARG	32	ISO 3166-2:AR	Americas	Latin America and the ...	South America	19	
12	11	Armenia	AM	ARM	51	ISO 3166-2:AM	Asia	Western Asia	[null]	142	

Total rows: 249 of 249 Query complete 00:00:00.083 Ln 48, Col 1

This SQL code creates a table with the columns of the ISO 4217 currency codes dataset, and inserts that data into the SQL table from the CSV file.

```
-- creating country currencies table
CREATE TABLE iso4217 (
    row_num INT PRIMARY KEY,
    Entity VARCHAR(255),
    Currency VARCHAR(255),
    AlphabeticCode VARCHAR(3),
    NumericCode INT,
    MinorUnit INT,
    WithdrawalDate VARCHAR(255)
);

-- copying CSV data into table
COPY iso4217
FROM
'/Users/av15397n/Documents/GitHub/CS673-Scalable-Databases/Project
1/data/iso_4217_fixed.csv'
DELIMITER ','
CSV HEADER;

-- checking values copied over
SELECT * FROM iso4217;
```

	row_num [PK] integer	entity character varying (255)	currency character varying (255)	alphabeticcode character varying (3)	numericcode integer	minorunit integer	withdrawaldate character varying (255)
1	0	AFGHANISTAN	Afghani	AFN	971	2	[null]
2	1	ÅLAND ISLANDS	Euro	EUR	978	2	[null]
3	2	ALBANIA	Lek	ALL	8	2	[null]
4	3	ALGERIA	Algerian Dinar	DZD	12	2	[null]
5	4	AMERICAN SAMOA	US Dollar	USD	840	2	[null]
6	5	ANDORRA	Euro	EUR	978	2	[null]
7	6	ANGOLA	Kwanza	AOA	973	2	[null]
8	7	ANGUILLA	East Caribbean Dollar	XCD	951	2	[null]
9	8	ANTARCTICA	No universal currency	[null]	0	0	[null]
10	9	ANTIGUA AND BARBU...	East Caribbean Dollar	XCD	951	2	[null]
11	10	ARGENTINA	Argentine Peso	ARS	32	2	[null]
12	11	ARMENIA	Armenian Dram	AMD	51	2	[null]

Total rows: 441 of 441 | Query complete 00:00:00.069 | Ln 68, Col 1

This SQL code creates a table with more descriptive terminology for some of the columns in the ds_salaries dataset.

```
-- creating secondary table
CREATE TABLE experience_levels (
    level_code VARCHAR(2) PRIMARY KEY,
    level_description VARCHAR(255)
);

-- inserting values
INSERT INTO experience_levels VALUES
    ('EX','Executive'),
    ('MI','Mid/Intermediate'),
    ('EN','Entry-Level'),
    ('SE','Senior');

-- checking values
SELECT * FROM experience_levels;
```

	level_code [PK] character varying (2)	level_description character varying (255)
1	EX	Executive
2	MI	Mid/Intermediate
3	EN	Entry-Level
4	SE	Senior

Total rows: 4 of 4 | Query complete 00:00:00.052 | Ln 84, Col 1

This SQL code creates a table with more descriptive terminology for some of the columns in the ds_salaries dataset.

```
-- creating secondary table
CREATE TABLE employment_types (
    type_code VARCHAR(2) PRIMARY KEY,
    type_description VARCHAR(255)
);
```

```
-- inserting values
INSERT INTO employment_types VALUES
    ('PT','Part-Time'),
    ('FL','Freelancer'),
    ('FT','Full-Time'),
    ('CT','Contractor');

-- checking values
SELECT * FROM employment_types;
```

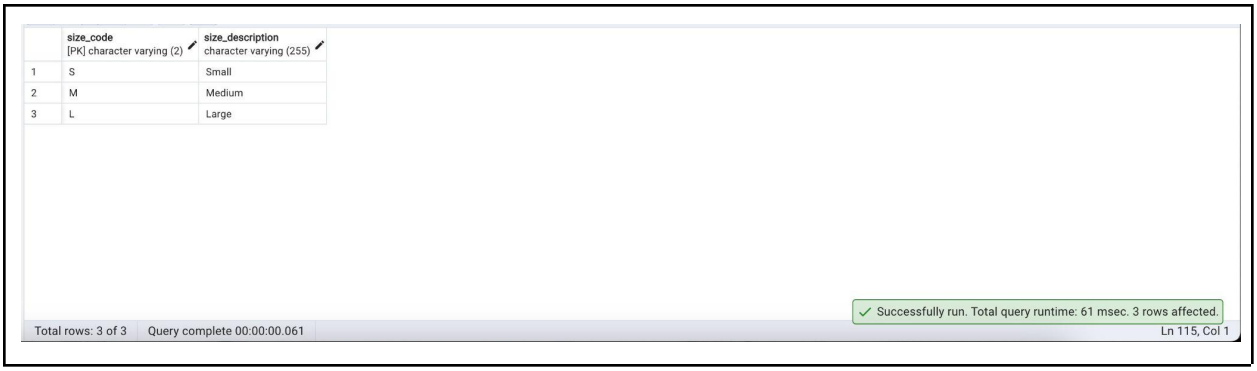
	type_code [PK] character varying (2)	type_description character varying (255)
1	PT	Part-Time
2	FL	Freelancer
3	FT	Full-Time
4	CT	Contractor

This SQL code creates a table with more descriptive terminology for some of the columns in the ds_salaries dataset.

```
-- creating secondary table
CREATE TABLE company_size (
    size_code VARCHAR(2) PRIMARY KEY,
    size_description VARCHAR(255)
);

-- inserting values
INSERT INTO company_size VALUES
    ('S','Small'),
    ('M','Medium'),
    ('L','Large');

-- checking values
SELECT * FROM company_size;
```



	size_code [PK] character varying (2)	size_description character varying (255)
1	S	Small
2	M	Medium
3	L	Large

Successfully run. Total query runtime: 61 msec. 3 rows affected.

Total rows: 3 of 3 | Query complete 00:00:00.061 | Ln 115, Col 1

TABLE INDEXING AND ALTERATIONS

Here, we create an index for each table.

```
-- creating indexing for each table
CREATE INDEX ds_salaries_index ON ds_salaries(id);
CREATE INDEX iso3166_index ON iso3166(row_num);
CREATE INDEX iso4217_index ON iso4217(row_num);
CREATE INDEX experience_levels ON experience_levels(level_code);
CREATE INDEX employment_types ON employment_types(type_code);
CREATE INDEX company_size ON company_size(size_code);
```

Here, we alter the ds_salaries table to add foreign key constraints.

```
-- adding FK constraint on ds_salaries referencing iso3166
ALTER TABLE ds_salaries
    ADD CONSTRAINT fk_employee_residence FOREIGN KEY
(employee_residence) REFERENCES iso3166(alpha_2),
    ADD CONSTRAINT fk_company_location FOREIGN KEY
(company_location) REFERENCES iso3166(alpha_2);
```

VIEW CREATION

We create a view that joins the ds_salaries table with full data from the ISO 4217 table.

```
CREATE VIEW v_ds_salaries_with_currency_info AS
    SELECT DISTINCT
        d.*,
        i.entity, i.currency, i.alphabeticcode
    FROM ds_salaries AS d
    LEFT JOIN iso4217 AS i
        ON d.salary_currency = i.AlphabeticCode;
```

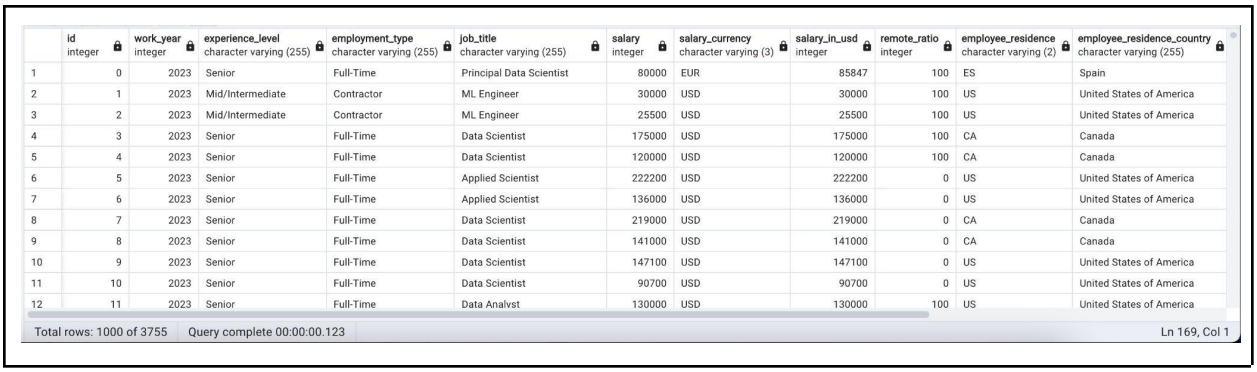
```
SELECT * FROM v_ds_salaries_with_currency_info;
```

	id integer	work_year integer	experience_level character varying (2)	employment_type character varying (2)	job_title character varying (255)	salary integer	salary_currency character varying (3)	salary_in_usd integer	employee_residence character varying (2)	remote_ratio integer	company_location character varying (2)	company_size character varying
1	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
2	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
3	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
4	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
5	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
6	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
7	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
8	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
9	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
10	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
11	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	L
12	0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES			

We create a view that joins all tables to the ds_salaries table, and replaces the codes with their respective descriptions, to make a more readable table.

```
CREATE VIEW v_ds_salaries_nice AS
    SELECT
        d.id, d.work_year,
        x.level_description AS experience_level,
        t.type_description AS employment_type,
        d.job_title, d.salary, d.salary_currency,
        d.salary_in_usd, d.remote_ratio,
        d.employee_residence,
        i.name AS employee_residence_country,
        d.company_location,
        i2.name AS company_location_country,
        s.size_description AS company_size
    FROM ds_salaries AS d
    LEFT JOIN iso3166 AS i
        ON d.employee_residence = i.alpha_2
    LEFT JOIN iso3166 AS i2
        ON d.company_location = i2.alpha_2
    LEFT JOIN experience_levels AS x
        ON d.experience_level = x.level_code
    LEFT JOIN employment_types AS t
        ON d.employment_type = t.type_code
    LEFT JOIN company_size AS s
        ON d.company_size = s.size_code;

SELECT * FROM v_ds_salaries_nice;
```



	id integer	work_year integer	experience_level character varying (255)	employment_type character varying (255)	job_title character varying (255)	salary integer	salary_currency character varying (3)	salary_in_usd integer	remote_ratio integer	employee_residence character varying (2)	employee_residence_country character varying (255)
1	0	2023	Senior	Full-Time	Principal Data Scientist	80000	EUR	85847	100	ES	Spain
2	1	2023	Mid/Intermediate	Contractor	ML Engineer	30000	USD	30000	100	US	United States of America
3	2	2023	Mid/Intermediate	Contractor	ML Engineer	25500	USD	25500	100	US	United States of America
4	3	2023	Senior	Full-Time	Data Scientist	175000	USD	175000	100	CA	Canada
5	4	2023	Senior	Full-Time	Data Scientist	120000	USD	120000	100	CA	Canada
6	5	2023	Senior	Full-Time	Applied Scientist	222200	USD	222200	0	US	United States of America
7	6	2023	Senior	Full-Time	Applied Scientist	136000	USD	136000	0	US	United States of America
8	7	2023	Senior	Full-Time	Data Scientist	219000	USD	219000	0	CA	Canada
9	8	2023	Senior	Full-Time	Data Scientist	141000	USD	141000	0	CA	Canada
10	9	2023	Senior	Full-Time	Data Scientist	147100	USD	147100	0	US	United States of America
11	10	2023	Senior	Full-Time	Data Scientist	90700	USD	90700	0	US	United States of America
12	11	2023	Senior	Full-Time	Data Analyst	130000	USD	130000	100	US	United States of America

Total rows: 1000 of 3755 | Query complete 00:00:00.123 | Ln 169, Col 1

We create a view to get a list of all employees whose residence is outside the USA, but who work for companies located in the USA.

```
-- Finding employees whose residence is not in the US but their
company location is in US
CREATE VIEW v_us_offshore_employees AS
    (SELECT
        company_location_country, employee_residence_country,
        job_title, salary_in_usd, remote_ratio,
        employment_type, experience_level
     FROM v_ds_salaries_nice
     WHERE employee_residence != 'US')
    EXCEPT
    (SELECT
        company_location_country, employee_residence_country,
        job_title, salary_in_usd, remote_ratio,
        employment_type, experience_level
     FROM v_ds_salaries_nice
     WHERE company_location != 'US')
    ORDER BY salary_in_usd ASC;

SELECT * FROM v_us_offshore_employees;
```

	company_location_country character varying (255)	employee_residence_country character varying (255)	job_title character varying (255)	salary_in_usd integer	remote_ratio integer	employment_type character varying (255)	experience_level character varying (255)
1	United States of America	India	Data Scientist	5679	100	Full-Time	Mid/Intermediate
2	United States of America	Pakistan	AI Scientist	12000	100	Part-Time	Entry-Level
3	United States of America	Brazil	AI Scientist	12000	100	Part-Time	Entry-Level
4	United States of America	Mexico	BI Analyst	12000	100	Part-Time	Entry-Level
5	United States of America	Italy	Data Engineer	20000	0	Freelancer	Mid/Intermediate
6	United States of America	Costa Rica	Data Analyst	20000	50	Full-Time	Entry-Level
7	United States of America	Spain	Data Analyst	25216	100	Part-Time	Entry-Level
8	United States of America	Romania	Data Engineer	26005	0	Full-Time	Mid/Intermediate
9	United States of America	Argentina	Data Analyst	30000	100	Full-Time	Entry-Level
10	United States of America	India	Data Scientist	31795	100	Full-Time	Mid/Intermediate
11	United States of America	Hungary	BI Data Analyst	36259	50	Full-Time	Mid/Intermediate
12	United States of America	Viet Nam	Applied Machine Learnin…	38400	100	Full-Time	Mid/Intern

✓ Successfully run. Total query runtime: 56 msec. 40 rows affected.

Total rows: 40 of 40 | Query complete 00:00:00.056 | Ln 74, Col 1

QUERIES

We write a query to find salary ranges with average for each unique job title.

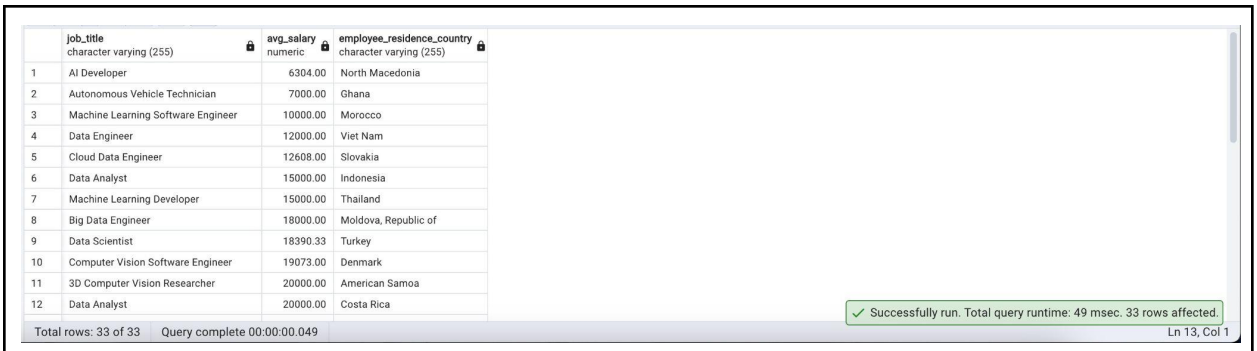
```
-- Finding salary ranges and avg salary for each unique job title
SELECT
    job_title,
    ROUND(AVG(salary_in_usd),2) AS avg_salary,
    MAX(salary_in_usd) AS max_salary,
    MIN(salary_in_usd) AS min_salary
FROM ds_salaries
GROUP BY job_title
ORDER BY min_salary DESC;
```

	job_title character varying (255)	avg_salary numeric	max_salary integer	min_salary integer
1	Data Science Tech Lead	375000.00	375000	375000
2	Cloud Data Architect	250000.00	250000	250000
3	Data Lead	212500.00	225000	200000
4	Principal Machine Learning Engineer	190000.00	190000	190000
5	Principal Data Engineer	192500.00	200000	185000
6	Data Infrastructure Engineer	175051.67	190000	143000
7	Business Intelligence Engineer	174150.00	225000	129300
8	Manager Data Management	125000.00	125000	125000
9	MLOps Engineer	129000.00	134000	124000
10	Deep Learning Researcher	123405.00	123405	123405
11	Machine Learning Manager	155701.33	200000	117104
12	Cloud Database Engineer	155000.00	190000	115000

Total rows: 93 of 93 Query complete 00:00:00.082 Ln 11, Col 26

We write a query to find the job titles and salaries of countries with average salaries greater than or equal to USD $50,000.

```
-- Finding job titles and salaries of countries with average
salaries <= USD $50000
SELECT
    job_title,
    ROUND(AVG(salary_in_usd),2) AS avg_salary,
    employee_residence_country
FROM v_ds_salaries_nice
WHERE employee_residence_country IN (
    SELECT employee_residence_country
    FROM v_ds_salaries_nice
    GROUP BY employee_residence_country
    HAVING MAX(salary_in_usd) <= 50000
)
GROUP BY job_title, employee_residence_country
HAVING ROUND(AVG(salary_in_usd),2) <= 50000
ORDER BY avg_salary ASC;
```



	job_title character varying (255)	avg_salary numeric	employee_residence_country character varying (255)
1	AI Developer	6304.00	North Macedonia
2	Autonomous Vehicle Technician	7000.00	Ghana
3	Machine Learning Software Engineer	10000.00	Morocco
4	Data Engineer	12000.00	Viet Nam
5	Cloud Data Engineer	12608.00	Slovakia
6	Data Analyst	15000.00	Indonesia
7	Machine Learning Developer	15000.00	Thailand
8	Big Data Engineer	18000.00	Moldova, Republic of
9	Data Scientist	18390.33	Turkey
10	Computer Vision Software Engineer	19073.00	Denmark
11	3D Computer Vision Researcher	20000.00	American Samoa
12	Data Analyst	20000.00	Costa Rica

We write a query to find all job titles with salaries greater than or equal to USD $100,000, where the employment type is either freelancer or contractor.

```
-- Finding job titles with salaries >= USD $100000 and either
freelancer or contractor emplyment types
(SELECT
    job_title, salary_in_usd, employment_type
 FROM v_ds_salaries_nice
 WHERE salary_in_usd >= 100000 AND employment_type = 'Freelancer'
)
UNION
(SELECT
    job_title, salary_in_usd, employment_type
 FROM v_ds_salaries_nice
 WHERE salary_in_usd >= 100000 AND employment_type = 'Contractor'
)
ORDER BY salary_in_usd DESC;
```

	job_title character varying (255)	salary_in_usd integer	employment_type character varying (255)
1	Principal Data Scientist	416000	Contractor
2	ML Engineer	270000	Contractor
3	Staff Data Scientist	105000	Contractor
4	Data Scientist	100000	Freelancer
5	Machine Learning Engineer	100000	Freelancer
6	Business Data Analyst	100000	Contractor

We write a query to find common job titles at small and large companies.

```
-- Finding common job titles at small and large companies with
their salaries and experience levels
(SELECT
    job_title, salary_in_usd, experience_level
 FROM v_ds_salaries_nice
 WHERE company_size = 'Small'
)
```

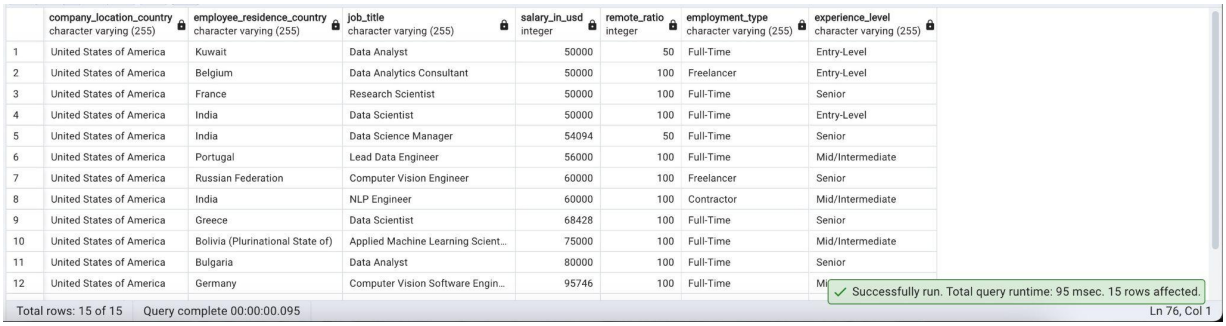
```
INTERSECT
(SELECT
    job_title, salary_in_usd, experience_level
 FROM v_ds_salaries_nice
 WHERE company_size = 'Large'
)
ORDER BY salary_in_usd DESC;
```

	job_title character varying (255)	salary_in_usd integer	employment_type character varying (255)
1	Principal Data Scientist	416000	Contractor
2	ML Engineer	270000	Contractor
3	Staff Data Scientist	105000	Contractor
4	Data Scientist	100000	Freelancer
5	Machine Learning Engineer	100000	Freelancer
6	Business Data Analyst	100000	Contractor

Total rows: 6 of 6 | Query complete 00:00:00.082 | Ln 43, Col 1

We write a query to find all the offshore employees whose salary is between USD $50,000 to $100,000.

```
-- Finding offshore employees whose salary is between USD $50000
and $100000
SELECT *
FROM v_us_offshore_employees
WHERE salary_in_usd BETWEEN 50000 AND 100000;
```



	company_location_country character varying (255)	employee_residence_country character varying (255)	job_title character varying (255)	salary_in_usd integer	remote_ratio integer	employment_type character varying (255)	experience_level character varying (255)
1	United States of America	Kuwait	Data Analyst	50000	50	Full-Time	Entry-Level
2	United States of America	Belgium	Data Analytics Consultant	50000	100	Freelancer	Entry-Level
3	United States of America	France	Research Scientist	50000	100	Full-Time	Senior
4	United States of America	India	Data Scientist	50000	100	Full-Time	Entry-Level
5	United States of America	India	Data Science Manager	54094	50	Full-Time	Senior
6	United States of America	Portugal	Lead Data Engineer	56000	100	Full-Time	Mid/Intermediate
7	United States of America	Russian Federation	Computer Vision Engineer	60000	100	Freelancer	Senior
8	United States of America	India	NLP Engineer	60000	100	Contractor	Mid/Intermediate
9	United States of America	Greece	Data Scientist	68428	100	Full-Time	Senior
10	United States of America	Bolivia (Plurinational State of)	Applied Machine Learning Scient...	75000	100	Full-Time	Mid/Intermediate
11	United States of America	Bulgaria	Data Analyst	80000	100	Full-Time	Senior
12	United States of America	Germany	Computer Vision Software Engin...	95746	100	Full-Time	Mi

Successfully run. Total query runtime: 95 msec. 15 rows affected.

Total rows: 15 of 15 | Query complete 00:00:00.095 | Ln 76, Col 1

We write a scalar subquery to get the total number of offshore employees by country.

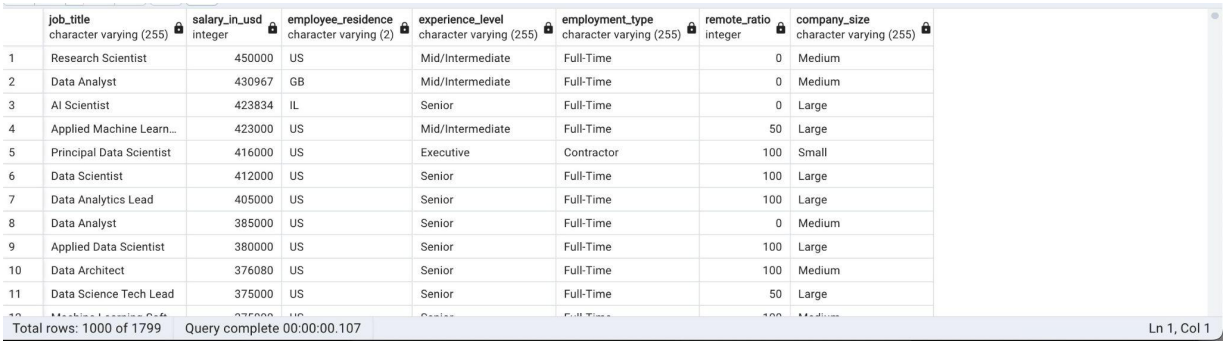
```
-- Finding number of offshore employees and their countries
-- [Scalar Subquery]
SELECT DISTINCT
    employee_residence_country,
        (SELECT COUNT(*)
         FROM v_us_offshore_employees AS v2
         WHERE v1.employee_residence_country =
v2.employee_residence_country
```

```
        )
    AS num_employees
FROM v_us_offshore_employees AS v1
ORDER BY num_employees DESC;
```

	employee_residence_country character varying (255)	num_employees bigint
1	India	7
2	Brazil	3
3	Spain	3
4	Argentina	2
5	Portugal	2
6	Russian Federation	2
7	Belgium	1
8	Bolivia (Plurinational State of)	1
9	Bulgaria	1
10	Canada	1
11	Chile	1
12	China	1

We find all job characteristics for employees who earn greater than or equal to the average salary of the dataset.

```
-- Finding job characteristics with greater than dataset average
salary
WITH avg_salary (value) AS
    (SELECT AVG(salary_in_usd)
     FROM v_ds_salaries_nice)
SELECT
    v.job_title, v.salary_in_usd,
    v.employee_residence, v.experience_level,
    v.employment_type, v.remote_ratio,
    v.company_size
FROM v_ds_salaries_nice AS v, avg_salary
WHERE v.salary_in_usd >= avg_salary.value
ORDER BY salary_in_usd DESC;
```



	job_title character varying (255)	salary_in_usd integer	employee_residence character varying (2)	experience_level character varying (255)	employment_type character varying (255)	remote_ratio integer	company_size character varying (255)
1	Research Scientist	450000	US	Mid/Intermediate	Full-Time	0	Medium
2	Data Analyst	430967	GB	Mid/Intermediate	Full-Time	0	Medium
3	AI Scientist	423834	IL	Senior	Full-Time	0	Large
4	Applied Machine Learn...	423000	US	Mid/Intermediate	Full-Time	50	Large
5	Principal Data Scientist	416000	US	Executive	Contractor	100	Small
6	Data Scientist	412000	US	Senior	Full-Time	100	Large
7	Data Analytics Lead	405000	US	Senior	Full-Time	100	Large
8	Data Analyst	385000	US	Senior	Full-Time	0	Medium
9	Applied Data Scientist	380000	US	Senior	Full-Time	100	Large
10	Data Architect	376080	US	Senior	Full-Time	100	Medium
11	Data Science Tech Lead	375000	US	Senior	Full-Time	50	Large

We create a procedure and a function to obtain the monthly salary of each employee.

```
-- Defining a procedure to obtain monthly salary
CREATE PROCEDURE monthly_salary()
```

```
LANGUAGE plpgsql AS $$
    DECLARE
    total float;
    BEGIN
        SELECT ROUND(salary_in_usd / 12, 2)
        FROM v_us_offshore_employees;
        --RETURN total;
    END;
$$
```

```
-- Defining a function to obtain monthly salary
CREATE FUNCTION func_monthly_salary()
RETURNS float
LANGUAGE plpgsql AS $$
    DECLARE
        monthly_salary_in_usd float;
    BEGIN
        SELECT ROUND(salary_in_usd / 12, 2)
            AS monthly_salary_in_usd
        FROM v_us_offshore_employees;
        RETURN monthly_salary_in_usd;
    END;
$$;
```

CONCLUSION

In this project, I ingested data from different sources (CSV, manual) into SQL, created tables, wrote queries to explore the data, and wrote a procedure and function. It was an insightful foray into learning about data science job types, salaries, and other job characteristics.