-- EXPLORATORY DATA ANALYSIS

-- IN THIS PROJECT WE WILL BE RUNNING DIFFERENT QUERIES EXPLORING OUR DATA THAT WE HAVE CLEANED IN OUR PREVIOUS DATA CLEANING PROJECT

select \*

FROM layoffs\_staging2

select MAX(total\_laid\_off), MAX(percentage\_laid\_off)

FROM layoffs\_staging2;

select \*

from layoffs\_staging2

where percentage\_laid\_off = 1;

select \*

from layoffs\_staging2

where percentage\_laid\_off = 1

order by total\_laid\_off DESC;

select \*

from layoffs\_staging2

where percentage\_laid\_off = 1

order by funds\_raised\_millions DESC;

SELECT company, country, sum(total\_laid\_off)

from layoffs\_staging2

group by company, country

order by 3 desc;

select min(`date`), max(`date`)

from layoffs\_staging2

SELECT industry, sum(total\_laid\_off)

from layoffs\_staging2

group by industry

order by 2 desc;

select `date`, sum(total\_laid\_off)

from layoffs\_staging2

group by `date`

order by 1 desc;

select stage, sum(total\_laid\_off)

from layoffs\_staging2

group by stage

order by 1 desc;

SELECT company, sum(percentage\_laid\_off)

from layoffs\_staging2

group by company, country

order by 2 desc;

select substring(`date`,1,7) as `MONTH`, sum(total\_laid\_off)

from layoffs\_staging2

where substring(`date`,1,7) IS NOT NULL

group by `MONTH`

order by 1 asc;

WITH ROLLING\_TOTAL AS

(

select substring(`date`,1,7) as `MONTH`, sum(total\_laid\_off)

from layoffs\_staging2

where substring(`date`,1,7) IS NOT NULL

group by `MONTH`

order by 1 asc;

)

SELECT `MONTH`, total\_off

,sum(total\_off) over(order by `MONTH`) as rolling\_total

FROM Rolling\_Total;

SELECT company, sum(total\_laid\_off)

from layoffs\_staging2

group by company, country

order by 2 desc;

SELECT company, YEAR(`date`), sum(total\_laid\_off)

from layoffs\_staging2

group by company, YEAR(`date`);

ORDER BY 3 desc;

-- CREATING A CTE

with Company\_YEAR (company, years, total\_laid\_off) AS

(

select company, YEAR(`date`), SUM(total\_laid\_off)

FROM layoffs\_staging2

group by company, YEAR(`date`)

)

SELECT \*, DENSE\_RANK() OVER (PARTITION BY years ORDER BY total\_laid\_off desc) as Ranking

from Company\_Year

where years is not null

order by Ranking ASC;