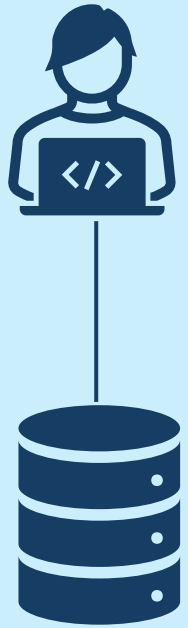
Global Layoffs
SQL Project



Syed Nadeem Yousuf

Introduction

In this project dataset which is used is Global Layoffs, This project is designed to research global layoff trends in a more profound way by identifying patterns of layoffs across various industries, companies, and regions for a comprehensive overview of workforce reduction worldwide..

Dataset Includes:

- Company : The companies which layoffs (String)
- Location HQ : The Location of the company's headquarter (String)
- Industry : Company belong to which industry (String)
- Laid OFF : Number of people laid off (Int)
- Date : Date of layoffs (date) **, Data is till 5th of June 2024.**
- Funds raised (In Millions or billions) : Amount of funding company has raised (string)
- Stage : Stage of the company (string)
- Country : Company headquarter is in which country (string)
- Percentage : Layoff Percentage

Start with Data Cleaning

- **Step 1 :** Making copy of actual data and working on the copy data because if any mistake happen at least original data will be there. This is a good practice to do.

```
create table copy_global_layoff
like global_layoffs;
```

Use this query to create the same table as original one.

```
insert into copy_global_layoff
select * from global_layoffs;
```

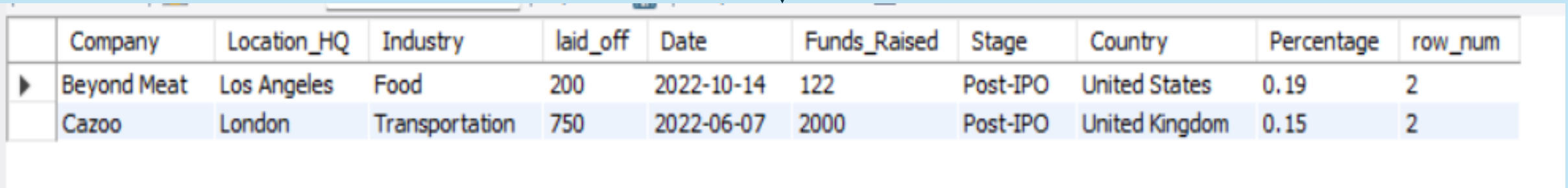
Use this query to insert same data as the original table is having.

	Company	Location_HQ	Industry	laid_off	Date	Funds_Raised	Stage	Country	Percentage
▶	Oda	Oslo	Food	150	2024-06-05	691	Unknown	Norway	
	Pagaya	Tel Aviv	Finance	100	2024-06-05	2000	Post-IPO	Israel	0.2
	Aleph Farms	Tel Aviv	Food	30	2024-06-05	119	Unknown	Israel	0.3
	MoonPay	Dover	Crypto	30	2024-06-05	651	Unknown	United States	0.1
	Yext	New York City	Marketing		2024-06-05	117	Post-IPO	United States	0.12
	Microsoft	Seattle	Other	1000	2024-06-03	1	Post-IPO	United States	
	OrCam	Jerusalem	Healthcare	100	2024-06-03	86	Unknown	Israel	0.5
	Google	SF Bay Area	Consumer	100	2024-05-31	26	Post-IPO	United States	
	Tropic	New York City	Finance	40	2024-05-31	67	Series B	United States	
	Gro Intelligence	New York City	Food		2024-05-31	118	Series B	United States	0.1

- **Step 2 :** Removing duplicates values

There was no key column in this dataset such as sr.no or something like uniquely identifier so I use window function (ROW_Number) and cte to identify duplicates and remove it.

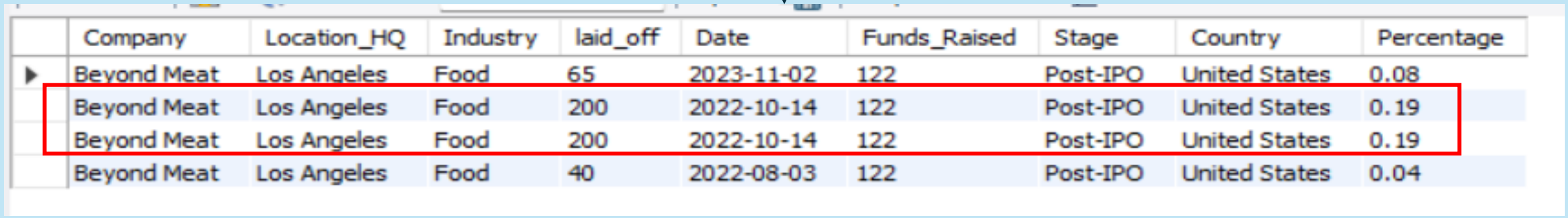
```
with cte as (select*,row_number()
over(partition by company,location_HQ,industry,laid_off,date,funds_raised,stage,country,percentage) as row_num
from copy_global_layoff)
select * from cte where row_num > 1;
```



Company	Location_HQ	Industry	laid_off	Date	Funds_Raised	Stage	Country	Percentage	row_num
Beyond Meat	Los Angeles	Food	200	2022-10-14	122	Post-IPO	United States	0.19	2
Cazoo	London	Transportation	750	2022-06-07	2000	Post-IPO	United Kingdom	0.15	2

This query shows that there are 2 duplicate records for both company.

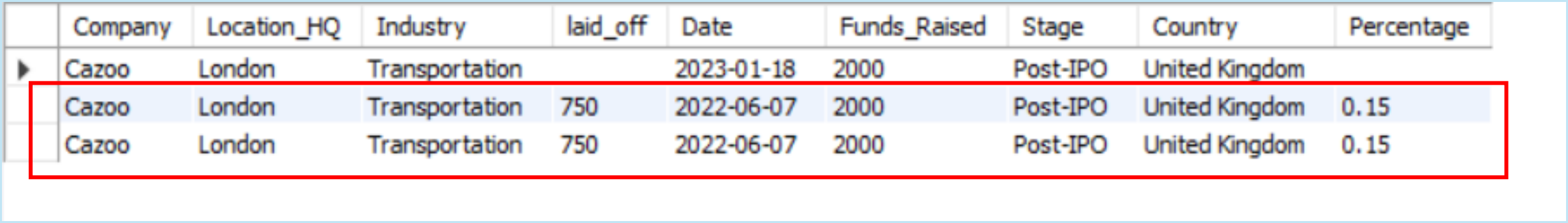
```sql
select * from copy_global_layoff
where company like '%meat'
```



Company	Location_HQ	Industry	laid_off	Date	Funds_Raised	Stage	Country	Percentage
Beyond Meat	Los Angeles	Food	65	2023-11-02	122	Post-IPO	United States	0.08
Beyond Meat	Los Angeles	Food	200	2022-10-14	122	Post-IPO	United States	0.19
Beyond Meat	Los Angeles	Food	200	2022-10-14	122	Post-IPO	United States	0.19
Beyond Meat	Los Angeles	Food	40	2022-08-03	122	Post-IPO	United States	0.04

This record was duplicate so removing one of them.

```sql
select * from copy_global_layoff
where company = 'cazoo';
```



Company	Location_HQ	Industry	laid_off	Date	Funds_Raised	Stage	Country	Percentage
Cazoo	London	Transportation		2023-01-18	2000	Post-IPO	United Kingdom	
Cazoo	London	Transportation	750	2022-06-07	2000	Post-IPO	United Kingdom	0.15
Cazoo	London	Transportation	750	2022-06-07	2000	Post-IPO	United Kingdom	0.15

This record was duplicate so removing one of them.

If there were lots of duplicate records, then creating a new table and inserting data from the window function (Row_Number) which was used to identify duplicate records. And then delete the duplicate records from the new table.

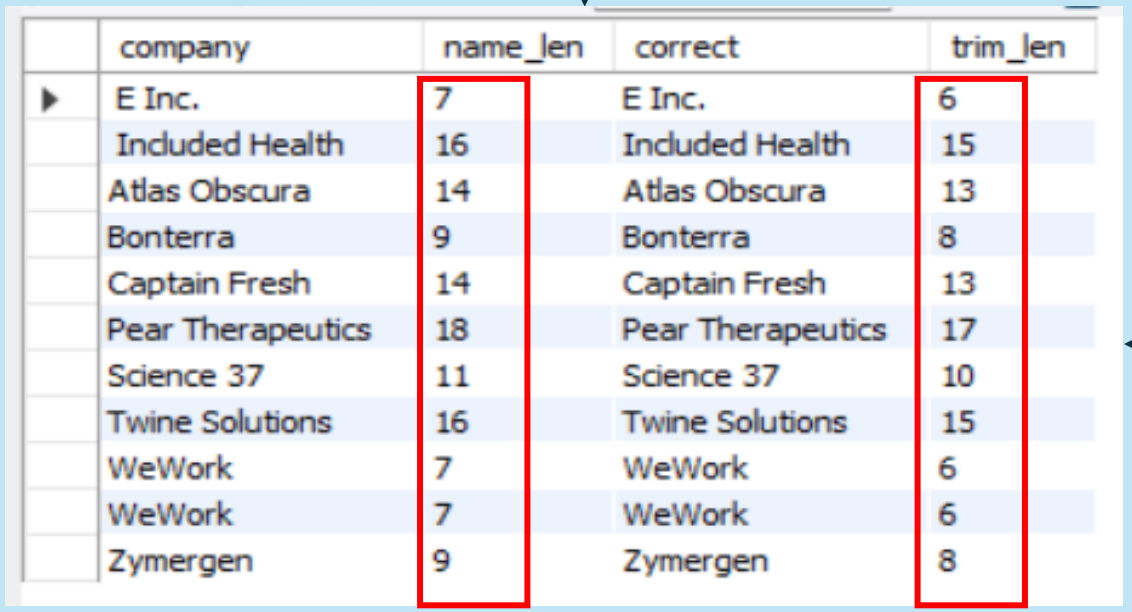
- **Step 3 :** Standardizing Formats

Check for any spelling errors or extra spaces or datatype of columns.

```
select company,length(company) name_len,
trim(company) correct,length(trim(company)) trim_len
from copy_global_layoff
having name_len != trim_len
order by company;
```

Use this query to Fix it.

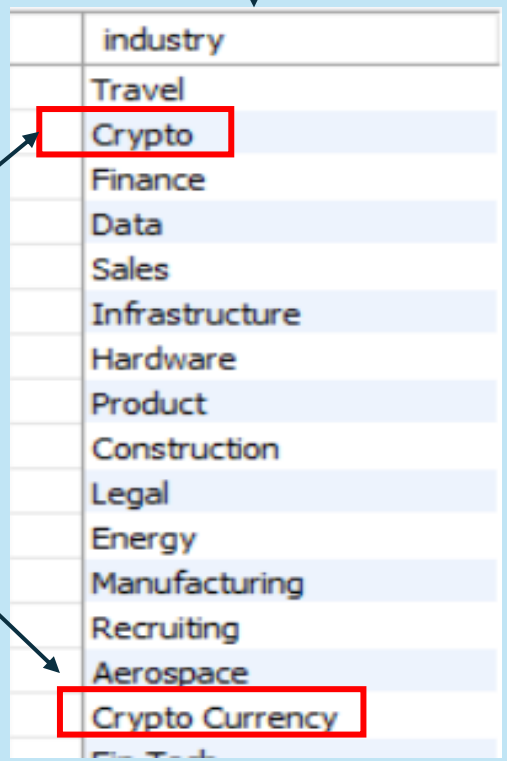
```
update copy_global_layoff
set company = trim(company);
```



company	name_len	correct	trim_len
E Inc.	7	E Inc.	6
Included Health	16	Included Health	15
Atlas Obscura	14	Atlas Obscura	13
Bonterra	9	Bonterra	8
Captain Fresh	14	Captain Fresh	13
Pear Therapeutics	18	Pear Therapeutics	17
Science 37	11	Science 37	10
Twine Solutions	16	Twine Solutions	15
WeWork	7	WeWork	6
WeWork	7	WeWork	6
Zymergen	9	Zymergen	8

In this dataset these records have extra spaces

```sql
select distinct industry
from copy_global_layoff;
```



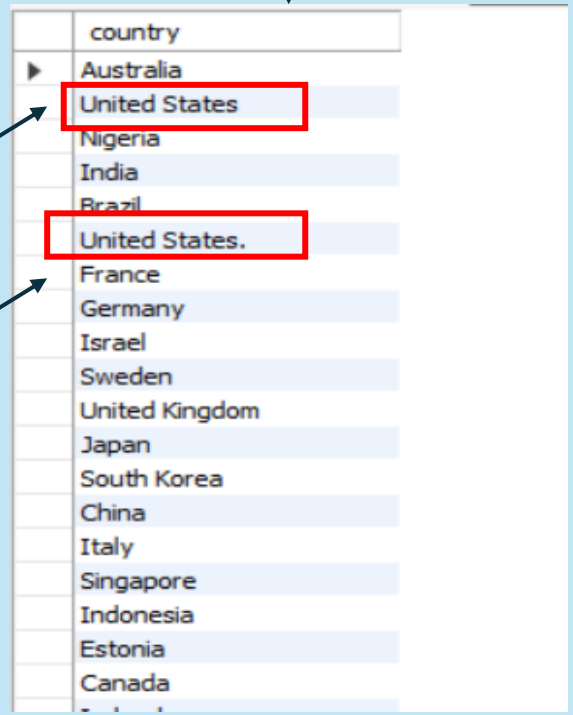
industry
Travel
Crypto
Finance
Data
Sales
Infrastructure
Hardware
Product
Construction
Legal
Energy
Manufacturing
Recruiting
Aerospace
Crypto Currency

These two are same , spelling mistake is there

Use this query to Fix it.

```sql
update copy_global_layoff
set industry = 'Crypto'
where industry like 'Crypto%'
```

```sql
select distinct country
from copy_global_layoff;
```



country
Australia
United States
Nigeria
India
Brazil
United States.
France
Germany
Israel
Sweden
United Kingdom
Japan
South Korea
China
Italy
Singapore
Indonesia
Estonia
Canada

Spelling mistake at the end of United States there is dot

Use this query to Fix it.

```sql
update copy_global_layoff
set country = 'United States'
where country = 'United States.';
```

```
desc copy_global_layoff
```

Field	Type	Null	Key	Default	Extra
Company	text	YES		NULL	
Location_HQ	text	YES		NULL	
Industry	text	YES		NULL	
laid_off	text	YES		NULL	
Date	text	YES		NULL	
Funds_Raised	double	YES		NULL	
Stage	text	YES		NULL	
Country	text	YES		NULL	
Percentage	text	YES		NULL	

Here these two-column datatype is wrong Laid_off should be integer and Date column datatype should be date.

```
alter table copy_global_layoff
modify column laid_off int;
```

Use this query to Fix it.

```
alter table copy_global_layoff
modify column `date` date;
```

Fixed

Field	Type	Null	Key	Default	Extra
Company	text	YES		NULL	
Location_HQ	text	YES		NULL	
Industry	text	YES		NULL	
Laid_Off	int	YES		NULL	
date	date	YES		NULL	
Funds_Raised	double	YES		NULL	
Stage	text	YES		NULL	
Country	text	YES		NULL	
Percentage	text	YES		NULL	

- **Step 4 :** Removing Null or Blank values

In this dataset there were no (Null) values but there are (Blank) values in Laid_off and percentage columns. But the thing was if there is a blank value in Laid_off column then there was no blank value in percentage column and vice versa. So, I decided to check whether both columns have blank value in the same row. If yes, then those records are of no use because there are no values then how can any own gather information from those records. So, I remove those records

```
select * from copy_global_layoff
where laid_off = '' and
Percentage = '';
```

Use this query to check blank values

Company	Location_HQ	Industry	laid_off	Date	Funds_Raised	Stage	Country	Percentage
Jasper Health	Boise	Healthcare		2024-05-31	32	Series A	United States	
Fisker	Los Angeles	Transportation		2024-05-29	1700	Post-IPO	United States	
Funding Circle	London	Finance		2024-05-29	746	Post-IPO	United Kingdom	
Hopin	London	Other		2024-05-07	1000	Series D	United Kingdom	
Google	SF Bay Area	Consumer		2024-04-30	26	Post-IPO	United States	
Fisker	Los Angeles	Transportation		2024-04-29	1700	Post-IPO	United States	
Grin	Sacramento	Marketing		2024-04-25	145	Series B	United States	
Expedia	Austin	Travel		2024-04-24	3300	Post-IPO	United States	
98point6	Seattle	Healthcare		2024-04-23	299	Acquired	United States	
Homie	Salt Lake City	Real Estate		2024-04-19	35	Series B	United States	
ConnectWise	Tampa Bay	Other		2024-04-17	1	Acquired	United States	
Google	SF Bay Area	Consumer		2024-04-17	26	Post-IPO	United States	
Agility Robotics	Portland	Other		2024-04-04	179	Series B	United States	
Amazon	Seattle	Retail		2024-04-03	108	Post-IPO	United States	
Kaseya	Miami	Security		2024-04-03	547	Unknown	United States	
New Relic	SF Bay Area	Infrastructure		2024-04-03	214	Acquired	United States	
Lentra	Pune	Finance		2024-04-02	104	Series B	India	
Verily	SF Bay Area	Healthcare		2024-03-22	3500	Subsidiary	United States	
Cybereason	Boston	Security		2024-03-20	750	Series F	United States	
Totango	SF Bay Area	Support		2024-03-07	147	Acquired	United States	

```sql
delete from copy_global_layoff
where laid_off = '' and
Percentage = '';
```


This delete all blank values from dataset. These records were of no use because how can we show information with blank values. So, removing blank values are the best solution I can say.

Company	Location_HQ	Industry	Laid_Off	date	Funds_Raised	Stage	Country	Percentage
&Open	Dublin	Marketing	9	2022-11-17	35	Series A	Ireland	0.09
#Paid	Toronto	Marketing	19	2023-01-27	21	Series B	Canada	0.17
10X Genomics	SF Bay Area	Healthcare	100	2022-08-04	242	Post-IPO	United States	0.08
1stdibs	New York City	Retail	70	2020-04-02	253	Series D	United States	0.17
23andMe	SF Bay Area	Healthcare	71	2023-08-08	1100	Post-IPO	United States	0.11
23andMe	SF Bay Area	Healthcare	75	2023-06-09	1100	Post-IPO	United States	0.09
2TM	Sao Paulo	Crypto	100	2022-09-01	250	Unknown	Brazil	0.15
2TM	Sao Paulo	Crypto	90	2022-06-01	250	Unknown	Brazil	0.12
54gene	Washington D.C.	Healthcare	95	2022-08-29	44	Series B	United States	0.3
6sense	SF Bay Area	Sales	150	2022-10-12	426	Series F	United States	0.1

During the cleaning process, 2 duplicate records were removed; 11 records were having extra spaces, so they were trimmed; spelling errors were found in 2 records and then corrected. Corrected datatypes for Laid_off and date columns and removed the records with blank values. Now this dataset is ready for further in-depth analysis.

Moving To Analysis

Building on the cleaned dataset prepared, this section further analyzes the layoff trend across the globe by industry, country, and time periods. The objective here is to extract out useful insights that will inform the trends and patterns for workforce reduction.

```
rename table copy_global_layoff to layoffs_analysis;
```

"Just renaming the table for my convenience."

```
select * from layoffs_analysis;
```

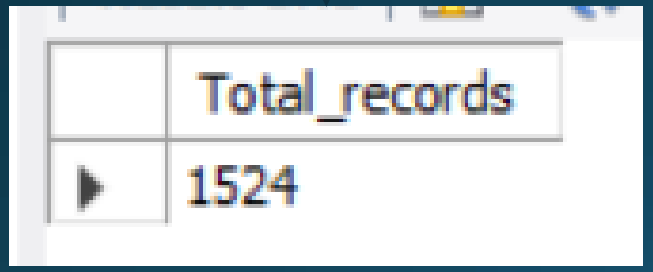
Company	Location_HQ	Industry	Laid_Off	date	Funds_Raised	Stage	Country	Percentage
&Open	Dublin	Marketing	9	2022-11-17	35	Series A	Ireland	0.09
#Paid	Toronto	Marketing	19	2023-01-27	21	Series B	Canada	0.17
10X Genomics	SF Bay Area	Healthcare	100	2022-08-04	242	Post-IPO	United States	0.08
1stdibs	New York City	Retail	70	2020-04-02	253	Series D	United States	0.17
23andMe	SF Bay Area	Healthcare	71	2023-08-08	1100	Post-IPO	United States	0.11
23andMe	SF Bay Area	Healthcare	75	2023-06-09	1100	Post-IPO	United States	0.09
2TM	Sao Paulo	Crypto	100	2022-09-01	250	Unknown	Brazil	0.15
2TM	Sao Paulo	Crypto	90	2022-06-01	250	Unknown	Brazil	0.12
54gene	Washington D.C.	Healthcare	95	2022-08-29	44	Series B	United States	0.3
6sense	SF Bay Area	Sales	150	2022-10-12	426	Series E	United States	0.1
7shifts	Saskatoon	Food	30	2023-09-15	131	Series C	Canada	0.07
7Shifts	Saskatoon	Food	68	2024-01-11	131	Series C	Canada	0.19
8x8	SF Bay Area	Support	155	2023-01-18	253	Post-IPO	United States	0.07
8x8	SF Bay Area	Support	200	2022-10-04	253	Post-IPO	United States	0.09
99	Sao Paulo	Transport...	75	2022-09-20	244	Acquired	Brazil	0.02
Abra	SF Bay Area	Crypto	12	2022-06-30	106	Series C	United States	0.05
Absci	Portland	Healthcare	30	2023-09-05	238	Post-IPO	United States	0.15
Acast	Stockholm	Media	70	2022-09-15	126	Post-IPO	Sweden	0.15
Acko	Mumbai	Finance	45	2020-04-01	143	Unknown	India	0.09
Ada	Toronto	Support	78	2022-09-20	190	Series C	Canada	0.16

"With the data cleaned and standardized, we're ready to begin our analysis."

Start with basic EDA

```
select count(*) from layoffs_analysis;
```

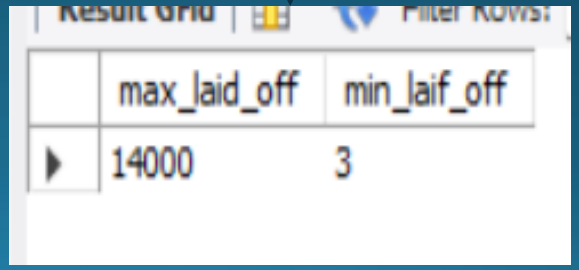
After cleaning total records present in the dataset



	Total_records
▶	1524

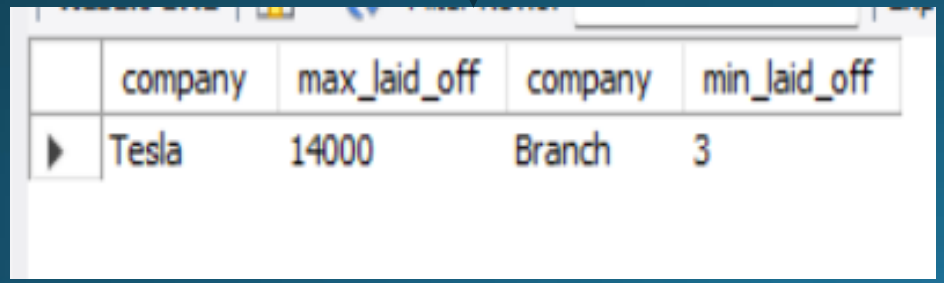
```
select max(laid_off) max_laid_off,min(laid_off) min_laif_off
from layoffs_analysis;
```

Maximum & Minimum Layoff in on go



	max_laid_off	min_laif_off
▶	14000	3

```sql
select max_company.company as company,max_company.max_laid_off,
min_company.company as company,min_company.min_laid_off
from (select company,max(laid_off) max_laid_off
from layoffs_analysis
group by company
order by max_laid_off desc
limit 1) as max_company
join
(select company,min(laid_off) min_laid_off
from layoffs_analysis
group by company
order by min_laid_off
limit 1) as min_company
on 1 = 1;
```



company	max_laid_off	company	min_laid_off
Tesla	14000	Branch	3

Name of the companies who laid off maximum & minimum in one go.

```sql
select company,sum(laid_off) total_laid_off
from layoffs_analysis
group by company
order by total_laid_off desc;
```

Company wise total laid off

company	total_laid_off
Amazon	18000
Tesla	14000
Google	12000
Meta	11000
SAP	11000
Salesforce	10700
Microsoft	10000
Ericsson	8500
Cisco	8350
Flink	8100
Uber	7525
Micron	7200
PayPal	4500
Peloton	4484
Wayfair	4270
Carvana	4000
Better.com	3900
Twitter	3900
Groupon	3300
Shopify	3300

Till now Amazon has layoffs the most

```sql
select industry,sum(laid_off) total_laid_off
from layoffs_analysis
group by industry
order by total_laid_off desc;
```

Industry wise total laid off

industry	total_laid_off
Transportation	49834
Retail	49454
Other	47341
Consumer	44626
Finance	31545
Food	30922
Real Estate	14831
Healthcare	14677
Sales	14346
Travel	13313
Infrastructure	11165
Hardware	10920
Crypto	10581
Education	9126
Fitness	8728
Marketing	7806
Security	7392
HR	6674
Media	6571
Data	4518

Transportation industry appears to be heavily impacted by layoffs

```sql
select country,sum(laid_off) total_laid_off
from layoffs_analysis
group by country
order by total_laid_off desc;
```

Country wise total laid off

country	total_laid_off
United States	299385
India	27537
Germany	22793
Sweden	12442
United Kingdom	11888
Canada	8312
Brazil	6929
Singapore	5090
Israel	4515
Indonesia	2721
Australia	2614
France	1040
New Zealand	1025
United Arab E...	995
Kenya	982
Nigeria	970
China	755
Hong Kong	730
Ireland	416
Russia	400

Huge workforce reduction in USA till now

```sql
select stage,sum(laid_off) total_laid_off
from layoffs_analysis
group by stage
order by total_laid_off desc;
```

Company stage wise total laid off

stage	total_laid_off
Post-IPO	244064
Unknown	29901
Acquired	26599
Series B	23767
Series D	19940
Series C	16984
Series E	15342
Series F	9070
Private Equity	7982
Series H	6781
Series A	4403
Series G	4352
Series J	2350
Series I	2055
Seed	1253
Subsidiary	714

Post-IPO are experiencing higher layoffs

```sql
select company,sum(Funds_Raised) total_funds_raised,sum(laid_off) total_laid_off
from layoffs_analysis
group by company,Funds_Raised
order by total_Funds_Raised desc;
```

company	total_funds_raised	total_laid_off
Netflix	243800	450
Uber	98800	7525
Xerox	27200	3000
Meta	26000	11000
Twitter	25800	3900
Tesla	20200	14000
Lyft	19600	2814
WeWork	19500	100
Robinhood	16800	1203
Lucid Motors	16600	1700
Grab	16500	1000
Cruise	15000	900
Flipkart	12900	1100
Byju's	11000	3000
Swiggy	10800	1030
Rivian	10700	840
Gopuff	10200	2050
Delivery Hero	9900	156
Grab	9900	360
Snap	9800	1780

Netflix raised the highest amount of funds, but it had a relatively low number of layoffs compared to others.

Comparing company's funds raised and their total layoffs

```sql
select year(date) Years, sum(Laid_Off) total_laid_off
from layoffs_analysis
group by years
order by years;
```

Layoffs over the years

Years	total_laid_off
2020	60960
2021	6490
2022	126502
2023	158363
2024	63242

In 2023, layoffs were at their highest

```sql
with cte as (select substring(date,1,7) Months, sum(laid_off) Total_laid_off
from layoffs_analysis
group by months
order by months)
select months,total_laid_off,sum(total_laid_off)
over(order by months) rolling_total
from cte;
```

months	total_laid_off	rolling_total
2020-03	7241	7241
2020-04	21064	28305
2020-05	21404	49709
2020-06	6442	56151
2020-07	2198	58349
2020-08	1853	60202
2020-09	339	60541
2020-10	110	60651
2020-11	189	60840
2020-12	120	60960
2021-01	516	61476
2021-02	332	61808
2021-04	160	61968
2021-06	2434	64402
2021-08	37	64439
2021-09	41	64480
2021-11	2070	66550
2021-12	900	67450
2022-01	80	67530
2022-02	3385	70915

End of 2020

End of 2021

Rolling total layoffs by month

```sql
select company,substring(date,1,7) Months, sum(Laid_Off) total_laid_off
from layoffs_analysis
group by company,months
having months >= '2024-01'
order by total_laid_off desc;
```

Tesla is having maximum layoffs in 2024

company	Months	total_laid_off
Tesla	2024-04	14000
SAP	2024-01	8000
Cisco	2024-02	4250
Xerox	2024-01	3000
PayPal	2024-01	2500
Farfetch	2024-02	2000
Unity	2024-01	1800
Wayfair	2024-01	1650
Expedia	2024-02	1500
Flipkart	2024-01	1100
Block	2024-01	1000
Citrix	2024-01	1000
eBay	2024-01	1000
Indeed	2024-05	1000
SolarEdge	2024-01	900
Vacasa	2024-05	800
Vroom	2024-01	800
Salesforce	2024-01	700
Electronic Arts	2024-02	670
Take-Two	2024-04	579

Company wise Total layoffs in 2024(Current year)

```sql
select
sum(case
    when date <'2024-01-01' then laid_off
    else 0
end) as Before_2024,

sum(case
    when date >= '2024-01-01' then laid_off
    else 0
end) as After_2024
from layoffs_analysis;
```

Total layoffs before & after 1st Jan 2024. And the layoffs after 1st Jan 2024 is till 5th of June 2024 as by dataset.

	Before_2024	After_2024
▶	352315	63242

```sql
with company_year as (select company, year(date) years, sum(laid_off) total_laid_off
from layoffs_analysis
group by company,years),
company_year_rank as
(select *,dense_rank () over(partition by years order by total_laid_off desc) as ranking
from company_year)
select * from company_year_rank
where ranking <=5;
```

company	years	total_laid_off	ranking
Uber	2020	7525	1
Groupon	2020	2800	2
Airbnb	2020	1900	3
PaisaBazaar	2020	1500	4
Swiggy	2020	1450	5
Katerra	2021	2434	1
Zillow	2021	2000	2
Better.com	2021	900	3
Dropbox	2021	315	4
Bounce	2021	200	5
Meta	2022	11000	1
Amazon	2022	10000	2
Cisco	2022	4100	3
Peloton	2022	4084	4
Carvana	2022	4000	5
Google	2023	12000	1
Microsoft	2023	10000	2
Ericsson	2023	8500	3
Flink	2023	8100	4
Amazon	2023	8000	5

In every year Top 5 companies who layoff maximum

Findings

- This analysis meets our objective of identifying layoff trends by showing that the U.S. and Post-IPO companies are the most affected.
- Our analysis indicated that even with high fundraising, companies like Meta, Tesla, Uber, Twitter, and others still had major layoffs.
- Layoffs were highest in 2023, with the transportation industry highly affected.
- Also found that Tesla had the maximum number of layoffs in a single day in 2024.

That’s all for this Project

Thank You