

Project- Acquisitions and IPOs

```
create database Project_SQL;
use Project_SQL;

--Cleaning Table(Removing Unwanted Columns)
ALTER TABLE acquisitions
DROP COLUMN column35, column36, column37, column38, column39, column40,
column41, column42, column43, column44,
column45, column46;

--Server Name "DESKTOP-K2S8TEA\PANKAJ"

/* Analysis 1 */
--"Distribution of acquirers who already did IPO vs who didn't"

select * from acquisitions;
select * from ipo;

acquisitions; --select and press alt+F1 to know information about acquisition table
IPO;         --select and press alt+F1 to know information about IPO table

--acquirers having ipos

--Base Code
select acquisitions.acquirer_uuid,* from acquisitions, IPO
where acquisitions.acquirer_uuid= IPO.org_uuid

--Modified Code
select acquisitions.acquirer_uuid,IPO.org_uuid, acquisitions.acquiree_name,
acquisitions.acquirer_name, acquisitions.acquirer_country_code,
IPO.type, acquisitions.price_usd,
IPO.money_raised
from acquisitions, IPO
where acquisitions.acquirer_uuid= IPO.org_uuid;

--acquirers not having ipos
--Base Code
select acquisitions.acquirer_uuid,* from acquisitions
where uuid not in (select acquisitions.uuid from acquisitions, IPO
where acquisitions.acquirer_uuid= IPO.org_uuid)

--Modified Code
select acquisitions.acquirer_uuid, acquisitions.acquiree_name,
acquisitions.acquirer_name, acquisitions.acquirer_country_code,
acquisitions.type, acquisitions.price_usd
from acquisitions
where uuid not in (select acquisitions.uuid from acquisitions, IPO
where acquisitions.acquirer_uuid= IPO.org_uuid);
```

```

/* Analysis 2 */
--"Regional Analysis of Acquirers and acquirees"
select acquirer_region, count(acquirer_region) as 'No. of Acquirer belongs'
from acquisitions group by acquirer_region;

select acquiree_region, count(acquiree_region) as 'No. of Acquiree belongs'
from acquisitions WHERE acquiree_region IS NOT NULL group by acquiree_region;

/* Analysis 3 */
--"Price analysis of different acquisitions- Make bins of price(Using SQL only) and see
in which bin max acquisitions lie as well"

SELECT
count(CASE WHEN price_usd >= 100000 AND price_usd < 100000000 THEN 1 END) AS '0.1M - 100M',
count(CASE WHEN price_usd >= 100000000 AND price_usd < 200000000 THEN 1 END) AS '100M - 200M',
count(CASE WHEN price_usd >= 200000000 AND price_usd < 300000000 THEN 1 END) AS '200M - 300M',
count(CASE WHEN price_usd >= 300000000 AND price_usd < 400000000 THEN 1 END) AS '300M - 400M',
count(CASE WHEN price_usd >= 400000000 AND price_usd < 500000000 THEN 1 END) AS '400M - 500M',
count(CASE WHEN price_usd >= 500000000 AND price_usd < 600000000 THEN 1 END) AS '500M - 600M',
count(CASE WHEN price_usd >= 600000000 AND price_usd < 700000000 THEN 1 END) AS '600M - 700M',
count(CASE WHEN price_usd >= 700000000 AND price_usd < 800000000 THEN 1 END) AS '700M - 800M'
FROM acquisitions AS Price_Bin

/* Analysis 4 */
--"Monthly analysis of number of acquisitions and IPOs"

--Months in String
SELECT DATENAME (MONTH, DATEADD(MONTH, MONTH(acquired_on) - 1, '1900-01-01')) Month,
COUNT(*) as 'Count of Acquisitions'
FROM acquisitions
GROUP BY MONTH(acquired_on);

SELECT DATENAME (MONTH, DATEADD(MONTH, MONTH(went_public_on) - 1, '1900-01-01')) Month,
COUNT(*) 'Count of IPOs'
FROM IPO
GROUP BY MONTH(went_public_on);

--Months in Number
SELECT MONTH(acquired_on) MONTH, COUNT(*) COUNT
FROM acquisitions
GROUP BY MONTH(acquired_on);

SELECT MONTH(went_public_on) MONTH, COUNT(*) COUNT
FROM IPO
GROUP BY MONTH(went_public_on);

```

```

/* Analysis 5 */
--"Analysis of social media presence of acquirers vs acquirees"

SELECT acquirer_social_media,count(*) as 'no_of_social_acquirers',
acquiree_social_media,count(acquiree_social_media) as 'no_of_social_acquirees'
FROM acquisitions group by acquirer_social_media,acquiree_social_media


/* Analysis 6 */
--"Qualification (Degree) and Institution Analysis of directors of acquirers vs
acquirees"
SELECT acquisitions.acquiree_Director_naname,
Degree.degree_type, Degree.institution_name
FROM acquisitions
INNER JOIN Degree
ON acquisitions.acquiree_Director_Id = Degree.person_uuid;

SELECT acquisitions.acquirer_director_name,
Degree.degree_type, Degree.institution_name
FROM acquisitions
INNER JOIN Degree
ON acquisitions.acquirer_director_Id= Degree.person_uuid;


/* Analysis 7 */
--"Check number of such acquisitions where both directors of acquirer and acquiree are
from the same institute"

select * from acquisitions;
select * from Degree;

--Base Code For Join
SELECT acquisitions.acquiree_Director_naname,acquisitions.acquirer_director_name,
Degree.degree_type, Degree.institution_name
FROM acquisitions
INNER JOIN Degree
ON acquisitions.acquiree_Director_Id = Degree.person_uuid;

--Code for Making Modified Table
--for table 1
SELECT acquisitions.acquiree_Director_naname,
Degree.degree_type, Degree.institution_name
FROM acquisitions
INNER JOIN Degree
ON acquisitions.acquiree_Director_Id = Degree.person_uuid;

--for table 2
SELECT acquisitions.acquirer_director_name,
Degree.degree_type, Degree.institution_name
FROM acquisitions
INNER JOIN Degree
ON acquisitions.acquirer_director_Id= Degree.person_uuid;

```

```

--Creating Two Temporary Table
--Table 1
SELECT acquisitions.acquiree_Director_name,
Degree.degree_type, Degree.institution_name
Into temp_tbl1
FROM acquisitions
INNER JOIN Degree
ON acquisitions.acquiree_Director_Id = Degree.person_uuid;

--Table 2
SELECT acquisitions.acquirer_director_name,
Degree.degree_type, Degree.institution_name
Into temp_tbl2
FROM acquisitions
INNER JOIN Degree
ON acquisitions.acquirer_director_Id= Degree.person_uuid;

select * from temp_tbl1;
select * from temp_tbl2;

--Inner Join Between Two Temporary table between Institute
SELECT temp_tbl1.acquiree_Director_name,
temp_tbl1.degree_type, temp_tbl1.institution_name,
temp_tbl2.acquirer_Director_name,
temp_tbl2.degree_type
FROM temp_tbl1
INNER JOIN temp_tbl2
ON temp_tbl1.institution_name = temp_tbl2.institution_name;

```