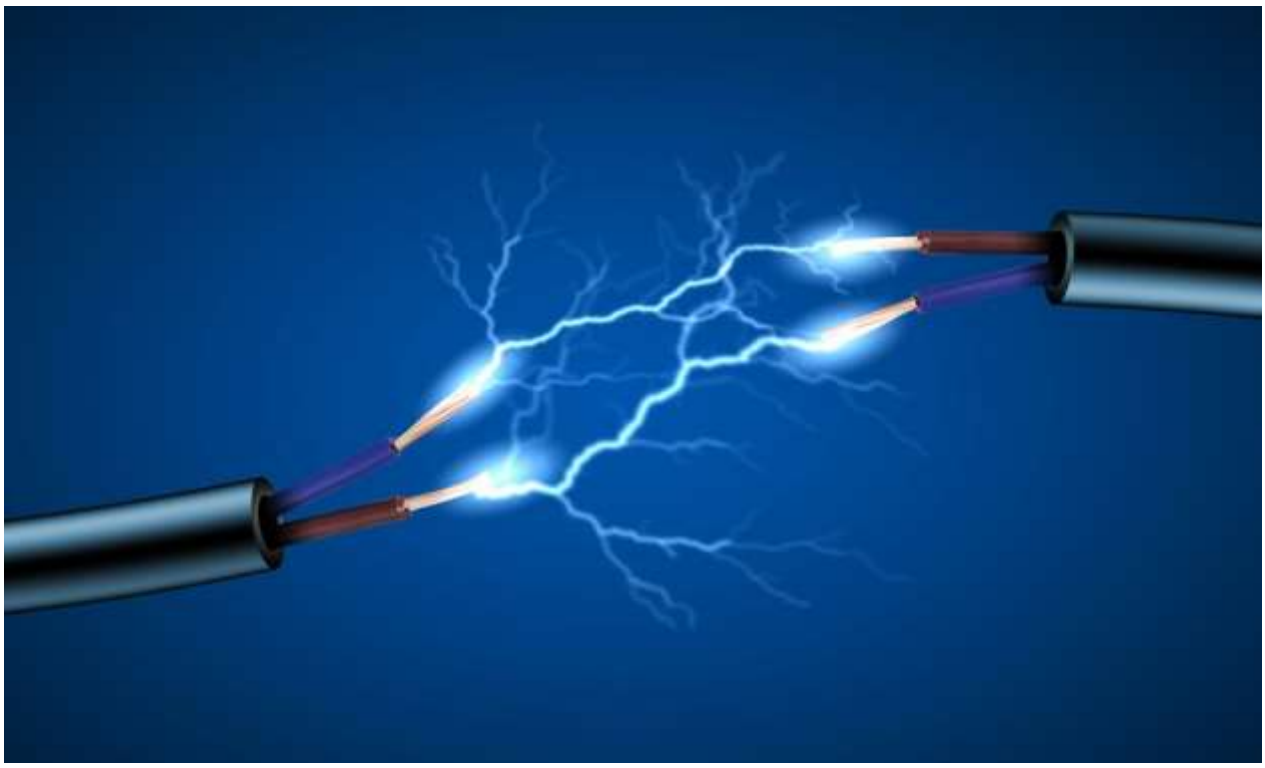
DBMS PROJECT AND TEAM DETAILS



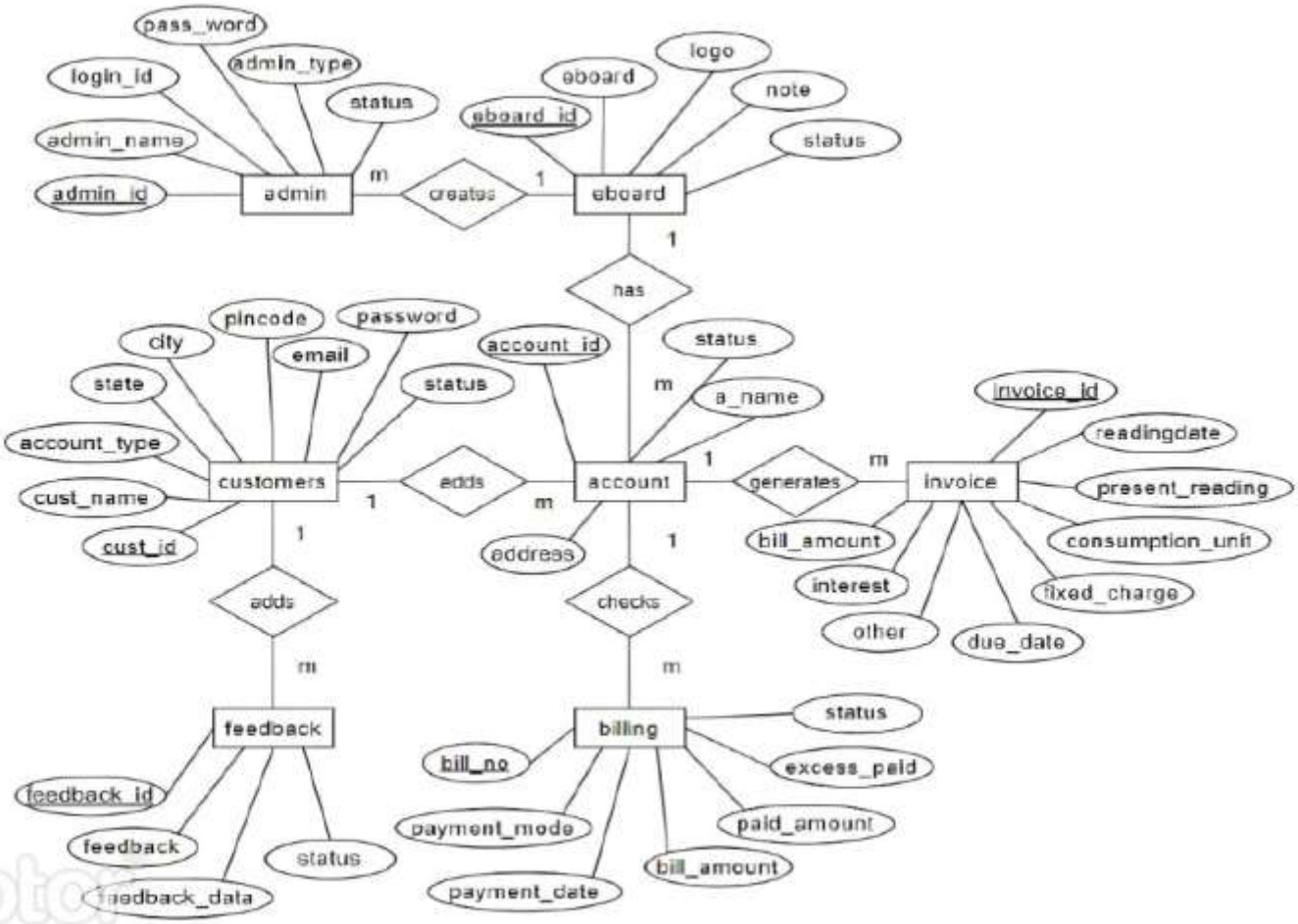
PROJECT NAME - Electricity Bill Management System

Explanation About The Database Project:

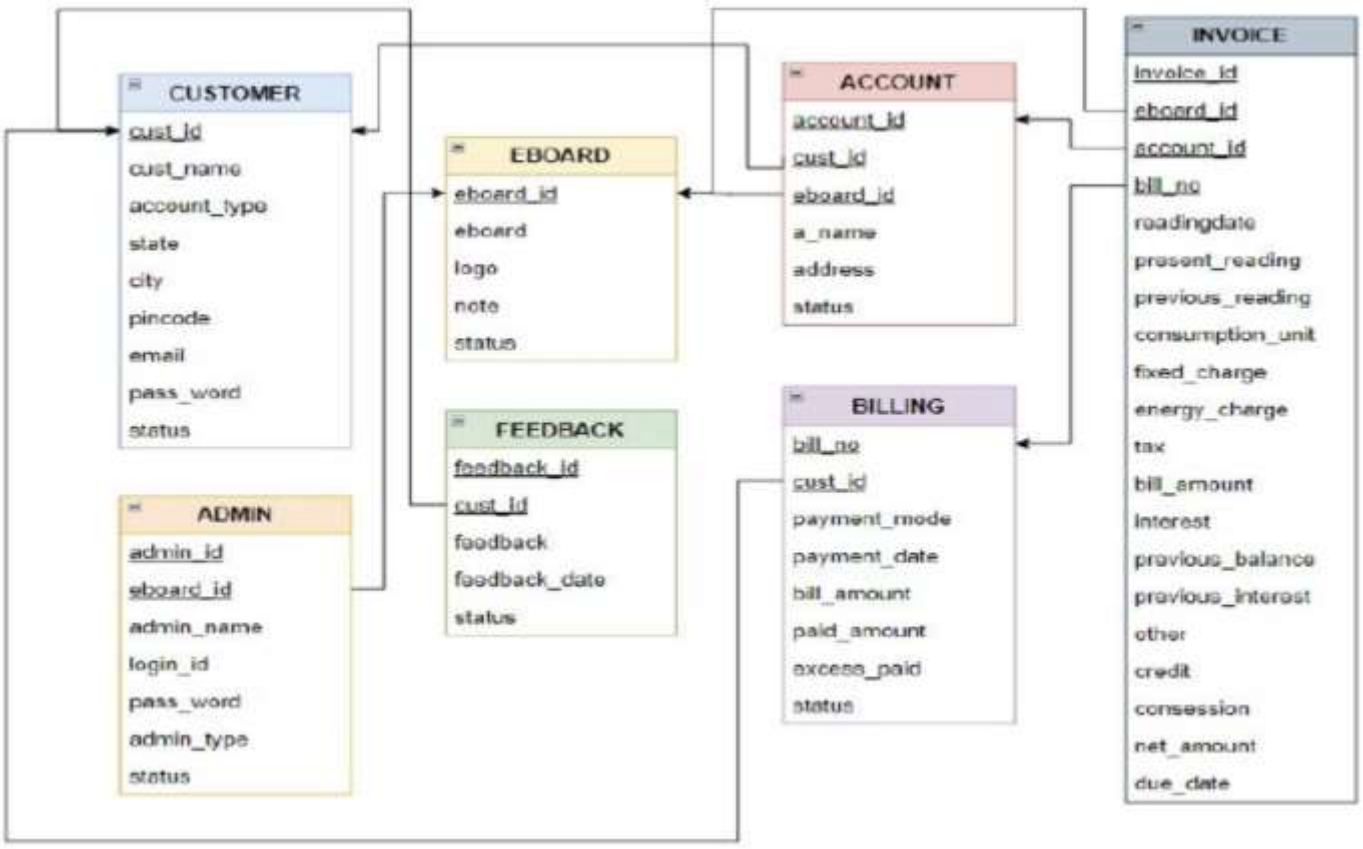
The main aim of developing the Electricity Bill Management System is to keep records of the customers' bills. The admin can manage all the customers' accounts, and the registered users like employees and customers can only manage their accounts. This system helps to maintain the bills and the payments.

In this project, different modules such as Login, User, Admin, Queries, Department, and Meters are designed considering the basic needs encountered at the time of generation, distribution, payment, payment, and payment of electricity bills.

Entity–relationship model



Relational model



NORMAL SQL TABLE:

```
Create TABLE Customer
(
  cust_id int primary key not null ,
  cust_name varchar(255),
  account_type varchar(20) ,
  state varchar(20) ,
  city varchar(20) ,
  pincode int ,
  email varchar(50) ,
  pass_word varchar(10),
  status varchar(10)
  );
```

```
  Create table EBOARD
  (
    eboard_id int primary key not null ,
    eboard varchar(20),
    logo varchar(20) ,
    note varchar(200) ,
    status varchar(10)

    );
```

```
  Create table ADMIN (
    admin_id int primary key not null ,
    eboard_id int ,
    admin_name varchar(20) ,
```

```
  login_id int ,
  pass_word varchar(20) ,
  admin_type varchar(20)  ,
  status varchar(10)  ,
  FOREIGN KEY (eboard_id) REFERENCES EBOARD(eboard_id)
  );



Create table FEEDBACK
(
  feedback_id int primary key not null ,
  cust_id int ,
  feedback varchar(10) ,
  feedback_date date,
  status varchar(10)  ,
  FOREIGN KEY (cust_id) REFERENCES Customer(cust_id)
  ) ;


  Create table BILLING
  (
    bill_no int primary key not null ,
    cust_id int ,
    payment_mode varchar(10) ,
    payment_date date ,
    bill_amount int ,
    paid_amount int ,
    excess_paid int  ,
    status varchar(10),
    FOREIGN KEY (cust_id) REFERENCES Customer(cust_id)
    ) ;

  Create table ACCOUNT
  (
    account_id int primary key not null ,
    cust_id int,
    eboard_id int ,
    a_name varchar(30),
    address varchar(255) ,
    status varchar(10) ,
    FOREIGN KEY (cust_id) REFERENCES Customer(cust_id)
```

```
,
      FOREIGN KEY (eboard_id) REFERENCES
EBOARD(eboard_id)
      );



      CREATE TABLE INVOICE
      (
        invoice_id int not null primary key ,
        eboard_id int ,

        account_id int ,
        bill_no int ,

        readingdate date ,
        present_reading varchar(20) ,
        previous_reading varchar(20) ,
        consumption_unit int ,
        fixed_charge float  ,
        energy_charge float,
        tax float ,
        bill_amount float ,
        interest float ,
        previous_balance float ,
        previous_interest float ,
        other float  ,
        credit float   ,
        consession float  ,
        net_amount float ,
        due_date DATE ,
         FOREIGN KEY (eboard_id) REFERENCES
EBOARD(eboard_id) ,
         FOREIGN KEY (account_id) REFERENCES
ACCOUNT(account_id) ,
         FOREIGN KEY (bill_no) REFERENCES
BILLING(bill_no)

        ) ;
```

NORMALIZE THE TABLE TO 3NF

To normalize the tables to 3NF, we need to follow the normalization rules to remove any functional dependencies and make sure each table contains only one type of data. Here are the normalized tables:

***The functional dependency is a relationship that exists between two attributes. Ittypically exists between the primary key and non-key attribute within a table.**

Customer Table:

cust_id (PK)
cust_name
state
city
pincode
email
pass_word
status
In this table, account_type was removed because it can be inferred from the Account table.

EBoard Table:

eboard_id (PK)
eboard
logo
note
status
Admin Table:

admin_id (PK)
eboard_id (FK)
admin_name
login_id
pass_word
admin_type
status

In this table, the Admin type was kept as it is related to the admin and not to the eboard.

Feedback Table:

feedback_id (PK)
cust_id (FK)
feedback
feedback_date
status
Billing Table:

bill_no (PK)
cust_id (FK)
payment_mode
payment_date
status
In this table, the bill amount and excess paid were removed since they can be inferred from the Invoice table.

Account Table:

account_id (PK)
cust_id (FK)
eboard_id (FK)
a_name
address
status
Invoice Table:

invoice_id (PK)
eboard_id (FK)
account_id (FK)
bill_no (FK)
readingdate
present_reading
previous_reading
consumption_unit
fixed_charge
energy_charge
tax

interest
previous_balance
previous_interest
other
credit
consession
net_amount
due_date
In this table, we removed bill amount and excess paid since they can be inferred from the Billing table.

These tables are now in third normal form, where each table contains only one type of data and there are no functional dependencies between non-key attributes.

Normalization TABLE:

```
CREATE TABLE Customer (
  cust_id INT NOT NULL PRIMARY KEY,
  cust_name VARCHAR(255),
  account_type VARCHAR(20),
  state VARCHAR(20),
  city VARCHAR(20),
  pincode INT,
  email VARCHAR(50),
  pass_word VARCHAR(10),
  status VARCHAR(10)
);

CREATE TABLE EBoard (
  eboard_id INT NOT NULL PRIMARY KEY,
  eboard VARCHAR(20),
  logo VARCHAR(20),
  note VARCHAR(200),
  status VARCHAR(10)
);
```

```
CREATE TABLE Admin (
  admin_id INT NOT NULL PRIMARY KEY,
  eboard_id INT,
  admin_name VARCHAR(20),
  login_id INT,
  pass_word VARCHAR(20),
  admin_type VARCHAR(20),
  status VARCHAR(10),
  FOREIGN KEY (eboard_id) REFERENCES EBoard(eboard_id)
);
```

```
CREATE TABLE Feedback (
  feedback_id INT NOT NULL PRIMARY KEY,
  cust_id INT,
  feedback VARCHAR(10),
  feedback_date DATE,
  status VARCHAR(10),
  FOREIGN KEY (cust_id) REFERENCES Customer(cust_id)
);
```

```
CREATE TABLE Billing (
  bill_no INT NOT NULL PRIMARY KEY,
  cust_id INT,
  payment_mode VARCHAR(10),
  payment_date DATE,
  bill_amount INT,
  paid_amount INT,
  excess_paid INT,
```

```
  status VARCHAR(10),
  FOREIGN KEY (cust_id) REFERENCES Customer(cust_id)
);



CREATE TABLE Account (
  account_id INT NOT NULL PRIMARY KEY,
  cust_id INT,
  eboard_id INT,
  a_name VARCHAR(30),
  address VARCHAR(255),
  status VARCHAR(10),
  FOREIGN KEY (cust_id) REFERENCES Customer(cust_id),
  FOREIGN KEY (eboard_id) REFERENCES EBoard(eboard_id)
);









CREATE TABLE Invoice (
  invoice_id INT NOT NULL PRIMARY KEY,
  eboard_id INT,
  account_id INT,
  bill_no INT,
  readingdate DATE,
  present_reading VARCHAR(20),
  previous_reading VARCHAR(20),
  consumption_unit INT,
```

```
  fixed_charge FLOAT,
  energy_charge FLOAT,
  tax FLOAT,
  bill_amount FLOAT,
  interest FLOAT,
  previous_balance FLOAT,
  previous_interest FLOAT,
  other FLOAT,
  credit FLOAT,
  consession FLOAT,
  net_amount FLOAT,
  due_date DATE,
  FOREIGN KEY (eboard_id) REFERENCES EBoard(eboard_id),
  FOREIGN KEY (account_id) REFERENCES
Account(account_id),
  FOREIGN KEY (bill_no) REFERENCES Billing(bill_no)
);
```

INSERTING INTO CUSTOMER

```
INSERT INTO Customer (cust_id, cust_name, account_type, state,
city, pincode, email, pass_word, status) VALUES
(1, 'John Smith', 'Residential', 'California', 'Los Angeles', 90001,
'john.smith@email.com', '123456', 'Active'),
(2, 'Sarah Johnson', 'Commercial', 'New York', 'New York City', 10001,
'sarah.johnson@email.com', 'abcdef', 'Active'),
(3, 'Robert Lee', 'Residential', 'Texas', 'Houston', 77001,
'robert.lee@email.com', 'qwerty', 'Inactive'),
(4, 'Megan Chen', 'Residential', 'California', 'San Francisco', 94102,
'megan.chen@email.com', 'p@ssword', 'Active'),
(5, 'David Kim', 'Commercial', 'Illinois', 'Chicago', 60601,
'david.kim@email.com', '1q2w3e', 'Active');
```

Customer

cust_id	cust_name	account_type	state	city	pincode	email	pass_word	status
1	John Smith	Residential	California	Los Angeles	90001	john.smith@email.com	123456	Active
2	Sarah Johnson	Commercial	New York	New York City	10001	sarah.johnson@email.com	abcdef	Active
3	Robert Lee	Residential	Texas	Houston	77001	robert.lee@email.com	qwerty	Inacti
4	Megan Chen	Residential	California	San Francisco	94102	megan.chen@email.com	p@ssword	Active
5	David Kim	Commercial	Illinois	Chicago	60601	david.kim@email.com	1q2w3e	Active

INSERTING INTO EBoard

```
INSERT INTO EBoard (eboard_id, eboard, logo, note, status)
VALUES
(1, 'ABC Electric', 'abc_logo.jpg', 'Providing electricity since 1980.',
'Active'),
(2, 'XYZ Power', 'xyz_logo.jpg', 'Sustainable energy for a better
future.', 'Active'),
(3, 'PQR Energy', 'pqr_logo.jpg', 'Affordable and reliable energy
solutions.', 'Inactive'),
(4, 'MNO Utilities', 'mno_logo.jpg', 'Connecting people to power.',
'Active'),
(5, 'LMN Electric', 'lmn_logo.jpg', 'Innovative solutions for your energy
needs.', 'Active');
```

EBoard

eboard_id	eboard	logo	note	status
1	ABC Electric	abc_logo.jpg	Providing electricity since 1980.	Active
2	XYZ Power	xyz_logo.jpg	Sustainable energy for a better future.	Active
3	PQR Energy	pqr_logo.jpg	Affordable and reliable energy solutions.	Inactive
4	MNO Utilities	mno_logo.jpg	Connecting people to power.	Active
5	LMN Electric	lmn_logo.jpg	Innovative solutions for your energy needs.	Active

INSERTING INTO Admin

```
INSERT INTO Admin (admin_id, eboard_id, admin_name, login_id,
pass_word, admin_type, status) VALUES
(1, 1, 'Admin 1', 123, 'a1password', 'Superuser', 'Active'),
```

(2, 1, 'Admin 2', 456, 'a2password', 'User', 'Active'),
(3, 2, 'Admin 3', 789, 'a3password', 'User', 'Inactive'),
(4, 3, 'Admin 4', 111, 'a4password', 'Superuser', 'Active'),
(5, 4, 'Admin 5', 222, 'a5password', 'User', 'Active');

Admin

admin_id	eboard_id	admin_name	login_id	pass_word	admin_type	status
1	1	Admin 1	123	a1password	Superuser	Active
2	1	Admin 2	456	a2password	User	Active
3	2	Admin 3	789	a3password	User	Inactive
4	3	Admin 4	111	a4password	Superuser	Active
5	4	Admin 5	222	a5password	User	Active

INSERTING INTO Feedback

INSERT INTO Feedback (feedback_id, cust_id, feedback, feedback_date, status) VALUES
(1, 1, 'Excellent service!', '2022-01-15', 'Active'),
(2, 2, 'Power outage on 5/10/2023.', '2023-05-12', 'Active'),
(3, 3, 'Billing issue resolved.', '2022-08-23', 'Inactive'),
(4, 4, 'High bill amount, need explanation.', '2023-04-01', 'Active'),
(5, 5, 'No electricity in my area.', '2023-05-08', 'Active');

Feedback

feedback_id	cust_id	feedback	feedback_date	status
1	1	Excellent service!	2022-01-15	Active
2	2	Power outage on 5/10/2023.	2023-05-12	Active
3	3	Billing issue resolved.	2022-08-23	Inactive
4	4	High bill amount, need explanation.	2023-04-01	Active
5	5	No electricity in my area.	2023-05-08	Active

INSERTING INTO Billing

INSERT INTO Billing (bill_no, cust_id, payment_mode, payment_date, bill_amount, paid_amount, excess_paid, status) VALUES
(1, 1, 'Credit Card', '2023-05-15', 100, 50, 50, 'Active'),
(2, 2, 'Debit Card', '2023-06-15', 150, 100, 50, 'Active'),
(3, 3, 'PayPal', '2023-07-15', 200, 150, 50, 'Active');

Billing

bill_no	cust_id	payment_mode	payment_date	bill_amount	paid_amount	excess_paid	status
1	1	Credit Card	2023-05-15	100	50	50	Active
2	2	Debit Card	2023-06-15	150	100	50	Active
3	3	PayPal	2023-07-15	200	150	50	Active

INSERTING INTO Account

INSERT INTO Account (account_id, cust_id, eboard_id, a_name, address, status) VALUES
(1, 1, 1, 'John Smith', '123 Main Street, San Francisco, CA 94107', 'Active'),
(2, 2, 2, 'Jane Doe', '456 Elm Street, Los Angeles, CA 90001', 'Active'),
(3, 3, 3, 'Peter Jones', '789 Pine Street, New York, NY 10001', 'Active');

Account

account_id	cust_id	eboard_id	a_name	address	status
1	1	1	John Smith	123 Main Street, San Francisco, CA 94107	Active
2	2	2	Jane Doe	456 Elm Street, Los Angeles, CA 90001	Active
3	3	3	Peter Jones	789 Pine Street, New York, NY 10001	Active

INSERTING INTO Invoice

INSERT INTO Invoice (invoice_id, eboard_id, account_id, bill_no, readingdate, present_reading, previous_reading, consumption_unit, fixed_charge, energy_charge, tax, bill_amount, interest, previous_balance, previous_interest, other, credit, consession, net_amount, due_date) VALUES
(1, 1, 1, 1, '2023-05-15', 100, 50, 50, 10, 20, 5, 0, 45, 0, 0, 0, 0, 0, 0, '2023-06-15'),
(2, 2, 2, 2, '2023-06-15', 150, 100, 50, 15, 30, 7.5, 0, 62.5, 0, 0, 0, 0, 0, 0, '2023-07-15'),
(3, 3, 3, 3, '2023-07-15', 200, 150, 50, 20, 40, 10, 0, 70, 0, 0, 0, 0, 0, 0, '2023-08-15');

Invoice

invoice_id	eboard_id	account_id	bill_no	readingdate	present_reading	previous_reading	consumption_unit
1	1	1	1	2023-05-15	100	50	50
2	2	2	2	2023-06-15	150	100	50
3	3	3	3	2023-07-15	200	150	50

Invoice

fixed_charge	energy_charge	tax	bill_amount	interest	previous_balance	previous_interest	other	credit
10	20	5	0	45	0	0	0	0
15	30	7.5	0	62.5	0	0	0	0
20	40	10	0	70	0	0	0	0

consession	net_amount	due_date
0	0	2023-06-15
0	0	2023-07-15
0	0	2023-08-15