🚀 SQL Data Cleaning Project: Transforming Raw Data for Meaningful Insights! 💡

In this project, I took messy layoff data and turned it into a **structured, clean dataset** using **MySQL**. The entire workflow included:

- 🔹 **Data Loading** – Imported raw CSV data into MySQL Workbench.
- 🔹 **Removing Duplicates** – Used `ROW_NUMBER()` and partitioning to detect duplicate entries, ensuring data integrity.
- 🔹 **Standardization** – Applied functions like `TRIM()` to remove unwanted spaces, corrected industry naming inconsistencies, and formatted dates with `STR_TO_DATE()`.
- 🔹 **Handling Missing Values** – Identified NULL values, updated necessary fields, and removed unusable records to enhance accuracy.
- 🔹 **Dropping Unnecessary Columns** – Cleaned up the dataset by removing redundant information.
- 🔹 **Exporting the Clean Data** – Delivered the refined dataset in a **CSV format**, ready for analysis and visualization.

This project reinforced the power of **SQL** in data transformation, ensuring reliable insights and efficient processing! ✅

Let's perform data cleaning

```
use world_layoffs_project
select*from layoffs;

CREATE TABLE layoffs_staging
LIKE layoffs;

insert layoffs_staging
select*from layoffs;

select*,
row_number() over(
partition by
company,location,industry,total_laid_off,percentage_laid_o
ff,`date`,stage,country,funds_raised_millions) as row_num
from layoffs_staging;

with duplicates_cte as
(
select*,
row_number() over(
```

```sql
partition by
company,location,industry,total_laid_off,percentage_laid_o
ff,`date`,stage,country,funds_raised_millions) as row_num
from layoffs_staging
)
select*from duplicates_cte
where row_num>1;

select*from layoffs_staging
where company='casper';

create table layoffs_staging2(
company text,
location text,
industry text,
total_laid_off text,
percentage_laid_off text,
date text,
stage text,
country text,
funds_raised_millons int default null,
row_num int
);

insert into layoffs_staging2
select*,
row_number() over(
partition by
company,location,industry,total_laid_off,percentage_laid_o
ff,`date`,stage,country,funds_raised_millions) as row_num
from layoffs_staging;

select*from layoffs_staging2
where row_num>1;

describe layoffs_staging2;

delete from layoffs_staging2
where row_num>1;
set sql_safe_updates=0;

select*from layoffs_staging2
```

```sql
where row_num>1;

select company, trim(company)

select distinct industry from layoffs_staging2
order by 1;


SELECT COUNT(DISTINCT industry) as
unique_industry_count,COUNT(DISTINCT country) AS
unique_country_count
FROM layoffs_staging2;

select* from layoffs_staging2
where industry like "crypto%";

UPDATE layoffs_staging2
SET industry = 'Crypto'
WHERE industry IN ('Crypto Currency', 'CryptoCurrency');

select* from layoffs_staging2
where country like "United States%";

select distinct country, trim(trailing '.' from country)
from layoffs_staging2
order by 1;

update layoffs_staging2
set country=trim(trailing '.' from country)
where country like 'United states%';

update layoffs_staging2
set date=str_to_date(date,'%m/%d/%Y');

select date from layoffs_staging2;

alter table layoffs_staging2
modify column date Date;

select * from layoffs_staging2
where total_laid_off is null;
```

```sql
select * from layoffs_staging2
where total_laid_off is null
and
percentage_laid_off is null;

select t1.industry, t2.industry
from layoffs_staging2 t1
join layoffs_staging2 t2
on t1.company=t2.company
where (t1.industry is null or t1.industry='')
and t2.industry is not null;

update layoffs_staging2 t1
join layoffs_staging2 t2
on t1.company=t2.company
set t1.industry=t2.industry
where (t1.industry is null or t1.industry='')
and t2.industry is not null;

select *
from layoffs_staging2
where company like 'Bally%';

select *
from layoffs_staging2
where company like 'Airbnb%';

select * from layoffs_staging2
where total_laid_off is null
and
percentage_laid_off is null;

delete from layoffs_staging2
where total_laid_off is null
and
percentage_laid_off is null;

select* from layoffs_staging2;
```

```sql
update layoffs_staging2
set percentage_laid_off=0
where percentage_laid_off is null;

update layoffs_staging2
set total_laid_off=0
where total_laid_off is null;
```