PROJECT: EXPLORING TRENDS IN AMERICAN BABY NAMES

datalab

How have American baby name tastes changed since 1920? Which names have remained popular for over 100 years, and how do those names compare to more recent top baby names? These are considerations for many new parents, but the skills you'll practice while answering these queries are broadly applicable. After all, understanding trends and popularity is important for many businesses, too!

You'll be working with data provided by the United States Social Security Administration, which lists first names along with the number and sex of babies they were given to in each year. For processing speed purposes, the dataset is limited to first names which were given to over 5,000 American babies in a given year. The data spans 101 years, from 1920 through 2020.

The Data

`baby_names`

column	type	description
`year`	int	year
`first_name`	varchar	first name
`sex`	varchar	`sex` of babies given `first_name`
`num`	int	number of babies of `sex` given `first_name` in that `year`

Projects Data DataFrame as `usa_baby_names`

```
-- Run this code to view the data in baby_names
SELECT *
FROM baby_names
LIMIT 5;
```

index	year	first_name	sex	num
0	1920	Mary	F	70982
1	1920	Dorothy	F	36643
2	1920	Helen	F	35097
3	1920	Margaret	F	27994
4	1920	Ruth	F	26101

Rows: 5 Expand

Projects Data DataFrame as name_types

```sql
-- List the overall top five names in alphabetical order and find out if each name is "Classic" or "Trendy."
SELECT first_name,
       COUNT(DISTINCT(year)) AS sum,
       CASE WHEN COUNT(DISTINCT(year)) >= 50 THEN 'Classic'
       ELSE 'Trendy' END AS popularity_type
FROM baby_names
GROUP BY first_name
ORDER BY first_name ASC
LIMIT 5;
```

index	first_name	sum	popularity_type
0	Aaliyah	3	Trendy
1	Aaron	51	Classic
2	Abigail	28	Trendy
3	Adam	46	Trendy
4	Addison	13	Trendy

Rows: 5 Expand

Projects Data DataFrame as top_20

```sql
-- What were the top 20 male names overall, and how did the name Paul rank?
WITH S as(
SELECT first_name,
    SUM(num)  AS sum,
    RANK() OVER(ORDER BY SUM(num) DESC) AS name_rank
FROM baby_names
WHERE sex = 'M'
GROUP BY first_name
)
SELECT name_rank,
       first_name,
       sum
FROM S
ORDER BY name_rank
LIMIT 20;
```

index	name_rank	first_name	sum
0	1	James	4748138
1	2	John	4510721
2	3	Robert	4495199
3	4	Michael	4278824
4	5	William	3614424
5	6	David	3571498
6	7	Richard	2414838
7	8	Joseph	2361382
8	9	Thomas	2166802
9	10	Charles	2112352
10	11	Christopher	2012792
11	12	Daniel	1824274
12	13	Matthew	1567204
13	14	Anthony	1344352
14	15	Donald	1280236
15	16	Mark	1265910

Rows: 20 Expand

Projects Data DataFrame as a_names

```sql
-- a_names
SELECT
    first_name,
    SUM(num) AS total_occurrences
FROM
    baby_names
WHERE
    sex = 'F'
    AND year IN (1920, 2020)
GROUP BY
    first_name
HAVING
    COUNT(DISTINCT year) = 2
ORDER BY
    first_name;
```

index	first_name	total_occurrences
0	Eleanor	14832
1	Elizabeth	23125
2	Emma	20818
3	Evelyn	23283
4	Grace	12741
5	Hazel	12765

Rows: 6 Expand