PROJECT: ANALYZING UNICORN COMPANIES

Did you know that the average return from investing in stocks is 10% per year (not accounting for inflation)? But who wants to be average?!

You have been asked to support an investment firm by analyzing trends in high-growth companies. They are interested in understanding which industries are producing the highest valuations and the rate at which new high-value companies are emerging. Providing them with this information gives them a competitive insight as to industry trends and how they should structure their portfolio looking forward.

You have been given access to their `unicorns` database, which contains the following tables:

dates

Column	Description
`company_id`	A unique ID for the company.
`date_joined`	The date that the company became a unicorn.
`year_founded`	The year that the company was founded.

funding

Column	Description
`company_id`	A unique ID for the company.
`valuation`	Company value in US dollars.
`funding`	The amount of funding raised in US dollars.
`select_investors`	A list of key investors in the company.

industries

Column	Description
`company_id`	A unique ID for the company.
`industry`	The industry that the company operates in.

companies

Column	Description
`company_id`	A unique ID for the company.
`company`	The name of the company.
`city`	The city where the company is headquartered.
`country`	The country where the company is headquartered.
`continent`	The continent where the company is headquartered.

Projects Data DataFrame as `data_types`

```
-- Check columns and data types for specific tables

SELECT table_name, column_name, data_type
FROM information_schema.columns
WHERE table_name IN ('dates', 'funding', 'industries', 'companies')
ORDER BY table_name, ordinal_position;
```

index	table_name	column_name	data_type
0	companies	company_id	integer
1	companies	company	character varying
2	companies	city	character varying
3	companies	country	character varying
4	companies	continent	character varying
5	dates	company_id	integer
6	dates	date_joined	date
7	dates	year_founded	integer
8	funding	company_id	integer
9	funding	valuation	bigint
10	funding	funding	bigint
11	funding	select_investors	character varying
12	industries	company_id	integer
13	industries	industry	character varying

Rows: 14 Expand

Projects Data DataFrame as `d`

```
--- Overview of table dates

SELECT *
FROM dates
LIMIT 5;
```

	c…	date_joined	year…
0	189	2017-06-24T00:00:00.000	1919
1	848	2021-06-01T00:00:00.000	2019
2	556	2022-02-15T00:00:00.000	2011
3	999	2021-11-17T00:00:00.000	2020
4	396	2021-10-21T00:00:00.000	2021

Rows: 5 Expand

Projects Data DataFrame as `f`

```
--- Overview of table funding

SELECT *
FROM funding
LIMIT 5;
```

	c…	v…	f…	select_investors
0	189	4000000000	0	EQT Partners
1	848	1000000000	100000000	Dragonfly Captial, Qiming Venture Partners, …
2	556	2000000000	100000000	Blackstone, Bessemer Venture Partners
3	999	1000000000	100000000	Goldman Sachs Asset Management, 3L
4	396	2000000000	100000000	Insight Partners, Softbank Group, Connect Ve…

Rows: 5 Expand

Projects Data DataFrame as `i`

```
--- Overview of table industries

SELECT *
FROM industries
LIMIT 5;
```

	c...	industry
0	189	Health
1	848	Fintech
2	556	Internet software & services
3	999	Internet software & services
4	396	Fintech

Rows: 5 Expand

Projects Data DataFrame as

```
--- Overview of table companies

SELECT *
FROM companies
LIMIT 5;
```

	c...	company	city	country	contin...
0	189	Otto Bock HealthCare	Duderstadt	Germany	Europe
1	848	Matrixport		Singapore	Asia
2	556	Cloudinary	Santa Clara	United States	North America
3	999	PLACE	Bellingham	United States	North America
4	396	candy.com	New York	United States	North America

Rows: 5 Expand

The output

Your query should return a table in the following format:

industry	year	num_unicorns	average_valuation_billions
industry1	2021	---	---
industry2	2020	---	---
industry3	2019	---	---
industry1	2021	---	---
industry2	2020	---	---
industry3	2019	---	---
industry1	2021	---	---
industry2	2020	---	---
industry3	2019	---	---

Where `industry1`, `industry2`, and `industry3` are the three top-performing industries.

Projects Data DataFrame as

```
SELECT
        i.industry,
        COUNT(*) AS number_of_companies
FROM industries AS i
INNER JOIN dates AS d
    USING(company_id)
WHERE EXTRACT(YEAR FROM d.date_joined) IN (2019, 2020, 2021)
GROUP BY i.industry
ORDER BY number_of_companies DESC
LIMIT 3;
```

	industry	number_of_com...
0	Fintech	173
1	Internet software & services	152
2	E-commerce & direct-to-consumer	75

Rows: 3 Expand

Projects Data DataFrame as

```
SELECT
        i.industry,
        EXTRACT(YEAR FROM d.date_joined) AS year,
        COUNT(i.company_id) AS num_unicorns,
        ROUND(AVG(f.valuation), 2) AS avg_valuation
FROM industries AS i
INNER JOIN dates AS d
    USING(company_id)
INNER JOIN funding AS f
    USING(company_id)
WHERE EXTRACT(YEAR FROM d.date_joined) IN (2019, 2020, 2021)
GROUP BY i.industry, year
ORDER BY year ASC
```

	industry		num...	avg_v...
0	Artificial intelligence	2019	14	4500000000
1	Auto & transportation	2019	6	4166666666.67
2	Consumer & retail	2019	3	3666666666.67
3	Cybersecurity	2019	4	2250000000
4	Data management & analytics	2019	4	11500000000
5	E-commerce & direct-to-consumer	2019	12	2583333333.33
6	Edtech	2019	1	1000000000
7	Fintech	2019	20	6800000000
8	Health	2019	3	3333333333.33
9	Internet software & services	2019	13	4230769230.77
10	Mobile & telecommunications	2019	4	2000000000
11	Other	2019	9	2888888888.89
12	Supply chain, logistics, & delivery	2019	8	3000000000
13	Travel	2019	3	4000000000
14	Artificial intelligence	2020	3	4000000000
15	Auto & transportation	2020	5	3000000000

Rows: 43 Expand

Projects Data DataFrame as

```
--- CTE1 top_performing_industries

WITH top_performing_industries AS
(
    SELECT
        i.industry,
        COUNT(*) AS number_of_companies
    FROM industries AS i
    INNER JOIN dates AS d
        USING(company_id)
    WHERE EXTRACT(YEAR FROM d.date_joined) IN (2019, 2020, 2021)
    GROUP BY i.industry
    ORDER BY number_of_companies DESC
    LIMIT 3
),

-----------------------------------------------------------------------------------------------------------

--- CTE2 top_valuation

top_valuation AS
(
    SELECT
        i.industry,
        EXTRACT(YEAR FROM d.date_joined) AS year,
        COUNT(i.company_id) AS num_unicorns,
        ROUND(AVG(f.valuation), 2) AS avg_valuation
    FROM industries AS i
    INNER JOIN dates AS d
        USING(company_id)
    INNER JOIN funding AS f
        USING(company_id)
    WHERE EXTRACT(YEAR FROM d.date_joined) IN (2019, 2020, 2021)
    GROUP BY i.industry, year
    ORDER BY year ASC
)

-----------------------------------------------------------------------------------------------------------

--- Final Query

SELECT
    industry,
    year,
    num_unicorns,
    ROUND(AVG(avg_valuation) / 1000000000, 2) AS average_valuation_billions
FROM top_valuation AS tv
INNER JOIN top_performing_industries AS tpi
    USING(industry)
WHERE year IN (2019, 2020, 2021)
  AND industry IN (SELECT industry FROM top_performing_industries)
GROUP BY industry, year, num_unicorns
ORDER BY year DESC, num_unicorns DESC;
```

	industry	year	num...	average_valuation_billions
0	Fintech	2021	138	2.75
1	Internet software & services	2021	119	2.15
2	E-commerce & direct-to-consumer	2021	47	2.47
3	Internet software & services	2020	20	4.35
4	E-commerce & direct-to-consumer	2020	16	4
5	Fintech	2020	15	4.33
6	Fintech	2019	20	6.8
7	Internet software & services	2019	13	4.23
8	E-commerce & direct-to-consumer	2019	12	2.58

Rows: 9 Expand