SELECT \* FROM layoffs;

-- Remove Duplicates

-- Standardize the data

-- NUll values or blank values

-- Remove Any Columns

create table layoffs\_staging

like layoffs;

SELECT \* FROM layoffs\_staging;

-- INSERTING DATA FROM LAYOFFS TO LAYOFFS\_STAGING

INSERT layoffs\_staging

SELECT \* FROM layoffs;

SELECT \*,

row\_number() over(

partition by company, industry, total\_laid\_off, percentage\_laid\_off, 'date') As row\_num

FROM layoffs\_staging;

with duplicate\_cte as(

SELECT \*,

row\_number() over(

partition by company, location, industry, total\_laid\_off, percentage\_laid\_off, 'date',stage, country,funds\_raised\_millions) As row\_num

FROM layoffs\_staging

)select \*

From duplicate\_cte

where row\_num >1

-- CHECKING IF the companies are twice or more

-- SELECT \* FROM layoffs\_staging where company = "casper";

CREATE TABLE `layoffs\_staging2` (

`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\_staging2;

-- INSETING DATA FROM LAYOFF\_STAGING TO LAYOFF\_STAGING2

insert into layoffs\_staging2

select \*,

row\_number() over(

partition by company, location, industry, total\_laid\_off, percentage\_laid\_off, 'date',stage, country,funds\_raised\_millions) As row\_num

FROM layoffs\_staging;

SELECT \* FROM layoffs\_staging2;

delete

FROM layoffs\_staging2

WHERE row\_num > 1;

SELECT \* FROM layoffs\_staging2;

-- Standardizeing data

SELECT company, TRIM(company) FROM layoffs\_staging2;

UPDATE layoffs\_staging2

SET company = TRIM(company);

SELECT DISTINCT industry

FROM layoffs\_staging2

order by 1;

-- we can see crypto, crypto currency and different same industry name so lets check crypto

SELECT \*

FROM layoffs\_staging2

WHERE industry like 'Crypto%';

UPDATE layoffs\_staging2

set industry = 'Crypto'

WHERE industry like 'crypto%';

-- LOCATION Removing the '.' from two same row United state

select distinct country, trim(Trailing '.' from country)

from layoffs\_staging2

order by 1;

-- Now Updatign the US. row with use

update layoffs\_staging2

set country = trim(Trailing '.' from country)

where country like 'United States%';

select distinct country

from layoffs\_staging2

order by 1;

-- Date is in text form we have to do it in time format

Select `date`from layoffs\_staging2;

-- Update date in new form

update layoffs\_staging2

set `date` = str\_to\_date(`date`, '%m/%d/%Y')

-- Changing text definition of Date colomn to DATE

ALTER table layoffs\_staging2

Modify column `date` Date;

-- Select \* from layoffs\_staging2;

Select \*

from layoffs\_staging2

where total\_laid\_off is NULL

and percentage\_laid\_off is NULL;

-- making all blank industry types as NULL

UPDATE layoffs\_staging2

set industry = NULL

where industry = '';

-- Selecting industries which are NULL or Blank

select \*

from layoffs\_staging2

where industry is NULL

or industry = '';

-- Checking if black industry company have another row

Select \* from layoffs\_staging2

where company = 'Airbnb';

-- IF company has different layoffs then industry should be same

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;

-- Updating the industry of same company with same industry type

update layoffs\_staging2 t1

join layoffs\_staging2 t2

on t1.company = t2.company

set t1.industry = t2.industry

where t1.industry is NULL

and t2.industry is NOT NULL;

-- we check if any company has industry NULL and Ballys comes so fix it

select \*

from layoffs\_staging2

where company like 'Bally%'; -- only one row with null row

-- Data cleaning for Total\_laid off and percentage\_laid\_off

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;

-- We dont need the row\_num column anymore so remove it "Drop a column"

alter table layoffs\_staging2

drop column row\_num;