A. WORLD LAYOFFS PROJECT IN 2020 TO 2023

Project ini untuk melihat layoff perusahaan yang ada didunia pada tahun 2020 – 2023, project ini terdiri dari beberapa tahapan yaitu data cleaning dengan menggunakan MySQL, explorasi data analysis dengan MySQL dan visualisasi dengan Tableau.

1. Data Cleaning

Pada tahapan ini terdiri dari beberapan tahapan yaitu:

- Remove Duplicate
- Standardize the Data
- Null values and blank values

Scripts MySQL:

```
# Data Cleaning
# 1. Remove Duplicate
select*
from world_layoffs.layoffs_project;

SELECT*,
ROW_NUMBER () OVER (PARTITION BY
company,location,total_laid_off,percentage_laid_off,'data') as row_num
FROM world_layoffs.layoffs_project;

WITH duplicate_cte as
(
SELECT*,
ROW_NUMBER () OVER (PARTITION BY company,
location, industry, percentage_laid_off, 'data', stage, country,
funds_raised_millions) as row_num
FROM world_layoffs.layoffs_project
)
select*
from duplicate_cte
where row_num > 1;

select*
from layoffs_project
where company = 'casper';

WITH duplicate_cte as
(
SELECT*,
ROW_NUMBER () OVER (PARTITION BY company,
location, industry, percentage_laid_off, 'data', stage, country,
funds_raised_millions) as row_num
FROM world_layoffs.layoffs_project
)
delete
from duplicate_cte
```

```
where row_num > 1;

CREATE TABLE `layoffs_project2` (
  `company` text,
  `location` text,
  `industry` text,
  `total_laid_off` int DEFAULT NULL,
  `percentage_laid_off` text,
  `date` text,
  `stage` text,
  `country` text,
  `funds_raised_millions` int DEFAULT NULL,
  `row_num` int
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

select*
from layoffs_project2
where row_num >1;

insert into layoffs_project2
SELECT*,
ROW_NUMBER () OVER (
PARTITION BY company,
location, industry, percentage_laid_off, 'data', stage, country,
funds_raised_millions) as row_num
FROM world_layoffs.layoffs_project;

select*
from layoffs_project2
where row_num >1;

# 2. Standardizing Data

# 2.1 trim / delete space
select distinct(TRIM(company))
from layoffs_project2;

select company, TRIM(company)
from layoffs_project2;

update layoffs_project2
set company = TRIM(company);

select distinct industry
from layoffs_project2
order by 1;

update layoffs_project2
set industry = 'Crypto'
where industry like 'Crypto%';

select*
from layoffs_project2
where industry like 'Crypto%';
```

```sql
select*
from layoffs_project2
where country like 'United States%'
order by 1;

# 2.2 delete titik '.' or function

select country, trim(trailing '.' from country)
from layoffs_project2
order by 1;

update layoffs_project2
set country = trim(trailing '.' from country)
where country like 'United States%';

select distinct country
from layoffs_project2
order by 1;

# 2.3 str_to_date

select `date`,
str_to_date(`date`, '%m/%d/%Y')
from layoffs_project2;

update layoffs_project2
set `date` = str_to_date(`date`, '%m/%d/%Y');

-- modif table / use ketika mengubah tipe pada table maka harus dirubah
selanjutnya tipe data pada table

alter table layoffs_project2
modify column `date` DATE;

# 3. Null Values and Blank Values / '' tobe null

# 3.1 change blank '' to null values

update layoffs_project2
set industry = null
where industry like '';

select*
from layoffs_project2
where total_laid_off is null
and percentage_laid_off is null;

select distinct industry
from layoffs_project2
where industry is null
or industry like '';

select*
```

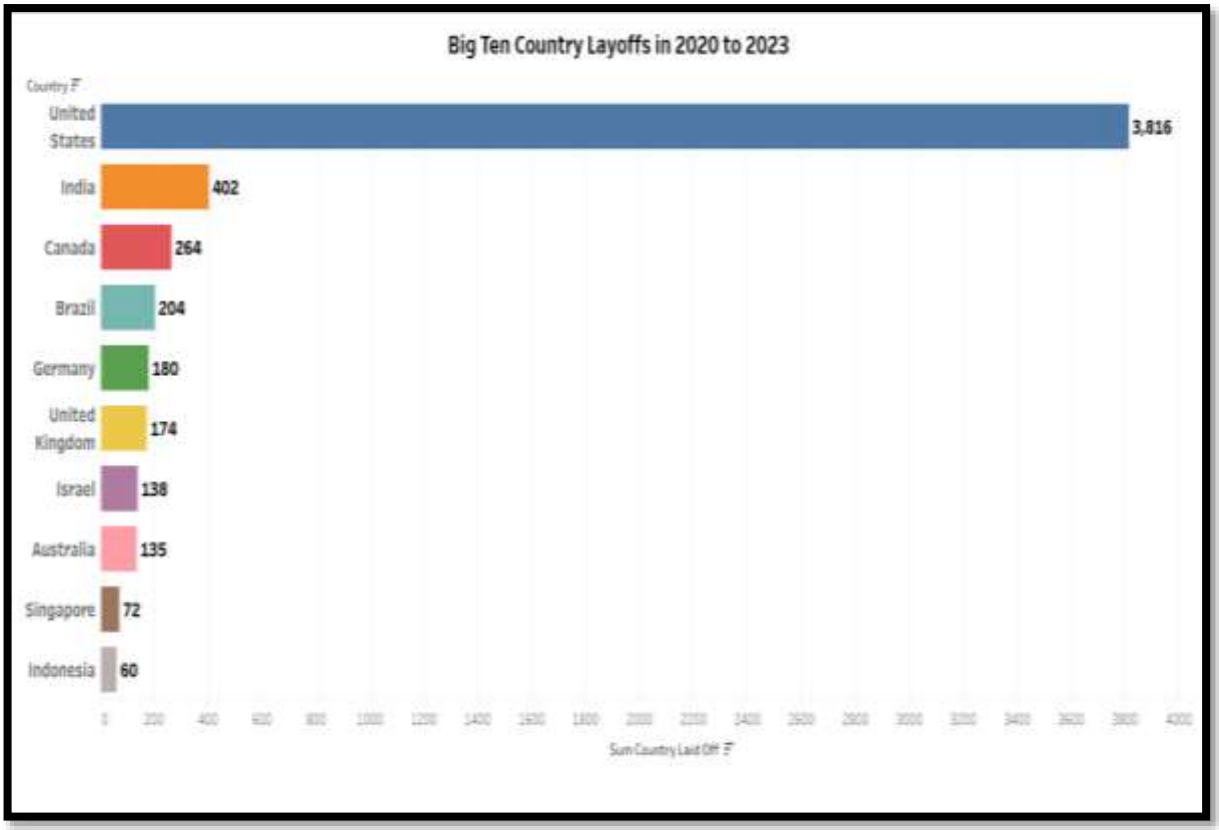
```
from layoffs_project2
where industry is null
or industry like '';

SELECT*
FROM layoffs_project2
WHERE company LIKE 'Bally%';

# 3.2 menggabungkan tabel blank with values table

select lp1.industry, lp2.industry
from layoffs_project2 lp1
join layoffs_project2 lp2
 on lp1.company = lp2.company
where (lp1.industry is null)
and lp2.industry is not null ;

update layoffs_project2 lp1
join layoffs_project2 lp2
      on lp1.company = lp2.company
set lp1.industry = lp2.industry
where lp1.industry is null
and lp2.industry is not null ;

select*
from layoffs_project2;



select*
from layoffs_project2
where ( industry like '');

-- delete the nul data

select*
from layoffs_project2
where total_laid_off is null
and percentage_laid_off is null;

delete
from  layoffs_project2
where total_laid_off is null
and percentage_laid_off is null;

select company, count(company)
from layoffs_project2
where total_laid_off is null
and percentage_laid_off is not null
group by company
order by 2;

-- menghapus column bantu / row_num column
```

```
select*
from layoffs_project2;

alter table layoffs_project2
drop column row_num;
```

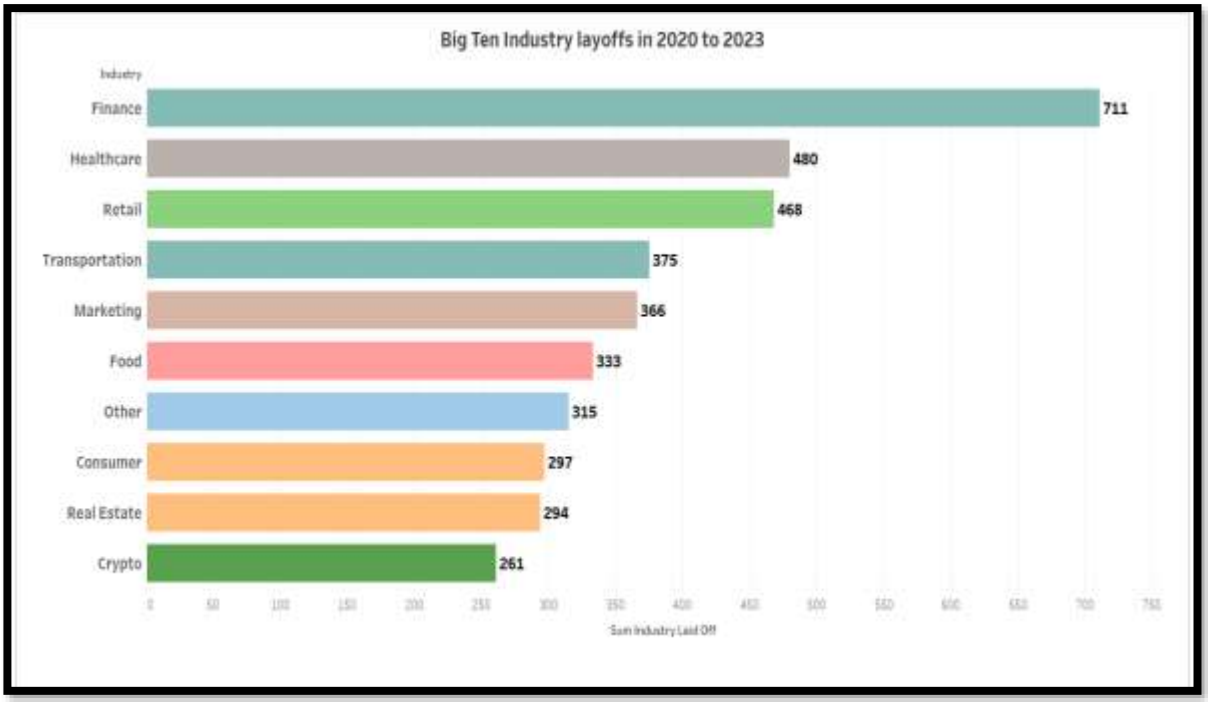
2. Exploratory Data Analyst with MySQL

```
# Negara dengan total layoffs perusahaan terbanyak di dunia pada 2020 to 2023
select country, count(country) as sum_country_laid_off
from layoffs_project2
group by country
order by 2 desc;

# industry dengan total layoffs perusahaan terbanyak di dunia pada 2020 to 2023
select industry, count(industry) as sum_industry_laid_off
from layoffs_project2
group by industry
order by 2 desc;

# industry rank dari country dengan layoff terbanyak ("United States")
with United_State as
(
select country, industry, count(industry) as sum_industry
from layoffs_project2
where country = "United States"
group by country, industry
)
select*, dense_rank() over(order by sum_industry desc)  as
United_States_Industry_Rank
from United_State;

# Negara dengan total_laid_off perusahaan terbanyak di dunia pada 2020 to 2023
select country, sum(total_laid_off)
from layoffs_project2
group by country
order by 2 desc;

# Industry dengan total laid_off perusahaan terbanyak di dunia pada 2020 to 2023
select industry, sum(total_laid_off)
from layoffs_project2
group by industry
order by 2 desc;

# big three country total_laid in years in 2020 to 2023
with country_total_laid_in_year as
(
select country, year(`date`) as years, sum(total_laid_off) total_laid
from layoffs_project2
group by country, year(`date`)
), country_total_laid_in_year_2 as
(
select*, dense_rank() over(partition by years order by total_laid desc) as
country_rank
from country_total_laid_in_year
where years is not null
)
select*
from country_total_laid_in_year_2
where country_rank <=3;

# big three industry rank in years in 2020 to 2023
with industry_total_laid_in_year (industry, years, total_laid) as
(
select industry, year(`date`), sum(total_laid_off)
from layoffs_project2
group by industry, year(`date`)
), industry_total_laid_in_year_2 as
(
select*, dense_rank() over(partition by years order by total_laid desc) as
industry_rank
from industry_total_laid_in_year
where years is not null
)
select*
from industry_total_laid_in_year_2
where industry_rank <= 3;
```

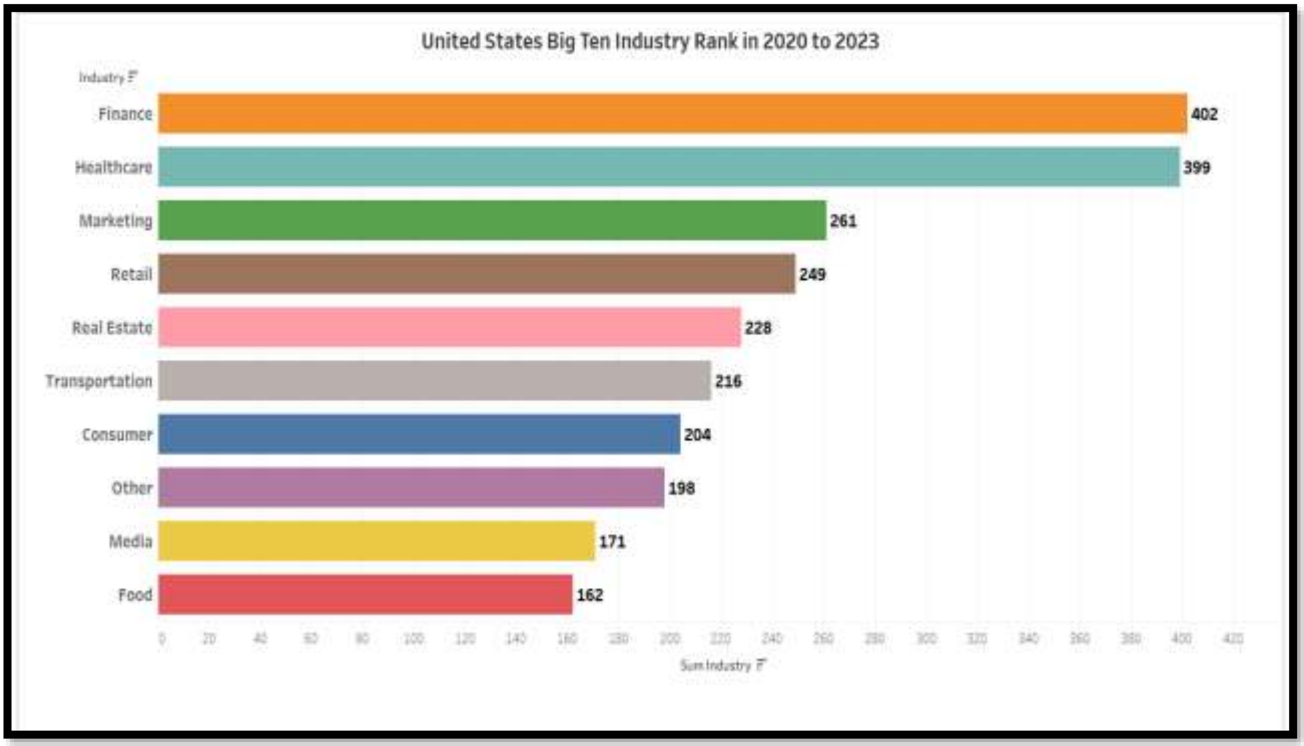
3. Visualization with Tableau

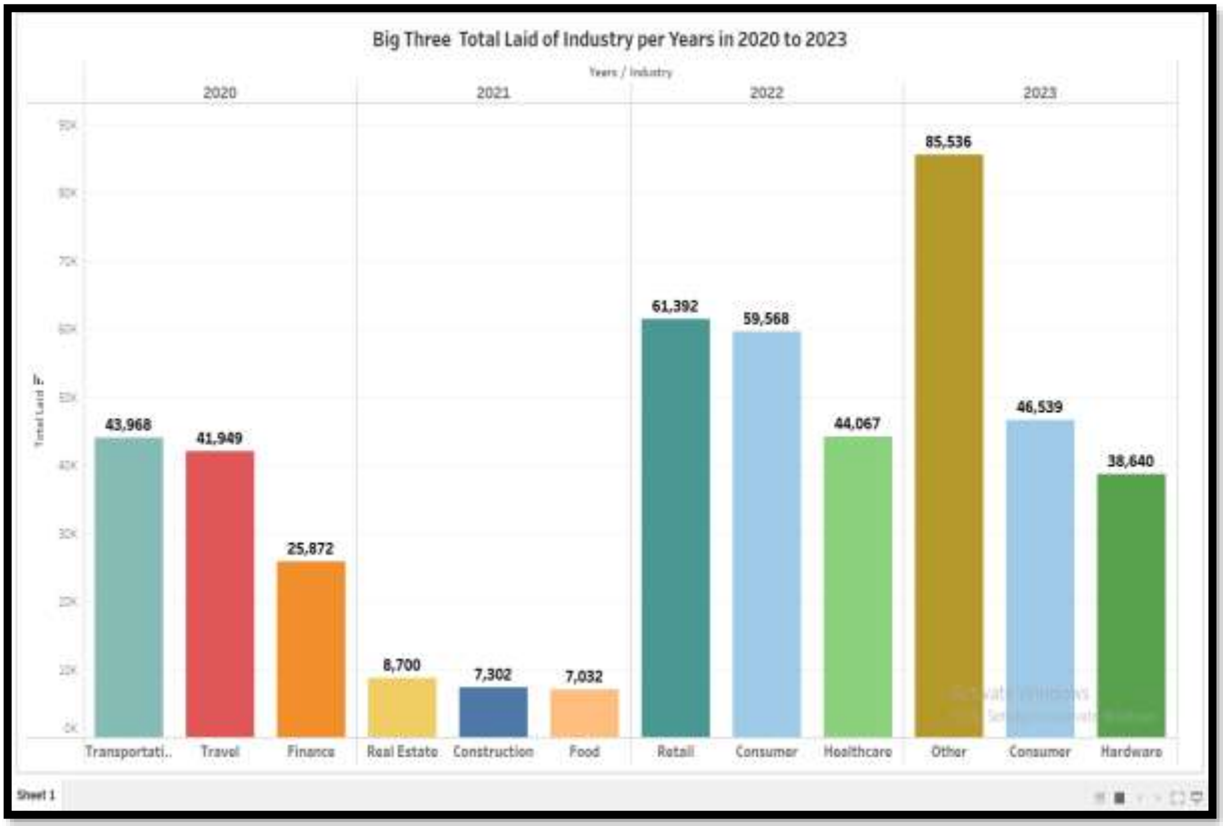


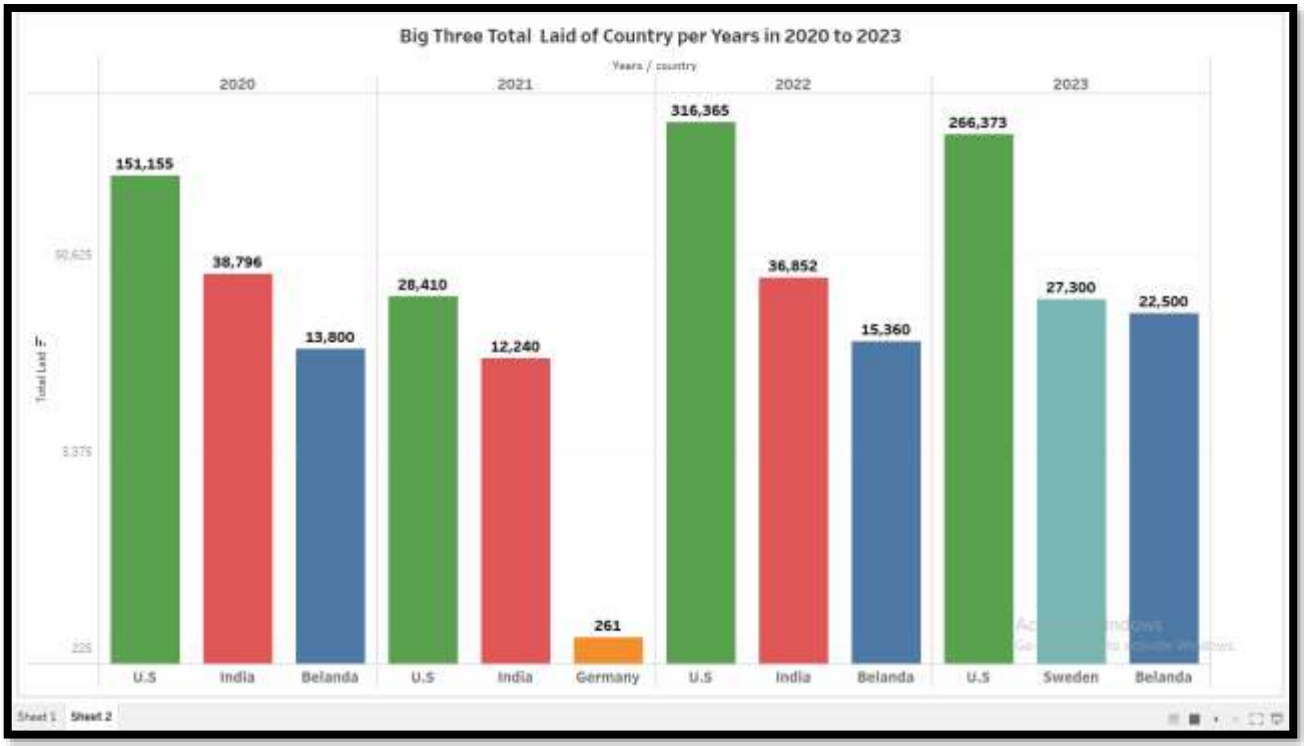
Big Ten Country Layoffs in 2020 to 2023



Big Ten Industry layoffs in 2020 to 2023


United States Big Ten Industry Rank in 2020 to 2023
Industry
Finance
402
Healthcare
399
Marketing
261
Retail
249
Real Estate
228
Transportation
216
Consumer
204
Other
198
Media
171
Food
162
0
20
40
60
80
100
120
140
160
180
200
220
240
260
280
300
320
340
360
380
400
420
Sum Industry

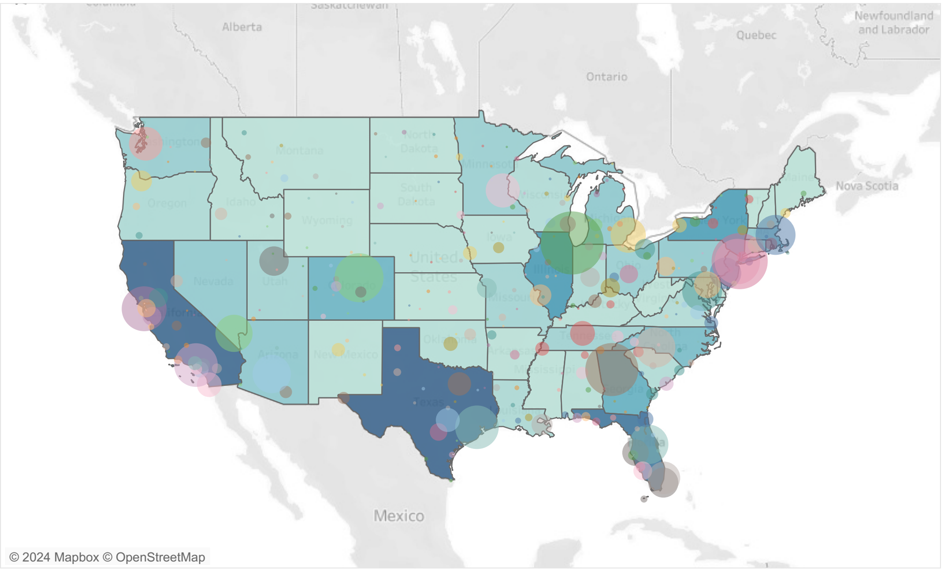

Big Three Total Laid of Industry per Years in 2020 to 2023
Years / Industry
2020
2021
2022
2023
Total Laid
90K
80K
70K
60K
50K
40K
30K
20K
10K
0K
43,968
41,949
25,872
8,700
7,302
7,032
61,392
59,568
44,067
85,536
46,539
38,640
Transportati..
Travel
Finance
Real Estate
Construction
Food
Retail
Consumer
Healthcare
Other
Consumer
Hardware
Sheet 1


Big Three Total Laid of Country per Years in 2020 to 2023
Years / country
2020
2021
2022
2023
Total Laid
50,625
3,375
225
151,155
38,796
13,800
28,410
12,240
261
316,365
36,852
15,360
266,373
27,300
22,500
U.S
India
Belanda
U.S
India
Germany
U.S
India
Belanda
U.S
Sweden
Belanda
Sheet 1
Sheet 2

Analyzing Aircraft Delay Data at US Airports in 2022

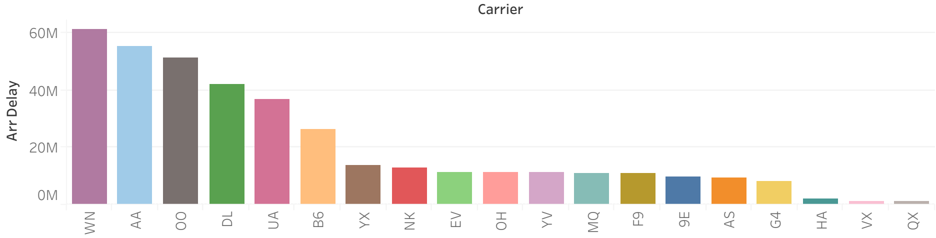
Arr Delay	Carrier Delay	Late Aircraft Delay	Nas Delay	Security Delay	Weather Delay
386,124,672	133,453,066	145,618,944	84,799,404	745,567	21,506,968

United State

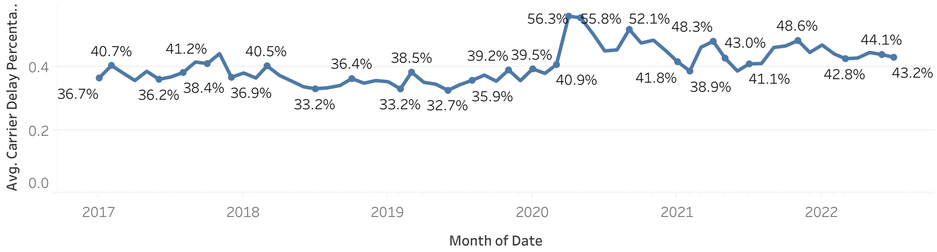


© 2024 Mapbox © OpenStreetMap

Carrier



Carrier Delay Percentage


Avg. Carrier Delay Percenta..
Month of Date