

**INTRODUCTION TO DATABASE [K]**

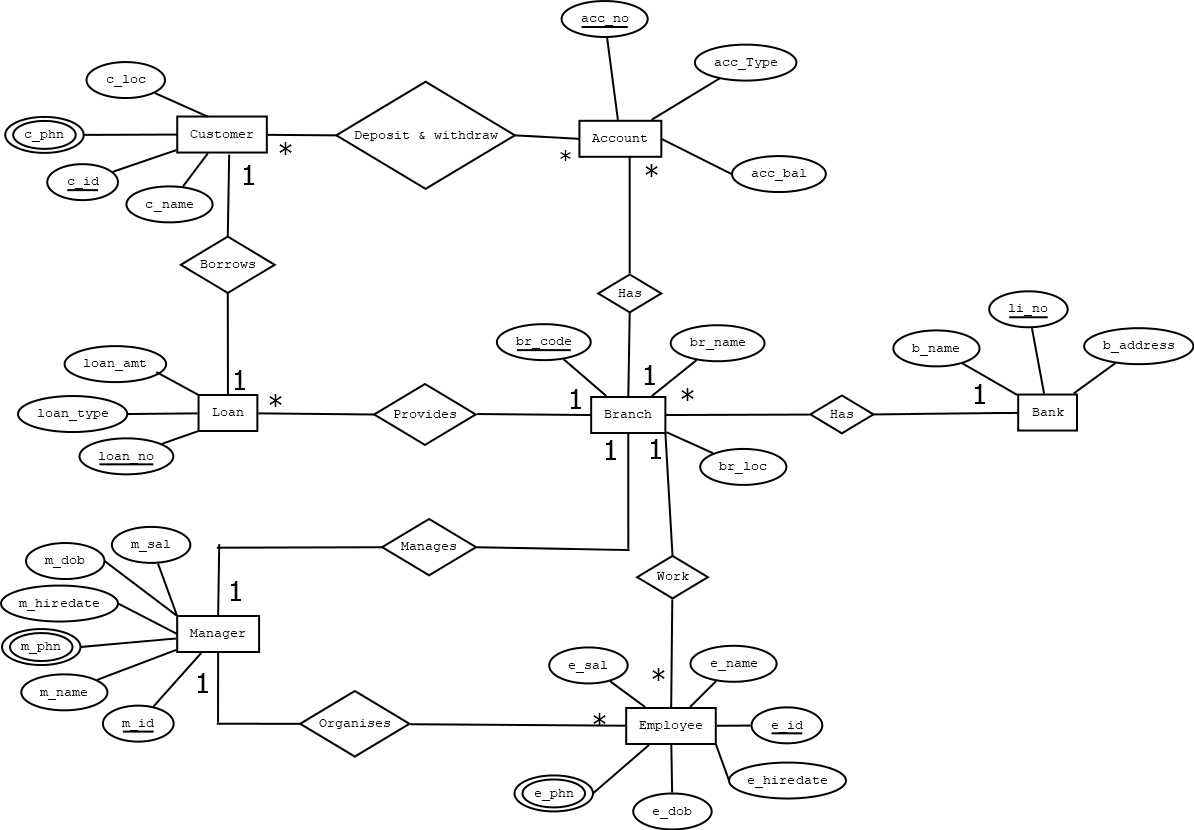
**Final-Term Project**

**Project Name: Bank Management System**

**Submitted by: Shadril Hassan Shifat [20-42451-1]**

**Submitted to: Rifat Tasnim Anannya [1801-1845-2]**

**ER Diagram**



**Scenario**

* Bank are identified by License No and Bank Name, Address are recorded.
* Employees are identified by Employee ID and Employee Name, Date of Birth, Employee Name, Salary, Joining Date, Phone No are recorded. Each Branch has many Employees.
* Customers are identified by Customer ID and Customer Name, Address, Phone No are recorded.
* Each customer has one or more Account / Accounts which is/are identified by Account No and Account Type, Balance are recorded. Many Customers can also deposit and withdraw from many Accounts.
* Accounts must be created on a specific Branch of the Bank. So, there is a many to one relationship between Account and Branch. So, Branch are identified by Branch Code and Branch Name, Location are recorded.
* Each Branch has a Manager (one to one) which is identified by Manager ID and Manager Name, Salary, Phone No , Date of Birth, Hire Date are recorded. Manager organizes many Employees. The cardinality of Manager and Employee is one to many.
* Each Branch provides many Loan (One to many) which is identified by Loan No and Loan Type, Amount are recorded.
* A Customer can borrow Loan (one to one) from a specific Branch. So, there is one to one relationship.

**Normalization:**

**Has**- (li\_no, b\_name, b\_address, br\_code, br\_name, br\_loc)

**1NF**- No multivalued attribute

**2NF**- li\_no, b\_name, b\_address

br\_code, br\_name, br\_loc

**3NF**- No transitive dependency

li\_no, b\_name, b\_address

br\_code, br\_name, br\_loc

**Table for Has-**

1. li\_no, b\_name, b\_address
2. br\_code, br\_name, br\_loc, li\_no

**Create**- ( acc\_no, acc\_type, acc\_bal, br\_code, br\_name, br\_loc)

**1NF**- No multivalued attribute

**2NF**- acc\_no, acc\_type, acc\_bal

br\_code, br\_name, br\_loc

**3NF**- No transitive dependency

acc\_no, acc\_type, acc\_bal

br\_code, br\_name, br\_loc

**Table for Create-**

1. acc\_no, acc\_type, acc\_bal, br\_code
2. br\_code, br\_name, br\_loc

**Deposit & Withdraw**- ( acc\_no, acc\_type, acc\_bal, c\_id, c\_name, c\_phn, c\_loc )

**1NF-** c\_phn is a multivalued attribute

**2NF-** acc\_no, acc\_type, acc\_bal

c\_id, c\_name, c\_phn, c\_loc

**3NF-** No transitive dependency

acc\_no, acc\_type, acc\_bal

c\_id, c\_name, c\_phn, c\_loc

**Table for Deposit & Withdraw-**

1. acc\_no, acc\_type, acc\_bal
2. c\_id, c\_name, c\_loc
3. n\_id, acc\_no, c\_id
4. c\_id, c\_phn – Composite primary key

**Borrow-** ( c\_id, c\_name, c\_phn, c\_loc, loan\_no, loan\_type, loan\_amt)

**1NF-** c\_phn is a multivalued attribute

**2NF-** c\_id, c\_name, c\_phn, c\_loc

loan\_no, loan\_type, loan\_amt

**3NF-** No transitive dependency

c\_id, c\_name, c\_phn, c\_loc

loan\_no, loan\_type, loan\_amt

**Table for Borrow-**

1. c\_id, c\_name, c\_loc, loan\_no
2. loan\_no, loan\_type, loan\_amt
3. c\_id, c\_phn – Composite primary key

**Provides-** ( br\_code, br\_name, br\_loc, loan\_no, loan\_type, loan\_amt)

**1NF-** No multivalued attribute

**2NF-** br\_code, br\_name, br\_loc

loan\_no, loan\_type, loan\_amt

**3NF-** No transitive dependency

br\_code, br\_name, br\_loc

loan\_no, loan\_type, loan\_amt

**Table for Provides-**

1. loan\_no, loan\_type, loan\_amt, br\_code
2. br\_code, br\_name, br\_loc

**Manages-** ( m\_id, m\_name, m\_sal, m\_dob, m\_hiredate, m\_phn, br\_code, br\_name, br\_loc)

**1NF-** m\_phn is a multivalued attribute

**2NF-** m\_id, m\_name, m\_sal, m\_dob, m\_hiredate, m\_phn

br\_code, br\_name, br\_loc

**3NF-** No transitive dependency

m\_id, m\_name, m\_sal, m\_dob, m\_hiredate, m\_phn

br\_code, br\_name, br\_loc

**Table for Manages-**

1. m\_id, m\_name, m\_sal, m\_dob, m\_hiredate, br\_code
2. br\_code, br\_name, br\_loc
3. m\_id, m\_phn – Composite primary key

**Organises-** (m\_id, m\_name, m\_sal, m\_dob, m\_hiredate, m\_phn, e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, e\_phn)

**1NF-** m\_phn & e\_phn are multivalued attributes

**2NF-** m\_id, m\_name, m\_sal, m\_dob, m\_hiredate, m\_phn

e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, e\_phn

**3NF-** No transitive dependency

m\_id, m\_name, m\_sal, m\_dob, m\_hiredate, m\_phn

e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, e\_phn

**Table for Organises-**

1. m\_id, m\_name, m\_sal, m\_dob, m\_hiredate
2. e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, m\_id
3. m\_id, m\_phn – Composite primary key
4. e\_id, e\_phn – Composite primary key

**Work-** ( e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, e\_phn, br\_code, br\_name, br\_loc )

**1NF-** e\_phn is a multivalued attribute

**2NF-** br\_code, br\_name, br\_loc

e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, e\_phn

**3NF-** No transitive dependency

br\_code, br\_name, br\_loc

e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, e\_phn

**Table for Work-**

1. br\_code, br\_name, br\_loc
2. e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, br\_code
3. e\_id, e\_phn – Composite primary key

**Final table list-**

1. li\_no, b\_name, b\_address - bank
2. br\_code, br\_name, br\_loc, li\_no - branch
3. acc\_no, acc\_type, acc\_bal, br\_code -account
4. n\_id, acc\_no, c\_id -customerinfo
5. c\_id, c\_phn -customer\_phn
6. c\_id, c\_name, c\_loc, loan\_no - customer
7. loan\_no, loan\_type, loan\_amt, br\_code -loan
8. m\_id, m\_name, m\_sal, m\_dob, m\_hiredate, br\_code - manager
9. m\_id, m\_phn – manager\_phn

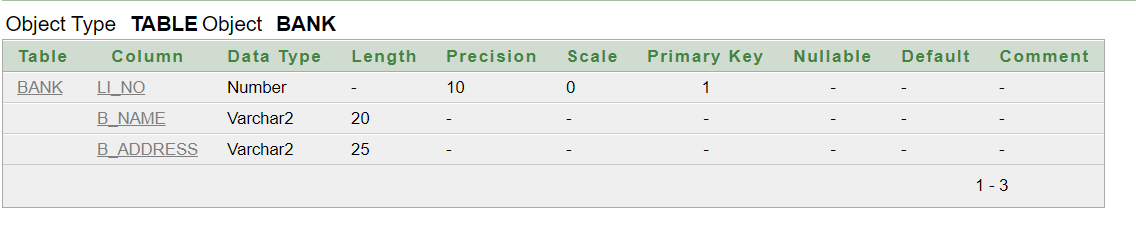
10. e\_id, e\_name, e\_sal, e\_dob, e\_hiredate, m\_id, br\_code -employee

11. e\_id, e\_phn - employee\_phn

# Table Screenshots-

1. **bank-**

Description-



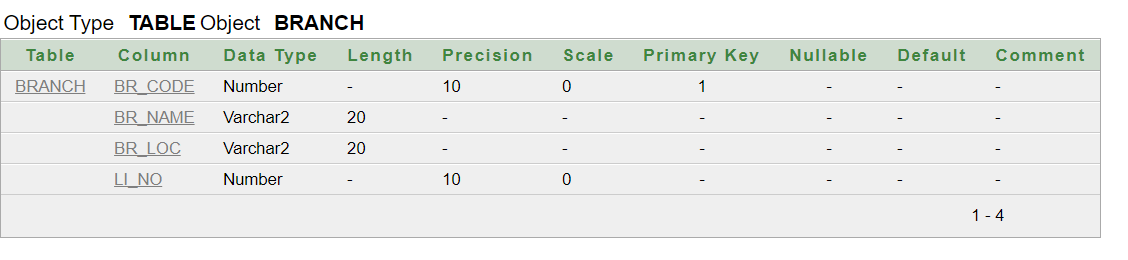
\*\*\* not null constraint added in bank\_name, b\_address

Select \* from bank -



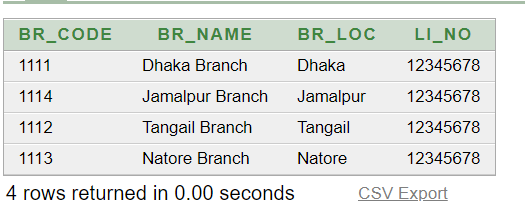
1. **branch-**

Description-



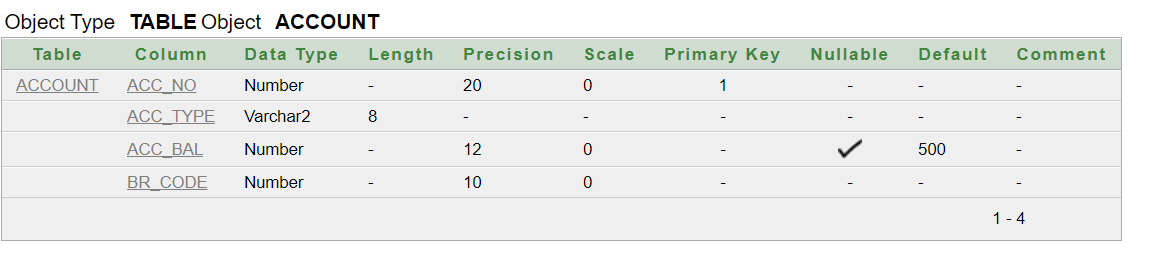
\*\*\* not null constraint added in br\_name, br\_loc, li\_no

Select \* from branch –



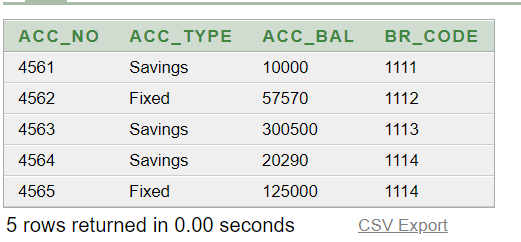
1. **account-**

Description –

****

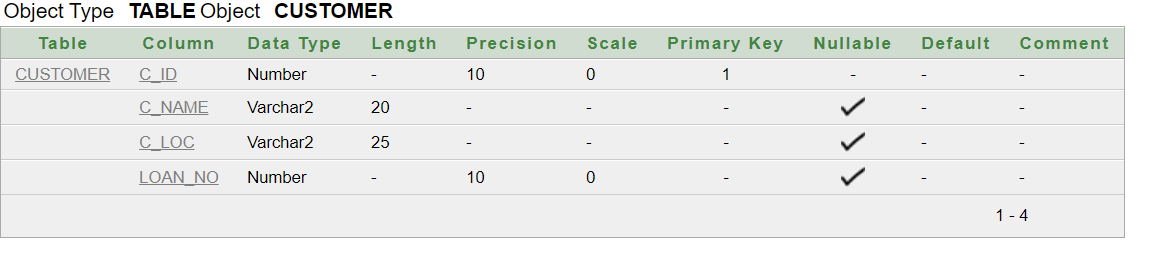
\*\*\* default constraint added in acc\_bal and not null constraint added in acc\_type, br\_code

Select \* from account –

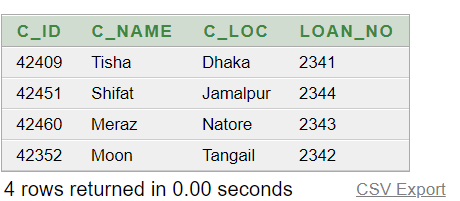


1. **customer-**

Description-

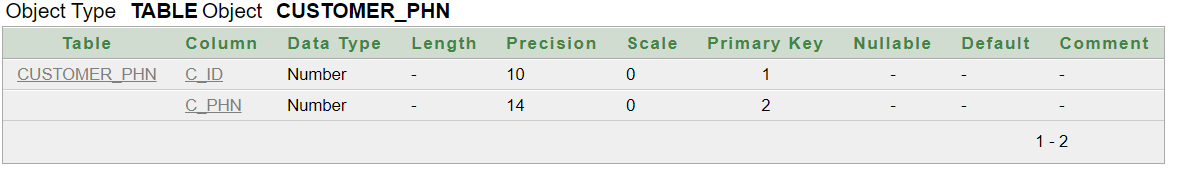


Select \* from customer –

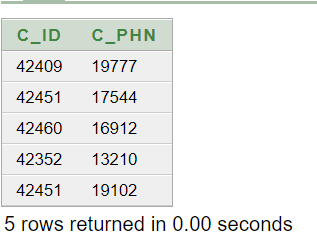


1. **customer\_phn –**

Description-

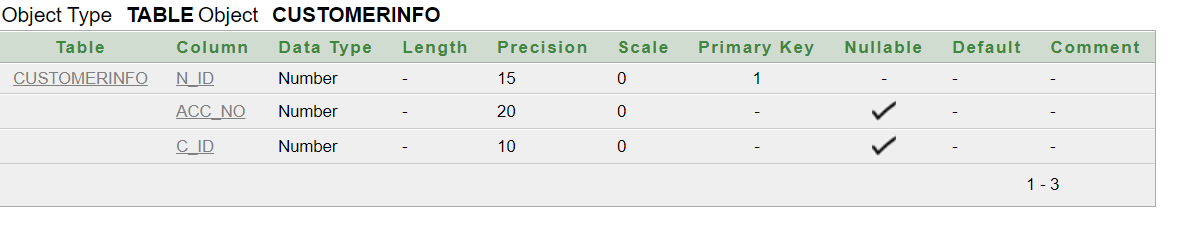


Select \* from customer\_phn –

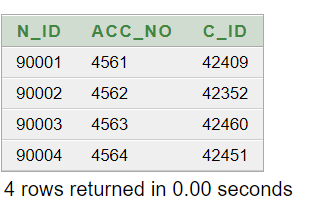


1. **customerinfo-**

Description-

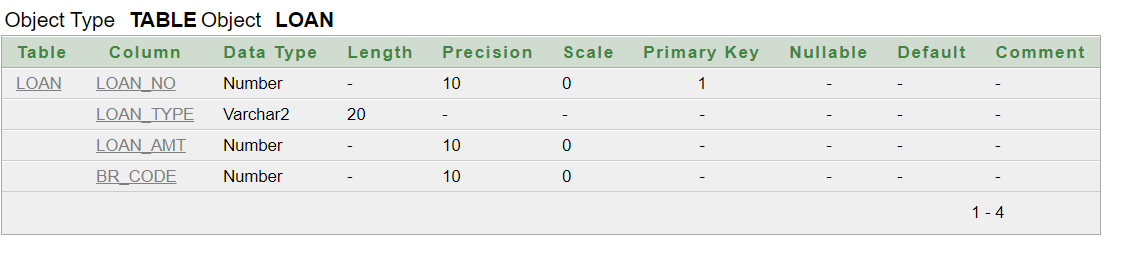


Select \* from customerinfo-



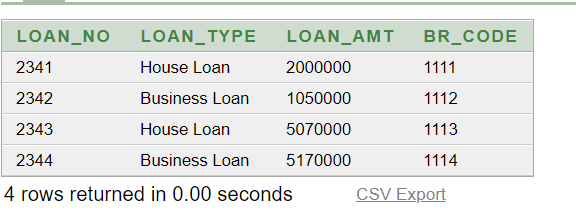
1. **loan-**

Description-



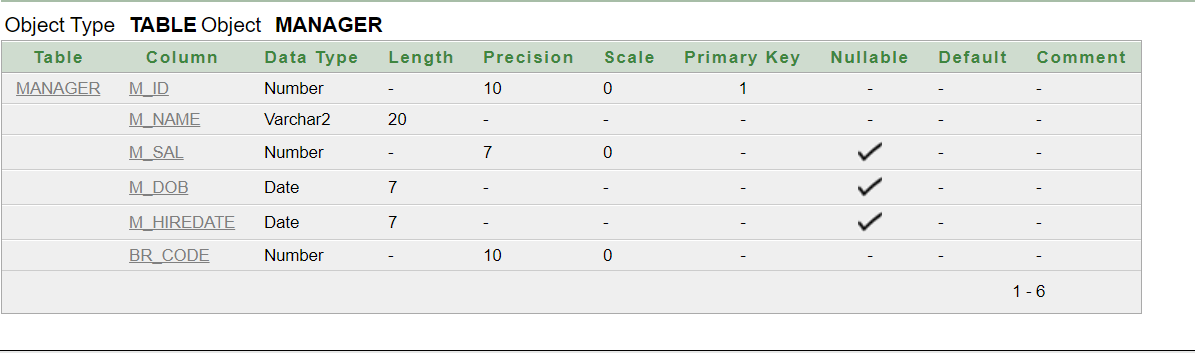
\*\*\* not null constraint added in loan\_type, loan\_amt, br\_code

Select \* from loan –



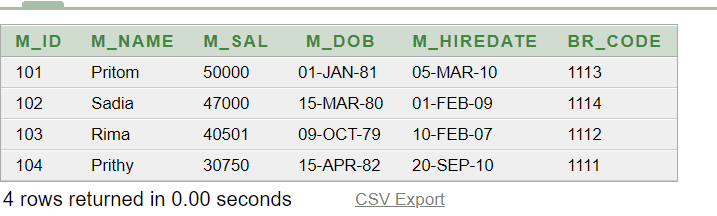
1. **manager –**

Description-



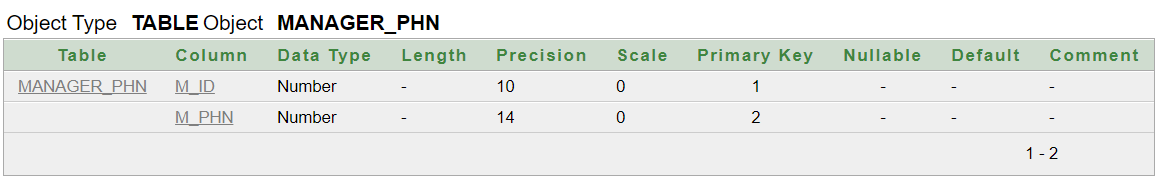
\*\*\* not null constraint added in m\_name, br\_code

Select \* from manager –

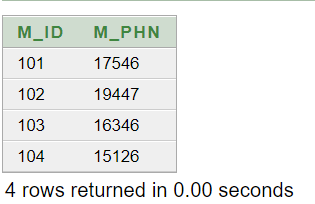


1. **manager\_phn –**

Description-

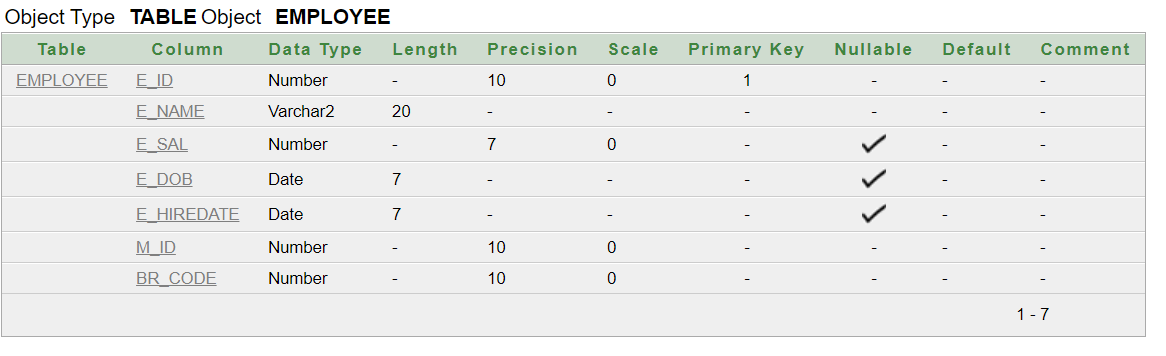


Select \* from manager\_phn –



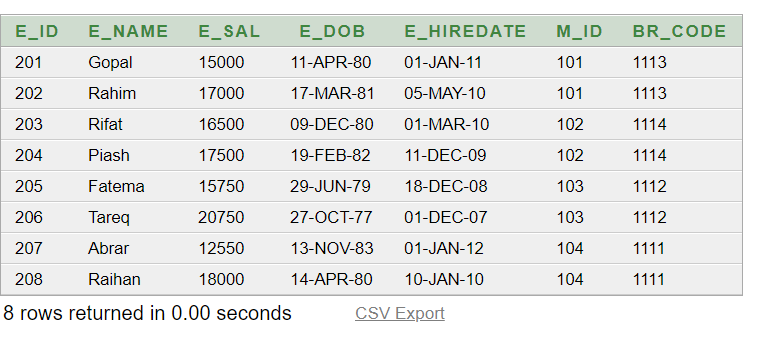
**10. employee-**

Description-

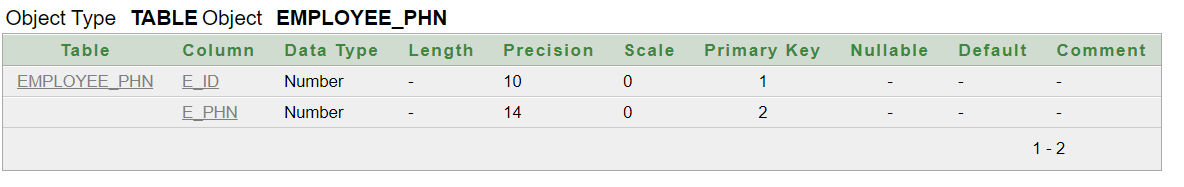


\*\*\* not null constraint added in e\_name, m\_id, br\_code

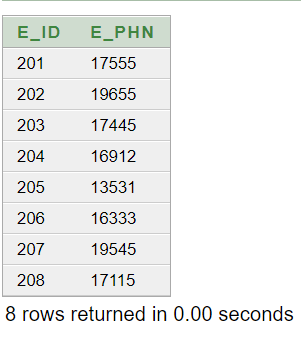
Select \* from employee-



**11. Employee\_phn**

Description- 

Select \* from employee\_phn –

****

**Questions –**

1. Find nid, account no and account balance of these person who have Savings account. (Join)
2. Find the manager id, manager name and manager dob of that employees who earn more than 17000. (Multiple Row Sub query)
3. Display those employee name, employee id and employee working years (Rounded) as Working Years whose manager get salary more than 39000. (Date, Join, Multiple Row Sub query)
4. Display branch name where loan amount is minimum of Business Loan Type . (Group Function, Sub Query)
5. Create sequence named Bankdata. Increment by 1,Max value 999,starts with 101,no min value,no cycle & no cache.(Sequence)
6. Create a view named CustomerLoanDetails which contains all customer id, customer name, customer location, loan no, loan type, loan amount.(view)
7. Display all the details employee whose name start with ‘R’ or the second letter is ‘a’.(like operator)
8. Display the minimum average salary of employee group by branch code. (group by)
9. Display the phone no of those customer whose name length is 5.(length, sub query)

**10 .** Display all the employees name and hiredate who joined after the manager. (Date, subquery)