Data Cleaning Using SQL

- Creating a Database to import a data table in this database

```
CREATE DATABASE data_cleaning_eda;
```

- importing loptop dataset manually by clicking table import wizard

```
SELECT * FROM data_cleaning_eda.laptop;
```

- STEP-1 : create a backup or copy of the data

```
USE data_cleaning_eda;
CREATE TABLE laptop_backup LIKE laptop;

INSERT INTO data_cleaning_eda.laptop_backup
SELECT * FROM data_cleaning_eda.laptop;

SELECT * FROM data_cleaning_eda.laptop_backup;
```

- STEP-2 : Count number of rows in the dataset

```
SELECT COUNT(*) FROM data_cleaning_eda.laptop;
```

- STEP-3 : Check memory consumtion for reference

```
SELECT * FROM information_schema.TABLES
WHERE TABLE_SCHEMA = 'data_cleaning_eda'
AND TABLE_NAME = 'laptop';
```

```
SELECT DATA_LENGTH AS 'bytes_length', DATA_LENGTH/1024 AS 'kb'
FROM information_schema.TABLES
WHERE TABLE_SCHEMA = 'data_cleaning_eda'
AND TABLE_NAME = 'laptop';
```

- How to rename a Column Name

```
ALTER TABLE `data_cleaning_eda`.`laptop`
CHANGE COLUMN `Unnamed: 0` `index` INT NULL DEFAULT NULL ;

SELECT * FROM data_cleaning_eda.laptop;
```

- Drop rows where all column values are null

```
DELETE FROM data_cleaning_eda.laptop
WHERE `index` IN(SELECT * FROM data_cleaning_eda.laptop WHERE
`index` IS NULL AND Company IS NULL AND TypeName IS NULL AND Inches IS NULL
AND ScreenResolution IS NULL AND Cpu IS NULL AND Ram IS NULL AND Memory IS NULL
AND Gpu IS NULL AND OpSys IS NULL AND Weight IS NULL AND Price IS NULL);
```

- # Drop Duplicates in SQL : You can do that by groupby, windows function and all
- # Applying DISTINCT() function to see categorical columns all item

```
SELECT DISTINCT(Company) FROM data_cleaning_eda.laptop;
SELECT DISTINCT(TypeName) FROM data_cleaning_eda.laptop;
```

- # How to modify DataTypes of a particular column

```
-- modifying `Inches` column
ALTER TABLE data_cleaning_eda.laptop
MODIFY COLUMN Inches DECIMAL(10, 1);

-- modify Price Column
ALTER TABLE data_cleaning_eda.laptop
MODIFY COLUMN Price DECIMAL(10, 1);
SELECT * FROM data_cleaning_eda.laptop;
```

- # Modify values from Ram column. Every Ram are like 4GB, 8GB, 16GB convert them into 4, 8, 16

```
UPDATE data_cleaning_eda.laptop
SET Ram = REPLACE(Ram, 'GB', '');
SELECT * FROM data_cleaning_eda.laptop;
```

- now modify ram DataTypes

```
ALTER TABLE data_cleaning_eda.laptop
MODIFY COLUMN Ram INTEGER;
```

- # Modify values from Weight column. Every Weight are like 2kg, 1.37 kg. convert them into 2, 1.37

```
SELECT Weight, REPLACE(Weight, 'kg', '') FROM data_cleaning_eda.laptop;

UPDATE data_cleaning_eda.laptop
SET Weight = REPLACE(Weight, 'kg', '');
```

- # now modify weight DataTYpes

```
ALTER TABLE data_cleaning_eda.laptop
MODIFY COLUMN Weight DECIMAL(10, 2);
-- this code is not running because weight have some non numerical value lets handle it
UPDATE data_cleaning_eda.laptop
SET Weight = NULL
WHERE Weight NOT REGEXP '^[0-9]+(\.[0-9]+)?$';

-- now run the previous code
ALTER TABLE data_cleaning_eda.laptop
MODIFY COLUMN Weight DECIMAL(10, 2);
```

- # Modify Operating System (OpSys)

```
-- view query
SELECT OpSys,
CASE
   WHEN OpSys LIKE '%mac%' THEN 'macos'
   WHEN OpSys LIKE '%Windows%' THEN 'windows'
   WHEN OpSys LIKE '%linux%' THEN 'linux'
   WHEN OpSys LIKE 'No OS' THEN 'N/A'
   ELSE 'other'
END AS 'Operating_system'
FROM data_cleaning_eda.laptop;
```

```
-- lets update operating_system
UPDATE data_cleaning_eda.laptop
SET OpSys = CASE
    WHEN OpSys LIKE '%mac%' THEN 'macos'
    WHEN OpSys LIKE '%Windows%' THEN 'windows'
    WHEN OpSys LIKE '%linux%' THEN 'linux'
    WHEN OpSys LIKE 'No OS' THEN 'N/A'
    ELSE 'other'
END;
SELECT * FROM data_cleaning_eda.laptop;
```

- # Create 2 new column(gpu_brand, gpu_name) from GPU column

```
-- lets at first create 2 new column
ALTER TABLE data_cleaning_eda.laptop
ADD COLUMN gpu_brand VARCHAR(255) AFTER Gpu,
ADD COLUMN gpu_name VARCHAR(255) AFTER gpu_brand;

-- Updating gpu_brand column
SELECT Gpu, SUBSTRING_INDEX(Gpu, ' ', 1) FROM data_cleaning_eda.laptop;
UPDATE data_cleaning_eda.laptop
SET gpu_brand = SUBSTRING_INDEX(Gpu, ' ', 1);

-- Updating gpu_name column
SELECT Gpu, gpu_brand, REPLACE(Gpu, gpu_brand, ' ') FROM data_cleaning_eda.laptop;
UPDATE data_cleaning_eda.laptop
SET gpu_name = REPLACE(Gpu, gpu_brand, ' ');
```

```
-- Now delete / drop Gpu column
ALTER TABLE data_cleaning_eda.laptop DROP COLUMN Gpu;

SELECT * FROM data_cleaning_eda.laptop;
```

- # Create 3 new column(cpu_brand, cpu_name, cpu_speed) from Cpu column

```
-- creating 3 new column first
ALTER TABLE data_cleaning_eda.laptop
ADD COLUMN cpu_brand VARCHAR(255) AFTER Cpu,
ADD COLUMN cpu_name VARCHAR(255) AFTER cpu_brand,
ADD COLUMN cpu_speed DECIMAL(10, 1) AFTER cpu_name;

-- extract cpu_brand info from Cpu column and insert into cpu_brand column
SELECT Cpu, SUBSTRING_INDEX(Cpu, ' ', 1) FROM data_cleaning_eda.laptop;
UPDATE data_cleaning_eda.laptop
SET cpu_brand = SUBSTRING_INDEX(Cpu, ' ',1);

-- extract cpu_speed info from Cpu column and insert into cpu_speed column
SELECT Cpu, REPLACE(SUBSTRING_INDEX(Cpu, ' ', -1),'GHz',' ') FROM data_cleaning_eda.laptop;
UPDATE data_cleaning_eda.laptop
SET cpu_speed = REPLACE(SUBSTRING_INDEX(Cpu, ' ', -1),'GHz',' ');

-- extract cpu_name info from Cpu column and insert into cpu_name column
SELECT Cpu, REPLACE(REPLACE(Cpu,cpu_brand, ''), cpu_speed, ''),
REPLACE(REPLACE(Cpu,cpu_brand,''),SUBSTRING_INDEX(REPLACE(Cpu,cpu_brand,''),' ',-1),'')
FROM data_cleaning_eda.laptop;
```

```
UPDATE data_cleaning_eda.laptop
SET cpu_name =
REPLACE(REPLACE(Cpu,cpu_brand,''),SUBSTRING_INDEX(REPLACE(Cpu,cpu_brand,''),' ',-1),'');

-- Now delete / drop Cpu column
ALTER TABLE data_cleaning_eda.laptop DROP COLUMN Cpu;

SELECT * FROM data_cleaning_eda.laptop;
```

- # Screenresolution Column have multiple information, Extract all into diff column .alter

```
-- we will breakdown our screenresolution column into three main column

-- lets create 2 column first resolution_width, resolution_height
ALTER TABLE data_cleaning_eda.laptop
ADD COLUMN resolution_width INTEGER AFTER ScreenResolution,
ADD COLUMN resolution_height INTEGER AFTER resolution_width;

-- extracting both of information and updating
SELECT ScreenResolution,
SUBSTRING_INDEX(ScreenResolution, ' ', -1),
SUBSTRING_INDEX(SUBSTRING_INDEX(ScreenResolution, ' ', -1),'x',1) AS 'resolution_width',
SUBSTRING_INDEX(SUBSTRING_INDEX(ScreenResolution, ' ', -1),'x',-1) AS 'resolution_height'
FROM data_cleaning_eda.laptop;

UPDATE data_cleaning_eda.laptop
SET resolution_width = SUBSTRING_INDEX(SUBSTRING_INDEX(ScreenResolution, ' ', -1),'x',1),
   resolution_height = SUBSTRING_INDEX(SUBSTRING_INDEX(ScreenResolution, ' ', -1),'x',-1);
```

```
-- Create one more column from ScreenResolution that is tourch_scree or not
ALTER TABLE data_cleaning_eda.laptop
ADD COLUMN touch_screen INTEGER AFTER resolution_height;

SELECT ScreenResolution, ScreenResolution LIKE '%Touch%' FROM data_cleaning_eda.laptop;
UPDATE data_cleaning_eda.laptop
SET touch_screen = ScreenResolution LIKE '%Touch%';

SELECT * FROM data_cleaning_eda.laptop
```

- # Extract information from Memory column. I will breakdown my Memory column into 3 new column

```
-- Create 3 new column(memory_type, primary_storage, secondary_storage) from GPU column
SELECT * FROM data_cleaning_eda.laptop;
ALTER TABLE data_cleaning_eda.laptop
ADD COLUMN memory_type VARCHAR(255) AFTER Memory,
ADD COLUMN primary_storage INTEGER AFTER memory_type,
ADD COLUMN secondary_storage INTEGER AFTER primary_storage;
```

```
-- extracting and updating memory_type column from Memory
UPDATE data_cleaning_eda.laptop
SET memory_type = CASE
    WHEN Memory LIKE '%SSD%' AND Memory LIKE '%HDD%' THEN 'Hybrid'
    WHEN Memory LIKE '%SSD%' THEN 'SSD'
    WHEN Memory LIKE '%HDD%' THEN 'HDD'
    WHEN Memory LIKE '%Flash Storage%' THEN 'Flash Storage'
    WHEN Memory LIKE '%Hybrid%' THEN 'Hybrid'
    WHEN Memory LIKE '%Flash Storage%' AND Memory LIKE '%HDD%' THEN 'Hybrid'
    ELSE NULL
END ;

-- extracting and updating primary storage and secondary storage info from memroy
SELECT Memory,
REGEXP_SUBSTR(SUBSTRING_INDEX(Memory,' ', 1),'[0-9]+') AS 'primary_storage',
CASE WHEN Memory LIKE '%+%' THEN REGEXP_SUBSTR(SUBSTRING_INDEX(Memory, '+', -
1),'[0-9]+') ELSE 0 END AS 'secondary_storage'
FROM data_cleaning_eda.laptop;

UPDATE data_cleaning_eda.laptop
SET primary_storage = REGEXP_SUBSTR(SUBSTRING_INDEX(Memory,' ', 1),'[0-9]+'),
    secondary_storage = CASE WHEN Memory LIKE '%+%' THEN
REGEXP_SUBSTR(SUBSTRING_INDEX(Memory, '+', -1),'[0-9]+') ELSE 0 END;
```

```
-- Primary storage and secondary storage have tb like 1 2 lets convert it into gb

SELECT primary_storage, secondary_storage,

CASE WHEN primary_storage <= 2 THEN primary_storage*1024 ELSE primary_storage END,

CASE WHEN secondary_storage <= 2 THEN secondary_storage*1024 ELSE secondary_storage
END

FROM data_cleaning_eda.laptop;


UPDATE data_cleaning_eda.laptop

SET primary_storage = CASE WHEN primary_storage <= 2 THEN primary_storage*1024 ELSE
primary_storage END,

        secondary_storage = CASE WHEN secondary_storage <= 2 THEN
secondary_storage*1024 ELSE secondary_storage END;


SELECT * FROM data_cleaning_eda.laptop
```

- Saving The cleaned laptop Data

```
USE data_cleaning_eda;

CREATE TABLE cleaned_laptop LIKE data_cleaning_eda.laptop;


-- inserting clean data

INSERT INTO data_cleaning_eda.cleaned_laptop

SELECT * FROM data_cleaning_eda.laptop;
```