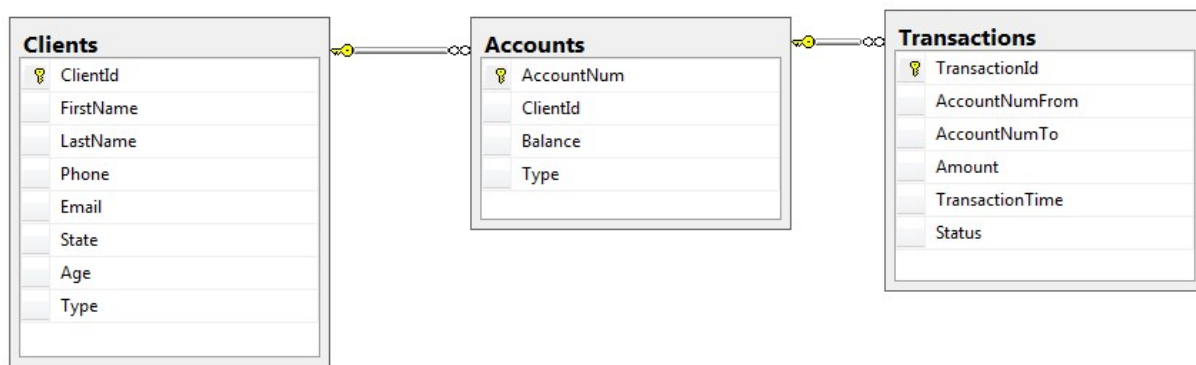# Database Bank.

Entity Relationship Diagram

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, ER diagrams illustrate the logical or physical structure of databases.

Schema: **Bank**

Data Sample.

Table **Bank.Clients**

	ClientId	FirstName	LastName	Phone	Email	State	Age	Type
1	1	John	Smith	703-543-3302	John@gmail.com	VA	33	Private
2	2	Jereny	Smith	723-543-3302	Jeremy@gmail.com	WA	19	Private
3	3	Long	Mark	722-366-5588	MarkLong@Yahoo.com	TN	41	Private
4	4	Bob	James	703-366-9632	bob@microsoft.com	VA	28	Business
5	5	Adam	Marcos	703-566-0000	adam@Marcos.com	CA	38	Business
6	6	Jason	Boley	345-234-9784	json@blabla.com	NY	31	Business
7	7	Tom	Soyer	572-223-5392	stom@hotmail.com	NJ	49	Private

Table **Bank.Accounts**

	AccountNum	ClientId	Balance	Type
1	1	1	10200.00	CHECKING
2	2	1	3550.00	CREDIT
3	3	2	1001.00	CHECKING
4	4	2	150.00	CREDIT
5	5	3	1303.00	CHECKING
6	6	3	25000.00	SAVING
7	7	4	15731.00	CHECKING
8	8	4	31014.00	SAVING
9	9	5	1724.00	CHECKING
10	10	5	3043.00	CREDIT
11	11	5	79320.00	SAVING

Table **Bank.Transactions**

	TransactionId	AccountNumFrom	AccountNumTo	Amount	TransactionTime	Status
1	1	1	2	150.00	2015-01-10 00:00:00.000	Pending
2	2	1	4	1000.00	2016-02-11 00:00:00.000	Commited
3	3	1	8	100.00	2016-04-01 00:00:00.000	Rejected
4	4	1	9	343.55	2017-01-18 00:00:00.000	Pending
5	5	2	9	36.70	2016-12-10 00:00:00.000	Commited
6	6	3	9	100.00	2016-12-12 00:00:00.000	Commited
7	7	5	9	1500.00	2015-01-10 00:00:00.000	Commited
8	8	5	10	1500.00	2016-06-13 00:00:00.000	Rejected
9	9	9	10	2300.00	2016-11-30 00:00:00.000	Commited
10	10	9	11	15000.00	2017-01-01 00:00:00.000	Commited

SQL Table definition (DDL).

Data Definition Language (DDL) is a standard for commands that define the structures in a database. DDL statements create, modify, and remove database objects such as tables, indexes, and users. Common DDL statements are CREATE, ALTER, and DROP

```sql
CREATE TABLE Bank.Clients
(
    ClientId int not null IDENTITY(1,1) PRIMARY KEY,
    FirstName varchar(100),
    LastName varchar(100),
    Phone varchar(20),
    Email varchar(20),
    State CHAR(2),
    Age INT,
    Type varchar(10)
);


CREATE TABLE Bank.Accounts
(
    AccountNum int not null PRIMARY KEY,
    ClientId INT NOT NULL,
    Balance numeric(10,2),
    Type CHAR(10)
);
ALTER TABLE Accounts ADD FOREIGN KEY (ClientId) REFERENCES Bank.Clients(ClientId);


CREATE TABLE Bank.Transactions
(
   TransactionId int not null IDENTITY(1,1) PRIMARY KEY,
   AccountNumFrom int,
   AccountNumTo int,
   Amount numeric(10,2),
   TransactionTime DateTime not null default GETDATE(),
   Status VARCHAR(10)
 );

ALTER TABLE Transactions ADD FOREIGN KEY (AccountNumFrom) REFERENCES Bank.Accounts(AccountNum);

ALTER TABLE Transactions ADD FOREIGN KEY (AccountNumTo) REFERENCES Bank.Accounts(AccountNum);
```

SQL Query tasks (DML)

Clients
1. Get all a data from Table Clients
2. Find all Clients with LastName ='Smith'
3. Get the FirstName,LastName,Phone,State for all clients from Virginia.
4. Get the FirstName,LastName,Phone,Email of all clients who have an Gmail account.
5. Get all data of client whose LastName starts from letter 'M'
6. Get all data of client whose LastName starts from letter 'M' and has the last letter 'S'
7. Get Names and phones of all Business clients;
8. Get information about clients younger than 20.
9. Get information about clients whose age between 20 and 40.
10. Find the age of the youngest client and average age of our clients.
11. Find average age and number of our Private clients.
12. Find the age of oldest client who lives not in New Jersey.

Accounts
1. Show balances of all 'CREDIT' accounts.
2. Find Account with the biggest balance.
3. Calculate the average balance by each Client.
4. Calculate the average balance and number of Accounts by each Client.

Transactions
1. Find all 'Pending' transactions.
2. Find all 'commited' transactions made since '1-jan-2016'
3. Find all transactions made in 2016 year.
4. Find average amount of transactions
5. Find average amount and number of transactions paid by each Account.