LOAN PROJECT SQL QUERIES

DATASETS:

- account.csv
- card.csv
- client.csv
- disp.csv
- district.csv
- loan.csv
- order.csv

EXPLORATORY DATA ANALYSIS (EDA):

Counting the number of records in each table.

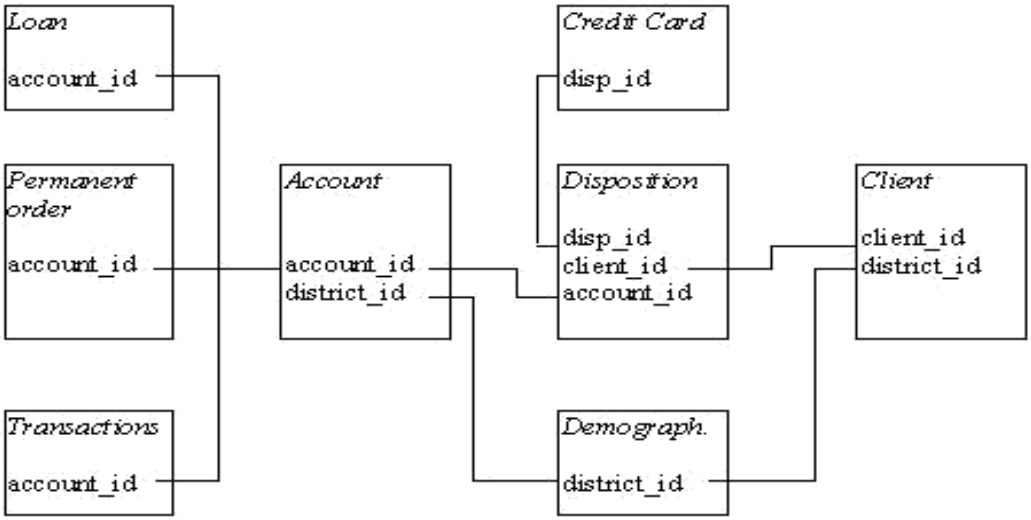
Queries:

```
select Count (*) from [dbo].[account]

select Count (*) from [dbo].[card]

select Count (*) from [dbo].[client]

select Count (*) from [dbo].[disp]

select Count (*) from [dbo].[district]

select Count (*) from [dbo].[loan]

select Count (*) from [dbo].[order]

select Count (*) from [dbo].[transaction_data]
```

SOLUTION APPROACH:

In this project, the raw data is stored in the database and from there all the ETL functionalities has to be performed as follows.

Datasets Entity Mapping:



By the above-mentioned entity mapping, master table is to be created that takes into consideration all the related tables above mentioned. The client id is the common to all from which relationships can be built.

DATA TRANSFORMATION:

Transaction Data and Loan Table:

Joining the transaction Data table and Loan Table using inner join.

```
select  * into  loan_trans  from (select td.*,ln.loan_id,ln.date as loan_date,ln.amount as loan_amount, ln.duration as loan_duration,ln.payments as loan_payments,ln.status as loan_status from loan ln join transaction_data td on ln.account_id = td.account_id)A;
```

Viewing the Merged Table:

```
select * from loan_trans
```

Account and Order Table:

Joining the Account Table and Orders Table using Inner Joins.

```
select * into acc_ord from (select o.*,acc.date as account_date,acc.district_id as account_district_id,acc.frequency from account acc left join [order] o on acc.account_id=o.account_id)B;
```

Viewing the Merged Table:

```
Select * from acc_ord
```

Card and Disposition Table:

Joining the Card Table and Disposition table by Inner Join.

```
select * into  card_disp from (select card .*,disp.account_id,disp.client_id as disposition_client_id, disp.type as disposition_type from card card join disp disp on card .disp_id=disp.disp_id)C;
```

Viewing the Merged Table:

```
Select * from card_disp
```

Card – Disposition and Client Table:

Joining the merged Card and Disposition Table with Client Table based on Client Id using inner join.

```
select  * into  card_disp_client from (select * from card_disp cd join client c on cd.disposition_client_id=c.client_id)D;
```

Viewing the Merged Table:

```
Select * from card_disp_client
```

Card - Disposition - Client and District Table:

Joining the merged card, Disposition and Client Table with District Table based on district id using inner join.

```
select * into card_disp_client_dist from (select * from card_disp_clent cdc join district dist on cdc.district_id=dist.A1)E;
```

Viewing the Merged Table:

Select * from card_disp_client_dist

Account - Order and Card - Disposition - Client - District Table:

Joining the merged Account and Order Table with merged Card, Disposition, Client and District Table based on account id by left join.

```
select * into acc_ord_card_disp_client_dist  from (select cdcd.*,ao.order_id,ao.bank_to,ao.account_to,ao.amount,ao.k_symbol,ao.account_date,ao.account_district_id, ao.frequency from acc_ord ao left join card_disp_client_dist cdcd on ao.account_id=cdcd.account_id)F;
```

Viewing the Merged Table:

Select * from acc_ord_card_disp_client_dist

Master Table:

Joining and Viewing the merged Account, Order Card, Disposition, Client, District with merged Transaction Data Table and Loan Table using inner Joins.

```
select * from acc_ord_card_disp_client_dist aocdcd join loan_trans lt on lt.account_id = aocdcd.account_id
```