

2. Create Tables

Create FACULTY Table:

```
CREATE TABLE FACULTY (  
  
    facultyId VARCHAR(5) PRIMARY KEY,  
  
    facultyName VARCHAR(100) NOT NULL,  
  
    NoOfStaff INT CHECK (NoOfStaff > 0)  
  
);
```

```
University=# CREATE TABLE FACULTY (  
University(#      facultyId VARCHAR(5) PRIMARY KEY,  
University(#      facultyName VARCHAR(100) NOT NULL,  
University(#      NoOfStaff INT CHECK (NoOfStaff > 0)  
University(# );  
CREATE TABLE
```

```
mysql> desc faculty;
```

Field	Type	Null	Key	Default	Extra
facultyid	varchar(6)	NO	PRI	NULL	
facultyname	varchar(15)	NO		NULL	
noofstaff	int	YES		NULL	

3 rows in set (0.03 sec)

Create STAFF Table:

```
CREATE TABLE STAFF (  
  
    staffId VARCHAR(10) PRIMARY KEY,  
  
    staffName VARCHAR(100) NOT NULL,  
  
    staffDOB DATE NOT NULL,  
  
    staffFaculty VARCHAR(5),
```

CONSTRAINT fk_faculty FOREIGN KEY (staffFaculty) REFERENCES

FACULTY(facultyId)

);

```
University=# CREATE TABLE STAFF (  
University(#      staffId VARCHAR(10) PRIMARY KEY,  
University(#      staffName VARCHAR(100) NOT NULL,  
University(#      staffDOB DATE NOT NULL,  
University(#      staffFaculty VARCHAR(5),  
University(#      CONSTRAINT fk_faculty FOREIGN KEY (staffFaculty) REFERENCES FACULTY(facultyId)  
University(# );  
CREATE TABLE
```

```
mysql> desc staff;
```

Field	Type	Null	Key	Default	Extra
staffid	varchar(6)	NO	PRI	NULL	
staffname	varchar(15)	NO		NULL	
staffdob	date	YES		NULL	
stafffaculty	varchar(15)	YES	MUL	NULL	

4 rows in set (0.00 sec)

3. Insert Data

Insert into FACULTY Table:

INSERT INTO FACULTY (facultyId, facultyName, NoOfStaff)

VALUES

('C001', 'Computing', 120),

('E002', 'Engineering', 76),

('M002', 'Mathematics', 56),

('B001', 'Business', 89);

```

CREATE TABLE
University=# INSERT INTO FACULTY (facultyId, facultyName, NoOfStaff)
University=# VALUES
University=# ('C001', 'Computing', 120),
University=# ('E002', 'Engineering', 76),
University=# ('M002', 'Mathematics', 56),
University=# ('B001', 'Business', 89);
INSERT 0 4

```

```
mysql> select * from faculty;
```

facultyid	facultyname	noofstaff
B001	Business	89
C001	Computing	120
E002	Engineering	76
M002	Mathematics	56

4 rows in set (0.01 sec)

Insert into STAFF Table:

```
INSERT INTO STAFF (staffId, staffName, staffDOB, staffFaculty)
```

```
VALUES
```

```
('AB9872', 'Mark White', '1978-01-01', 'M002'),
```

```
('DL2314', 'Jas Singh', '1982-03-14', 'M002'),
```

```
('AF4512', 'Alison Green', '1998-12-23', 'C001'),
```

```
('BK2134', 'Kieran West', '1992-01-16', 'B001'),
```

```
('FG3124', 'Lucy Liu', '1997-08-03', 'E002');
```

```

University=# INSERT INTO STAFF (staffId, staffName, staffDOB, staffFaculty)
University=# VALUES
University=# ('AB9872', 'Mark White', '1978-01-01', 'M002'),
University=# ('DL2314', 'Jas Singh', '1982-03-14', 'M002'),
University=# ('AF4512', 'Alison Green', '1998-12-23', 'C001'),
University=# ('BK2134', 'Kieran West', '1992-01-16', 'B001'),
University=# ('FG3124', 'Lucy Liu', '1997-08-03', 'E002');
INSERT 0 5
University=#

```

```
mysql> select * from staff;
```

staffid	staffname	staffdob	stafffaculty
AB9872	Mark White	1978-01-01	M002
AF4512	Alison Green	1998-12-23	C001
BK2134	Kieran West	1992-01-16	B001
DL2314	Jas Singh	1982-03-14	M002
FG3124	Lucy Liu	1997-08-03	E002

```
5 rows in set (0.01 sec)
```

4. Query Data

a. Retrieve all data from the STAFF table:

```
SELECT * FROM STAFF;
```

```
University=# SELECT * FROM STAFF;
```

staffid	staffname	staffdob	stafffaculty
AB9872	Mark White	1978-01-01	M002
DL2314	Jas Singh	1982-03-14	M002
AF4512	Alison Green	1998-12-23	C001
BK2134	Kieran West	1992-01-16	B001
FG3124	Lucy Liu	1997-08-03	E002

```
(5 rows)
```

b. Find facultyName where NoOfStaff is less than 75:

```
SELECT facultyName
```

```
FROM FACULTY
```

```
WHERE NoOfStaff < 75;
```

```
University=# SELECT facultyName
University=# FROM FACULTY
University=# WHERE NoOfStaff < 75;
facultyname
-----
Mathematics
(1 row)
```

c. List all staff born in the 1980s:

```
SELECT staffId, staffName, staffDOB, staffFaculty
```

```
FROM STAFF
```

```
WHERE staffDOB BETWEEN '1980-01-01' AND '1989-12-31';
```

```
University=# SELECT staffId, staffName, staffDOB, staffFaculty
University=# FROM STAFF
University=# WHERE staffDOB BETWEEN '1980-01-01' AND '1989-12-31';
 staffid | staffname | staffdob | stafffaculty
-----+-----+-----+-----
DL2314  | Jas Singh | 1982-03-14 | M002
(1 row)
```

d. List all columns of the STAFF table in descending order by name (Z-A) and

rename the output headers:

```
SELECT staffId AS "Staff ID",
```

```
      staffName AS "Staff Name",
```

```
      staffDOB AS "Date of Birth",
```

```
      staffFaculty AS "Faculty"
```

```
FROM STAFF
```

```
ORDER BY staffName DESC;
```

```

University=# SELECT staffId AS "Staff ID",
University=#         staffName AS "Staff Name",
University=#         staffDOB AS "Date of Birth",
University=#         staffFaculty AS "Faculty"
University=# FROM STAFF
University=# ORDER BY staffName DESC;

```

Staff ID	Staff Name	Date of Birth	Faculty
AB9872	Mark White	1978-01-01	M002
FG3124	Lucy Liu	1997-08-03	E002
BK2134	Kieran West	1992-01-16	B001
DL2314	Jas Singh	1982-03-14	M002
AF4512	Alison Green	1998-12-23	C001

(5 rows)

e. Update Alison Green's faculty to 'Engineering':

```
UPDATE STAFF
```

```
SET staffFaculty = 'E002'
```

```
WHERE staffName = 'Alison Green';
```

```

University=# UPDATE STAFF
University=# SET staffFaculty = 'E002'
University=# WHERE staffName = 'Alison Green';
UPDATE 1
University=#

```

f. Delete the record for Kieran West:

```
DELETE FROM STAFF
```

```
WHERE staffName = 'Kieran West';
```

```

University=# DELETE FROM STAFF
University=# WHERE staffName = 'Kieran West';
DELETE 1
University=#

```

5.

-- Table: PRODUCT_TYPE

```
CREATE TABLE PRODUCT_TYPE (  
    PRODUCT_TYPE_CD VARCHAR(255) PRIMARY KEY,  
    NAME VARCHAR(50)  
);
```

```
bank_management=# CREATE TABLE PRODUCT_TYPE (  
bank_management(#    PRODUCT_TYPE_CD VARCHAR(255) PRIMARY KEY,  
bank_management(#    NAME VARCHAR(50)  
bank_management(# );  
CREATE TABLE
```

-- Table: PRODUCT

```
CREATE TABLE PRODUCT (  
    PRODUCT_CD VARCHAR(10) PRIMARY KEY,  
    DATE_OFFERED DATE,  
    DATE_RETIRED DATE,  
    NAME VARCHAR(50) NOT NULL,  
    PRODUCT_TYPE_CD VARCHAR(255),  
    FOREIGN KEY (PRODUCT_TYPE_CD) REFERENCES PRODUCT_TYPE(PRODUCT_TYPE_CD)  
);
```

```
bank_management=# CREATE TABLE PRODUCT (  
bank_management(#    PRODUCT_CD VARCHAR(10) PRIMARY KEY,  
bank_management(#    DATE_OFFERED DATE,  
bank_management(#    DATE_RETIRED DATE,  
bank_management(#    NAME VARCHAR(50) NOT NULL,  
bank_management(#    PRODUCT_TYPE_CD VARCHAR(255),  
bank_management(#    FOREIGN KEY (PRODUCT_TYPE_CD) REFERENCES PRODUCT_TYPE(PRODUCT_TYPE_CD)  
bank_management(# );  
CREATE TABLE
```

-- Table: CUSTOMER

```
CREATE TABLE CUSTOMER (  
    CUST_ID NUMERIC(10) PRIMARY KEY,  
    ADDRESS VARCHAR(30),  
    CITY VARCHAR(20),  
    CUST_TYPE_CD VARCHAR(1) NOT NULL,  
    FED_ID VARCHAR(12) NOT NULL,  
    POSTAL_CODE VARCHAR(10),  
    STATE VARCHAR(20)  
);
```

```

bank_management=# CREATE TABLE CUSTOMER (
bank_management(#      CUST_ID NUMERIC(10) PRIMARY KEY,
bank_management(#      ADDRESS VARCHAR(30),
bank_management(#      CITY VARCHAR(20),
bank_management(#      CUST_TYPE_CD VARCHAR(1) NOT NULL,
bank_management(#      FED_ID VARCHAR(12) NOT NULL,
bank_management(#      POSTAL_CODE VARCHAR(10),
bank_management(#      STATE VARCHAR(20)
bank_management(# );
CREATE TABLE
bank_management=#

```

-- Table: BRANCH

```

CREATE TABLE BRANCH (
    BRANCH_ID NUMERIC(10) PRIMARY KEY,
    ADDRESS VARCHAR(30),
    CITY VARCHAR(20),
    NAME VARCHAR(20) NOT NULL,
    STATE VARCHAR(12),
    ZIP_CODE VARCHAR(10)
);

```

```

bank_management=# CREATE TABLE BRANCH (
bank_management(#      BRANCH_ID NUMERIC(10) PRIMARY KEY,
bank_management(#      ADDRESS VARCHAR(30),
bank_management(#      CITY VARCHAR(20),
bank_management(#      NAME VARCHAR(20) NOT NULL,
bank_management(#      STATE VARCHAR(12),
bank_management(#      ZIP_CODE VARCHAR(10)
bank_management(# );
CREATE TABLE
bank_management=#

```

-- Table: DEPARTMENT

```

CREATE TABLE DEPARTMENT (
    DEPT_ID NUMERIC(10) PRIMARY KEY,
    NAME VARCHAR(20) NOT NULL
);

```

```

bank_management=# CREATE TABLE DEPARTMENT (
bank_management(#      DEPT_ID NUMERIC(10) PRIMARY KEY,
bank_management(#      NAME VARCHAR(20) NOT NULL
bank_management(# );
CREATE TABLE

```

-- Table: EMPLOYEE

```

CREATE TABLE EMPLOYEE (
    EMP_ID NUMERIC(10) PRIMARY KEY,
    END_DATE DATE,
    FIRST_NAME VARCHAR(20) NOT NULL,

```



```

        LAST_NAME VARCHAR(20) NOT NULL,
        START_DATE DATE NOT NULL,
        TITLE VARCHAR(20),
        ASSIGNED_BRANCH_ID NUMERIC(10),
        DEPT_ID NUMERIC(10),
        SUPERIOR_EMP_ID NUMERIC(10),
        FOREIGN KEY (ASSIGNED_BRANCH_ID) REFERENCES BRANCH(BRANCH_ID),
        FOREIGN KEY (DEPT_ID) REFERENCES DEPARTMENT(DEPT_ID),
        FOREIGN KEY (SUPERIOR_EMP_ID) REFERENCES EMPLOYEE(EMP_ID)
    );

bank_management=# CREATE TABLE EMPLOYEE (
bank_management(#      EMP_ID NUMERIC(10) PRIMARY KEY,
bank_management(#      END_DATE DATE,
bank_management(#      FIRST_NAME VARCHAR(20) NOT NULL,
bank_management(#      LAST_NAME VARCHAR(20) NOT NULL,
bank_management(#      START_DATE DATE NOT NULL,
bank_management(#      TITLE VARCHAR(20),
bank_management(#      ASSIGNED_BRANCH_ID NUMERIC(10),
bank_management(#      DEPT_ID NUMERIC(10),
bank_management(#      SUPERIOR_EMP_ID NUMERIC(10),
bank_management(#      FOREIGN KEY (ASSIGNED_BRANCH_ID) REFERENCES BRANCH(BRANCH_ID),
bank_management(#      FOREIGN KEY (DEPT_ID) REFERENCES DEPARTMENT(DEPT_ID),
bank_management(#      FOREIGN KEY (SUPERIOR_EMP_ID) REFERENCES EMPLOYEE(EMP_ID)
bank_management(# );
CREATE TABLE
bank_management=#

```

-- Table: ACCOUNT

```

CREATE TABLE ACCOUNT (
    ACCOUNT_ID SERIAL PRIMARY KEY,
    AVAIL_BALANCE NUMERIC(14, 2),
    CLOSE_DATE DATE,
    LAST_ACTIVITY_DATE DATE,
    OPEN_DATE DATE NOT NULL,
    PENDING_BALANCE NUMERIC(14, 2),
    STATUS VARCHAR(10),
    CUST_ID NUMERIC(10),
    OPEN_BRANCH_ID NUMERIC(10) NOT NULL,
    OPEN_EMP_ID NUMERIC(10) NOT NULL,
    PRODUCT_CD VARCHAR(10) NOT NULL,
    FOREIGN KEY (CUST_ID) REFERENCES CUSTOMER(CUST_ID),
    FOREIGN KEY (OPEN_BRANCH_ID) REFERENCES BRANCH(BRANCH_ID),
    FOREIGN KEY (OPEN_EMP_ID) REFERENCES EMPLOYEE(EMP_ID),
    FOREIGN KEY (PRODUCT_CD) REFERENCES PRODUCT(PRODUCT_CD)
);

```

```

bank_management=# CREATE TABLE ACCOUNT (
bank_management(# ACCOUNT_ID SERIAL PRIMARY KEY,
bank_management(# AVAIL_BALANCE NUMERIC(14, 2),
bank_management(# CLOSE_DATE DATE,
bank_management(# LAST_ACTIVITY_DATE DATE,
bank_management(# OPEN_DATE DATE NOT NULL,
bank_management(# PENDING_BALANCE NUMERIC(14, 2),
bank_management(# STATUS VARCHAR(10),
bank_management(# CUST_ID NUMERIC(10),
bank_management(# OPEN_BRANCH_ID NUMERIC(10) NOT NULL,
bank_management(# OPEN_EMP_ID NUMERIC(10) NOT NULL,
bank_management(# PRODUCT_CD VARCHAR(10) NOT NULL,
bank_management(# FOREIGN KEY (CUST_ID) REFERENCES CUSTOMER(CUST_ID),
bank_management(# FOREIGN KEY (OPEN_BRANCH_ID) REFERENCES BRANCH(BRANCH_ID),
bank_management(# FOREIGN KEY (OPEN_EMP_ID) REFERENCES EMPLOYEE(EMP_ID),
bank_management(# FOREIGN KEY (PRODUCT_CD) REFERENCES PRODUCT(PRODUCT_CD)
bank_management(# );
CREATE TABLE

```

-- Table: BUSINESS

```

CREATE TABLE BUSINESS (
    CUST_ID NUMERIC(10) PRIMARY KEY,
    INCORP_DATE DATE,
    NAME VARCHAR(255) NOT NULL,
    STATE_ID VARCHAR(10) NOT NULL,
    FOREIGN KEY (CUST_ID) REFERENCES CUSTOMER(CUST_ID)
);

```

```

bank_management=# CREATE TABLE BUSINESS (
bank_management(# CUST_ID NUMERIC(10) PRIMARY KEY,
bank_management(# INCORP_DATE DATE,
bank_management(# NAME VARCHAR(255) NOT NULL,
bank_management(# STATE_ID VARCHAR(10) NOT NULL,
bank_management(# FOREIGN KEY (CUST_ID) REFERENCES CUSTOMER(CUST_ID)
bank_management(# );
CREATE TABLE

```

-- Table: INDIVIDUAL

```

CREATE TABLE INDIVIDUAL (
    CUST_ID NUMERIC(10) PRIMARY KEY,
    BIRTH_DATE DATE,
    FIRST_NAME VARCHAR(30) NOT NULL,
    LAST_NAME VARCHAR(30) NOT NULL,
    FOREIGN KEY (CUST_ID) REFERENCES CUSTOMER(CUST_ID)
);

```

```

CREATE TABLE
bank_management=# CREATE TABLE INDIVIDUAL (
bank_management(# CUST_ID NUMERIC(10) PRIMARY KEY,
bank_management(# BIRTH_DATE DATE,
bank_management(# FIRST_NAME VARCHAR(30) NOT NULL,
bank_management(# LAST_NAME VARCHAR(30) NOT NULL,
bank_management(# FOREIGN KEY (CUST_ID) REFERENCES CUSTOMER(CUST_ID)
bank_management(# );
CREATE TABLE

```

-- Table: OFFICER

```

CREATE TABLE OFFICER (

```

```

OFFICER_ID NUMERIC(10) PRIMARY KEY,
END_DATE DATE,
FIRST_NAME VARCHAR(30) NOT NULL,
LAST_NAME VARCHAR(30) NOT NULL,
START_DATE DATE NOT NULL,
TITLE VARCHAR(20),
CUST_ID NUMERIC(10),
FOREIGN KEY (CUST_ID) REFERENCES CUSTOMER(CUST_ID)
);

```

```

bank_management=# CREATE TABLE OFFICER (
bank_management(# OFFICER_ID NUMERIC(10) PRIMARY KEY,
bank_management(# END DATE DATE,
bank_management(# FIRST_NAME VARCHAR(30) NOT NULL,
bank_management(# LAST_NAME VARCHAR(30) NOT NULL,
bank_management(# START_DATE DATE NOT NULL,
bank_management(# TITLE VARCHAR(20),
bank_management(# CUST_ID NUMERIC(10),
bank_management(# FOREIGN KEY (CUST_ID) REFERENCES CUSTOMER(CUST_ID)
bank_management(# );
CREATE TABLE

```

-- Table: ACC_TRANSACTION

```

CREATE TABLE ACC_TRANSACTION (
    TXN_ID SERIAL PRIMARY KEY,
    AMOUNT NUMERIC(14, 2) NOT NULL,
    FUNDS_AVAIL_DATE TIMESTAMP NOT NULL,
    TXN_DATE TIMESTAMP NOT NULL,
    TXN_TYPE_CD VARCHAR(10),
    ACCOUNT_ID INTEGER,
    EXECUTION_BRANCH_ID NUMERIC(10),
    TELLER_EMP_ID NUMERIC(10),
    FOREIGN KEY (ACCOUNT_ID) REFERENCES ACCOUNT(ACCOUNT_ID),
    FOREIGN KEY (EXECUTION_BRANCH_ID) REFERENCES BRANCH(BRANCH_ID),
    FOREIGN KEY (TELLER_EMP_ID) REFERENCES EMPLOYEE(EMP_ID)
);

```

```

bank_management=# CREATE TABLE ACC_TRANSACTION (
bank_management(# TXN_ID SERIAL PRIMARY KEY,
bank_management(# AMOUNT NUMERIC(14, 2) NOT NULL,
bank_management(# FUNDS_AVAIL_DATE TIMESTAMP NOT NULL,
bank_management(# TXN_DATE TIMESTAMP NOT NULL,
bank_management(# TXN_TYPE_CD VARCHAR(10),
bank_management(# ACCOUNT_ID INTEGER,
bank_management(# EXECUTION_BRANCH_ID NUMERIC(10),
bank_management(# TELLER_EMP_ID NUMERIC(10),
bank_management(# FOREIGN KEY (ACCOUNT_ID) REFERENCES ACCOUNT(ACCOUNT_ID),
bank_management(# FOREIGN KEY (EXECUTION_BRANCH_ID) REFERENCES BRANCH(BRANCH_ID),
bank_management(# FOREIGN KEY (TELLER_EMP_ID) REFERENCES EMPLOYEE(EMP_ID)
bank_management(# );
CREATE TABLE
bank_management=#

```

```

INSERT INTO PRODUCT_TYPE (PRODUCT_TYPE_CD, NAME) VALUES
('ACCOUNT', 'Customer Accounts'),
('LOAN', 'Individual and Business Loans'),
('INSURANCE', 'Insurance Offerings');

```

```

bank_management=# INSERT INTO PRODUCT_TYPE (PRODUCT_TYPE_CD, NAME) VALUES
bank_management=# ('ACCOUNT', 'Customer Accounts'),
bank_management=# ('LOAN', 'Individual and Business Loans'),
bank_management=# ('INSURANCE', 'Insurance Offerings');
INSERT 0 3
bank_management=#

```

```

INSERT INTO PRODUCT (PRODUCT_CD, DATE_OFFERED, DATE_RETIRED, NAME,
PRODUCT_TYPE_CD) VALUES
('CHK', '2000-01-01', NULL, 'checking account', 'ACCOUNT'),
('SAV', '2000-01-01', NULL, 'savings account', 'ACCOUNT'),
('MM', '2000-01-01', NULL, 'money market account', 'ACCOUNT'),
('CD', '2000-01-01', NULL, 'certificate of deposit', 'ACCOUNT'),
('MRT', '2000-01-01', NULL, 'home mortgage', 'LOAN'),
('AUT', '2000-01-01', NULL, 'auto loan', 'LOAN'),
('BUS', '2000-01-01', NULL, 'business line of credit', 'LOAN'),
('SBL', '2000-01-01', NULL, 'small business loan', 'LOAN');

```

```

bank_management=# INSERT INTO PRODUCT (PRODUCT_CD, DATE_OFFERED, DATE_RETIRED, NAME, PRODUCT_TYPE_CD) VALUES
bank_management=# ('CHK', '2000-01-01', NULL, 'checking account', 'ACCOUNT'),
bank_management=# ('SAV', '2000-01-01', NULL, 'savings account', 'ACCOUNT'),
bank_management=# ('MM', '2000-01-01', NULL, 'money market account', 'ACCOUNT'),
bank_management=# ('CD', '2000-01-01', NULL, 'certificate of deposit', 'ACCOUNT'),
bank_management=# ('MRT', '2000-01-01', NULL, 'home mortgage', 'LOAN'),
bank_management=# ('AUT', '2000-01-01', NULL, 'auto loan', 'LOAN'),
bank_management=# ('BUS', '2000-01-01', NULL, 'business line of credit', 'LOAN'),
bank_management=# ('SBL', '2000-01-01', NULL, 'small business loan', 'LOAN');
INSERT 0 8
bank_management=#

```

```

INSERT INTO CUSTOMER (CUST_ID, ADDRESS, CITY, CUST_TYPE_CD, FED_ID, POSTAL_CODE,
STATE) VALUES
(1, '47 Mockingbird Ln', 'Lynnfield', 'I', '111-11-1111', '1940', 'MA'),
(2, '372 Clearwater Blvd', 'Woburn', 'I', '222-22-2222', '1801', 'MA'),
(3, '18 Jessup Rd', 'Quincy', 'I', '333-33-3333', '2169', 'MA'),
(4, '12 Buchanan Ln', 'Waltham', 'I', '444-44-4444', '2451', 'MA'),
(5, '2341 Main St', 'Salem', 'I', '555-55-5555', '3079', 'NH'),
(6, '12 Blaylock Ln', 'Waltham', 'I', '666-66-6666', '2451', 'MA'),
(7, '29 Admiral Ln', 'Wilmington', 'I', '777-77-7777', '1887', 'MA'),
(8, '472 Freedom Rd', 'Salem', 'I', '888-88-8888', '3079', 'NH'),
(9, '29 Maple St', 'Newton', 'I', '999-99-9999', '2458', 'MA'),
(10, '7 Industrial Way', 'Salem', 'B', '04-1111111', '3079', 'NH'),
(11, '287A Corporate Ave', 'Wilmington', 'B', '04-2222222', '1887', 'MA'),
(12, '789 Main St', 'Salem', 'B', '04-3333333', '3079', 'NH'),
(13, '4772 Presidential Way', 'Quincy', 'B', '04-4444444', '2169', 'MA');

```

```

bank_management=# INSERT INTO CUSTOMER (CUST_ID, ADDRESS, CITY, CUST_TYPE_CD, FED_ID, POSTAL_CODE, STATE) VALUES
bank_management=# (1, '47 Mockingbird Ln', 'Lynnfield', 'I', '111-11-1111', '1940', 'MA'),
bank_management=# (2, '372 Clearwater Blvd', 'Woburn', 'I', '222-22-2222', '1801', 'MA'),
bank_management=# (3, '18 Jessup Rd', 'Quincy', 'I', '333-33-3333', '2169', 'MA'),
bank_management=# (4, '12 Buchanan Ln', 'Waltham', 'I', '444-44-4444', '2451', 'MA'),
bank_management=# (5, '2341 Main St', 'Salem', 'I', '555-55-5555', '3079', 'NH'),
bank_management=# (6, '12 Blaylock Ln', 'Waltham', 'I', '666-66-6666', '2451', 'MA'),
bank_management=# (7, '29 Admiral Ln', 'Wilmington', 'I', '777-77-7777', '1887', 'MA'),
bank_management=# (8, '472 Freedom Rd', 'Salem', 'I', '888-88-8888', '3079', 'NH'),
bank_management=# (9, '29 Maple St', 'Newton', 'I', '999-99-9999', '2458', 'MA'),
bank_management=# (10, '7 Industrial Way', 'Salem', 'B', '04-111111', '3079', 'NH'),
bank_management=# (11, '287A Corporate Ave', 'Wilmington', 'B', '04-222222', '1887', 'MA'),
bank_management=# (12, '789 Main St', 'Salem', 'B', '04-333333', '3079', 'NH'),
bank_management=# (13, '4772 Presidential Way', 'Quincy', 'B', '04-444444', '2169', 'MA');
INSERT 0 13
bank_management=#

```

INSERT INTO BRANCH (BRANCH_ID, ADDRESS, CITY, NAME, STATE, ZIP_CODE) VALUES

- (1, '3882 Main St.', 'Waltham', 'Headquarters', 'MA', '2451'),
- (2, '422 Maple St.', 'Woburn', 'Woburn Branch', 'MA', '1801'),
- (3, '125 Presidential Way', 'Quincy', 'Quincy Branch', 'MA', '2169'),
- (4, '378 Maynard Ln.', 'Salem', 'So. NH Branch', 'NH', '3079');

```

bank_management=# INSERT INTO BRANCH (BRANCH_ID, ADDRESS, CITY, NAME, STATE, ZIP_CODE) VALUES
bank_management=# (1, '3882 Main St.', 'Waltham', 'Headquarters', 'MA', '2451'),
bank_management=# (2, '422 Maple St.', 'Woburn', 'Woburn Branch', 'MA', '1801'),
bank_management=# (3, '125 Presidential Way', 'Quincy', 'Quincy Branch', 'MA', '2169'),
bank_management=# (4, '378 Maynard Ln.', 'Salem', 'So. NH Branch', 'NH', '3079');
INSERT 0 4

```

INSERT INTO DEPARTMENT (DEPT_ID, NAME) VALUES

- (1, 'Operations'),
- (2, 'Loans'),
- (3, 'Administration'),
- (4, 'IT');

```

bank_management=# INSERT INTO DEPARTMENT (DEPT_ID, NAME) VALUES
bank_management=# (1, 'Operations'),
bank_management=# (2, 'Loans'),
bank_management=# (3, 'Administration'),
bank_management=# (4, 'IT');
INSERT 0 4

```

INSERT INTO EMPLOYEE (EMP_ID, END_DATE, FIRST_NAME, LAST_NAME, START_DATE, TITLE, ASSIGNED_BRANCH_ID, DEPT_ID, SUPERIOR_EMP_ID) VALUES

- (1, NULL, 'Michael', 'Smith', '2001-06-22', 'President', 1, 3, NULL),
- (2, NULL, 'Susan', 'Barker', '2002-09-12', 'Vice President', 1, 3, 1),
- (3, NULL, 'Robert', 'Tyler', '2002-02-09', 'Treasurer', 1, 3, 1),
- (4, NULL, 'Susan', 'Hawthorne', '2004-04-24', 'Operations Manager', 1, 1, 3),
- (5, NULL, 'John', 'Gooding', '2003-11-14', 'Loan Manager', 1, 2, 4),
- (6, NULL, 'Helen', 'Fleming', '2004-03-17', 'Head Teller', 1, 1, 4),
- (7, NULL, 'Chris', 'Tucker', '2004-09-15', 'Teller', 1, 1, 6),
- (8, NULL, 'Sarah', 'Parker', '2002-12-02', 'Teller', 1, 1, 6),
- (9, NULL, 'Jane', 'Grossman', '2002-05-03', 'Teller', 1, 1, 6),
- (10, NULL, 'Paula', 'Roberts', '2002-07-27', 'Head Teller', 2, 1, 4),
- (11, NULL, 'Thomas', 'Ziegler', '2000-10-23', 'Teller', 2, 1, 10),
- (12, NULL, 'Samantha', 'Jameson', '2003-01-08', 'Teller', 2, 1, 10),
- (13, NULL, 'John', 'Blake', '2011-05-11', 'Head Teller', 3, 1, 4),
- (14, NULL, 'Cindy', 'Mason', '2002-08-09', 'Teller', 3, 1, 13),

(15, NULL, 'Frank', 'Portman', '2003-04-01', 'Teller', 3, 1, 13),
 (16, NULL, 'Theresa', 'Markham', '2001-03-15', 'Head Teller', 4, 1, 4),
 (17, NULL, 'Beth', 'Fowler', '2002-06-29', 'Teller', 4, 1, 16),
 (18, NULL, 'Rick', 'Tulman', '2002-12-12', 'Teller', 4, 1, 16);

```
bank_management=# INSERT INTO EMPLOYEE (EMP_ID, END_DATE, FIRST_NAME, LAST_NAME, START_DATE, TITLE, ASSIGNED_BRANCH_ID,
DEPT_ID, SUPERIOR_EMP_ID) VALUES
bank_management=# (1, NULL, 'Michael', 'Smith', '2001-06-22', 'President', 1, 3, NULL),
bank_management=# (2, NULL, 'Susan', 'Barker', '2002-09-12', 'Vice President', 1, 3, 1),
bank_management=# (3, NULL, 'Robert', 'Tyler', '2002-02-09', 'Treasurer', 1, 3, 1),
bank_management=# (4, NULL, 'Susan', 'Hawthorne', '2004-04-24', 'Operations Manager', 1, 1, 3),
bank_management=# (5, NULL, 'John', 'Gooding', '2003-11-14', 'Loan Manager', 1, 2, 4),
bank_management=# (6, NULL, 'Helen', 'Fleming', '2004-03-17', 'Head Teller', 1, 1, 4),
bank_management=# (7, NULL, 'Chris', 'Tucker', '2004-09-15', 'Teller', 1, 1, 6),
bank_management=# (8, NULL, 'Sarah', 'Parker', '2002-12-02', 'Teller', 1, 1, 6),
bank_management=# (9, NULL, 'Jane', 'Grossman', '2002-05-03', 'Teller', 1, 1, 6),
bank_management=# (10, NULL, 'Paula', 'Roberts', '2002-07-27', 'Head Teller', 2, 1, 4),
bank_management=# (11, NULL, 'Thomas', 'Ziegler', '2000-10-23', 'Teller', 2, 1, 10),
bank_management=# (12, NULL, 'Samantha', 'Jameson', '2003-01-03', 'Teller', 2, 1, 10),
bank_management=# (13, NULL, 'John', 'Blake', '2011-05-11', 'Head Teller', 3, 1, 4),
bank_management=# (14, NULL, 'Cindy', 'Mason', '2002-08-09', 'Teller', 3, 1, 13),
bank_management=# (15, NULL, 'Frank', 'Portman', '2003-04-01', 'Teller', 3, 1, 13),
bank_management=# (16, NULL, 'Theresa', 'Markham', '2001-03-15', 'Head Teller', 4, 1, 4),
bank_management=# (17, NULL, 'Beth', 'Fowler', '2002-06-29', 'Teller', 4, 1, 16),
bank_management=# (18, NULL, 'Rick', 'Tulman', '2002-12-12', 'Teller', 4, 1, 16);
INSERT 0 18
bank_management=#
```

```
INSERT INTO ACCOUNT (ACCOUNT_ID, AVAIL_BALANCE, CLOSE_DATE,
LAST_ACTIVITY_DATE, OPEN_DATE, PENDING_BALANCE, STATUS, CUST_ID,
OPEN_BRANCH_ID, OPEN_EMP_ID, PRODUCT_CD) VALUES
(1, 1057.75, NULL, '2005-01-04', '2000-01-15', 1057.75, 'ACTIVE', 1, 2, 10, 'CHK'),
(2, 500.00, NULL, '2004-12-19', '2000-01-15', 500.00, 'ACTIVE', 1, 2, 10, 'SAV'),
(3, 3000.00, NULL, '2004-06-30', '2004-06-30', 3000.00, 'ACTIVE', 1, 2, 10, 'CD'),
(4, 2258.02, NULL, '2004-12-27', '2001-03-12', 2258.02, 'ACTIVE', 2, 2, 11, 'CHK'),
(5, 200.00, NULL, '2004-12-11', '2001-03-12', 200.00, 'ACTIVE', 2, 2, 11, 'SAV'),
(6, 1057.75, NULL, '2004-11-30', '2002-11-23', 1057.75, 'ACTIVE', 3, 3, 14, 'CHK'),
(7, 2212.50, NULL, '2004-12-05', '2002-12-15', 2212.50, 'ACTIVE', 3, 3, 14, 'MM'),
(8, 534.12, NULL, '2005-01-03', '2003-09-12', 534.12, 'ACTIVE', 4, 1, 8, 'CHK'),
(9, 767.77, NULL, '2004-10-24', '2000-01-15', 767.77, 'ACTIVE', 4, 1, 8, 'SAV'),
(10, 5487.09, NULL, '2004-11-11', '2004-09-30', 5487.09, 'ACTIVE', 4, 1, 8, 'MM'),
(11, 2237.97, NULL, '2005-01-05', '2004-01-27', 2897.97, 'ACTIVE', 5, 4, 17, 'CHK'),
(12, 122.37, NULL, '2004-11-29', '2002-08-24', 122.37, 'ACTIVE', 6, 1, 7, 'CHK'),
(13, 10000.00, NULL, '2004-12-28', '2004-12-28', 10000.00, 'ACTIVE', 6, 1, 7, 'CD'),
(14, 5000.00, NULL, '2004-01-12', '2004-01-12', 5000.00, 'ACTIVE', 7, 2, 12, 'CD'),
(15, 3487.19, NULL, '2005-01-03', '2001-05-23', 3487.19, 'ACTIVE', 8, 4, 18, 'CHK'),
(16, 387.99, NULL, '2004-10-12', '2001-05-23', 387.99, 'ACTIVE', 8, 4, 18, 'SAV'),
(17, 125.67, NULL, '2004-12-15', '2003-07-30', 125.67, 'ACTIVE', 9, 1, 9, 'CHK'),
(18, 9345.55, NULL, '2004-10-28', '2004-10-28', 9845.55, 'ACTIVE', 9, 1, 9, 'MM'),
(19, 1500.00, NULL, '2004-06-30', '2004-06-30', 1500.00, 'ACTIVE', 9, 1, 9, 'CD'),
(20, 23575.12, NULL, '2004-12-15', '2002-09-30', 23575.12, 'ACTIVE', 10, 4, 16, 'CHK'),
(21, 0.00, NULL, '2004-08-28', '2002-10-01', 0.00, 'ACTIVE', 10, 4, 16, 'BUS'),
(22, 38552.05, NULL, '2004-12-15', '2003-07-30', 38552.05, 'ACTIVE', 12, 4, 16, 'CHK'),
(23, 50000.00, NULL, '2004-12-17', '2004-04-22', 50000.00, 'ACTIVE', 13, 3, 15, 'SBL');
```

```

bank_management=# INSERT INTO ACCOUNT (ACCOUNT_ID, AVAIL_BALANCE, CLOSE_DATE, LAST_ACTIVITY_DATE, OPEN_DATE, PENDING_BALANCE, STATUS, CUST_ID, OPEN_BRANCH_ID, OPEN_EMP_ID, PRODUCT_CD) VALUES
bank_management=# (1, 1057.75, NULL, '2005-01-04', '2000-01-15', 1057.75, 'ACTIVE', 1, 2, 10, 'CHK'),
bank_management=# (2, 500.00, NULL, '2004-12-19', '2000-01-15', 500.00, 'ACTIVE', 1, 2, 10, 'SAV'),
bank_management=# (3, 3000.00, NULL, '2004-06-30', '2004-06-30', 3000.00, 'ACTIVE', 1, 2, 10, 'CD'),
bank_management=# (4, 2258.02, NULL, '2004-12-27', '2001-03-12', 2258.02, 'ACTIVE', 2, 2, 11, 'CHK'),
bank_management=# (5, 200.00, NULL, '2004-12-11', '2001-03-12', 200.00, 'ACTIVE', 2, 2, 11, 'SAV'),
bank_management=# (6, 1057.75, NULL, '2004-11-30', '2002-11-23', 1057.75, 'ACTIVE', 3, 3, 14, 'CHK'),
bank_management=# (7, 2212.50, NULL, '2004-12-05', '2002-12-15', 2212.50, 'ACTIVE', 3, 3, 14, 'MM'),
bank_management=# (8, 534.12, NULL, '2005-01-03', '2003-09-12', 534.12, 'ACTIVE', 4, 1, 8, 'CHK'),
bank_management=# (9, 767.77, NULL, '2004-10-24', '2000-01-15', 767.77, 'ACTIVE', 4, 1, 8, 'SAV'),
bank_management=# (10, 5437.09, NULL, '2004-11-11', '2004-09-30', 5437.09, 'ACTIVE', 4, 1, 8, 'MM'),
bank_management=# (11, 2237.97, NULL, '2005-01-05', '2004-01-27', 2237.97, 'ACTIVE', 5, 4, 17, 'CHK'),
bank_management=# (12, 122.37, NULL, '2004-11-29', '2002-02-24', 122.37, 'ACTIVE', 5, 4, 17, 'CHK'),
bank_management=# (13, 10000.00, NULL, '2004-12-28', '2004-12-28', 10000.00, 'ACTIVE', 6, 1, 7, 'CD'),
bank_management=# (14, 5000.00, NULL, '2004-01-12', '2004-01-12', 5000.00, 'ACTIVE', 7, 2, 12, 'CD'),
bank_management=# (15, 3437.19, NULL, '2005-01-03', '2001-05-23', 3437.19, 'ACTIVE', 8, 4, 18, 'CHK'),
bank_management=# (16, 387.99, NULL, '2004-10-12', '2001-05-23', 387.99, 'ACTIVE', 8, 4, 18, 'SAV'),
bank_management=# (17, 125.67, NULL, '2004-12-15', '2003-07-30', 125.67, 'ACTIVE', 9, 1, 9, 'CHK'),
bank_management=# (18, 9345.55, NULL, '2004-10-28', '2004-10-28', 9345.55, 'ACTIVE', 9, 1, 9, 'MM'),
bank_management=# (19, 1500.00, NULL, '2004-06-30', '2004-06-30', 1500.00, 'ACTIVE', 9, 1, 9, 'CD'),
bank_management=# (20, 23575.12, NULL, '2004-12-15', '2002-09-30', 23575.12, 'ACTIVE', 10, 4, 16, 'CHK'),
bank_management=# (21, 0.00, NULL, '2004-08-28', '2002-10-01', 0.00, 'ACTIVE', 10, 4, 16, 'BUS'),
bank_management=# (22, 38552.05, NULL, '2004-12-15', '2003-07-30', 38552.05, 'ACTIVE', 12, 4, 16, 'CHK'),
bank_management=# (23, 50000.00, NULL, '2004-12-17', '2004-04-22', 50000.00, 'ACTIVE', 15, 3, 15, 'SEL'),
INSERT 0 23

```

INSERT INTO BUSINESS (INCORP_DATE, NAME, STATE_ID, CUST_ID) VALUES

```

('1995-05-01', 'Chilton Engineering', '12-345-678', 10),
('2001-01-01', 'Northeast Cooling Inc.', '23-456-789', 11),
('2002-06-30', 'Superior Auto Body', '34-567-890', 12),
('1999-05-01', 'AAA Insurance Inc.', '45-678-901', 13);

```

```

bank_management=# INSERT INTO BUSINESS (INCORP_DATE, NAME, STATE_ID, CUST_ID) VALUES
bank_management=# ('1995-05-01', 'Chilton Engineering', '12-345-678', 10),
bank_management=# ('2001-01-01', 'Northeast Cooling Inc.', '23-456-789', 11),
bank_management=# ('2002-06-30', 'Superior Auto Body', '34-567-890', 12),
bank_management=# ('1999-05-01', 'AAA Insurance Inc.', '45-678-901', 13);
INSERT 0 4

```

INSERT INTO INDIVIDUAL (CUST_ID, BIRTH_DATE, FIRST_NAME, LAST_NAME) VALUES

```

(1, '1972-04-22', 'James', 'Hadley'),
(2, '1968-08-15', 'Susan', 'Tingley'),
(3, '1958-02-06', 'Frank', 'Tucker'),
(4, '1966-12-22', 'John', 'Hayward'),
(5, '1971-08-25', 'Charles', 'Frasier'),
(6, '1962-09-14', 'John', 'Spencer'),
(7, '1947-03-19', 'Margaret', 'Young'),
(8, '1977-07-01', 'Louis', 'Blake'),
(9, '1968-06-16', 'Richard', 'Farley');

```

```

bank_management=# INSERT INTO INDIVIDUAL (CUST_ID, BIRTH_DATE, FIRST_NAME, LAST_NAME) VALUES
bank_management=# (1, '1972-04-22', 'James', 'Hadley'),
bank_management=# (2, '1968-08-15', 'Susan', 'Tingley'),
bank_management=# (3, '1958-02-06', 'Frank', 'Tucker'),
bank_management=# (4, '1966-12-22', 'John', 'Hayward'),
bank_management=# (5, '1971-08-25', 'Charles', 'Frasier'),
bank_management=# (6, '1962-09-14', 'John', 'Spencer'),
bank_management=# (7, '1947-03-19', 'Margaret', 'Young'),
bank_management=# (8, '1977-07-01', 'Louis', 'Blake'),
bank_management=# (9, '1968-06-16', 'Richard', 'Farley');
INSERT 0 9

```

INSERT INTO OFFICER (OFFICER_ID, FIRST_NAME, LAST_NAME, START_DATE, END_DATE, TITLE, CUST_ID) VALUES

```

(1, 'John', 'Chilton', '1995-05-01', NULL, 'President', 10),
(2, 'Paul', 'Hardy', '2001-01-01', NULL, 'President', 11),
(3, 'Carl', 'Lutz', '2002-06-30', NULL, 'President', 12),
(4, 'Stanley', 'Cheswick', '1999-05-01', NULL, 'President', 13);

```

```

bank_management=# INSERT INTO OFFICER (OFFICER_ID, FIRST_NAME, LAST_NAME, START_DATE, END_DATE, TITLE, CUST_ID) VALUES
bank_management=# (1, 'John', 'Chilton', '1995-05-01', NULL, 'President', 10),
bank_management=# (2, 'Paul', 'Hardy', '2001-01-01', NULL, 'President', 11),
bank_management=# (3, 'Carl', 'Lutz', '2002-06-30', NULL, 'President', 12),
bank_management=# (4, 'Stanley', 'Cheswick', '1999-05-01', NULL, 'President', 13);
INSERT 0 4

```

```

INSERT INTO ACC_TRANSACTION (TXN_ID, AMOUNT, FUNDS_AVAIL_DATE, TXN_DATE,
TXN_TYPE_CD, ACCOUNT_ID, EXECUTION_BRANCH_ID, TELLER_EMP_ID) VALUES
(1, 100, '2000-01-15', '2000-01-15', 'CDT', 1, NULL, NULL),
(2, 100, '2000-01-15', '2000-01-15', 'CDT', 2, NULL, NULL),
(3, 100, '2004-06-30', '2004-06-30', 'CDT', 3, NULL, NULL),
(4, 100, '2001-03-12', '2001-03-12', 'CDT', 4, NULL, NULL),
(5, 100, '2001-03-12', '2001-03-12', 'CDT', 5, NULL, NULL),
(6, 100, '2002-11-23', '2002-11-23', 'CDT', 6, NULL, NULL),
(7, 100, '2002-12-15', '2002-12-15', 'CDT', 7, NULL, NULL),
(8, 100, '2003-09-12', '2003-09-12', 'CDT', 8, NULL, NULL),
(9, 100, '2000-01-15', '2000-01-15', 'CDT', 9, NULL, NULL),
(10, 100, '2004-09-30', '2004-09-30', 'CDT', 10, NULL, NULL),
(11, 100, '2004-01-27', '2004-01-27', 'CDT', 11, NULL, NULL),
(12, 100, '2002-08-24', '2002-08-24', 'CDT', 12, NULL, NULL),
(13, 100, '2004-12-28', '2004-12-28', 'CDT', 13, NULL, NULL),
(14, 100, '2004-01-12', '2004-01-12', 'CDT', 14, NULL, NULL),
(15, 100, '2001-05-23', '2001-05-23', 'CDT', 15, NULL, NULL),
(16, 100, '2001-05-23', '2001-05-23', 'CDT', 16, NULL, NULL),
(17, 100, '2003-07-30', '2003-07-30', 'CDT', 17, NULL, NULL),
(18, 100, '2004-10-28', '2004-10-28', 'CDT', 18, NULL, NULL),
(19, 100, '2004-06-30', '2004-06-30', 'CDT', 19, NULL, NULL),
(20, 100, '2002-09-30', '2002-09-30', 'CDT', 20, NULL, NULL),
(21, 100, '2003-07-30', '2003-07-30', 'CDT', 22, NULL, NULL);

```

```

bank_management-# INSERT INTO ACC_TRANSACTION (TXN_ID, AMOUNT, FUNDS_AVAIL_DATE, TXN_DATE, TXN_TYPE_CD, ACCOUNT_ID, EXECUTION_BRANCH_ID, TELLER_EMP_ID) VALUES
bank_management-# (1, 100, '2000-01-15', '2000-01-15', 'CDT', 1, NULL, NULL),
bank_management-# (2, 100, '2000-01-15', '2000-01-15', 'CDT', 2, NULL, NULL),
bank_management-# (3, 100, '2004-06-30', '2004-06-30', 'CDT', 3, NULL, NULL),
bank_management-# (4, 100, '2001-03-12', '2001-03-12', 'CDT', 4, NULL, NULL),
bank_management-# (5, 100, '2001-03-12', '2001-03-12', 'CDT', 5, NULL, NULL),
bank_management-# (6, 100, '2002-11-23', '2002-11-23', 'CDT', 6, NULL, NULL),
bank_management-# (7, 100, '2002-12-15', '2002-12-15', 'CDT', 7, NULL, NULL),
bank_management-# (8, 100, '2003-09-12', '2003-09-12', 'CDT', 8, NULL, NULL),
bank_management-# (9, 100, '2000-01-15', '2000-01-15', 'CDT', 9, NULL, NULL),
bank_management-# (10, 100, '2004-09-30', '2004-09-30', 'CDT', 10, NULL, NULL),
bank_management-# (11, 100, '2004-01-27', '2004-01-27', 'CDT', 11, NULL, NULL),
bank_management-# (12, 100, '2002-08-24', '2002-08-24', 'CDT', 12, NULL, NULL),
bank_management-# (13, 100, '2004-12-28', '2004-12-28', 'CDT', 13, NULL, NULL),
bank_management-# (14, 100, '2004-01-12', '2004-01-12', 'CDT', 14, NULL, NULL),
bank_management-# (15, 100, '2001-05-23', '2001-05-23', 'CDT', 15, NULL, NULL),
bank_management-# (16, 100, '2001-05-23', '2001-05-23', 'CDT', 16, NULL, NULL),
bank_management-# (17, 100, '2003-07-30', '2003-07-30', 'CDT', 17, NULL, NULL),
bank_management-# (18, 100, '2004-10-28', '2004-10-28', 'CDT', 18, NULL, NULL),
bank_management-# (19, 100, '2004-06-30', '2004-06-30', 'CDT', 19, NULL, NULL),
bank_management-# (20, 100, '2002-09-30', '2002-09-30', 'CDT', 20, NULL, NULL),
bank_management-# (21, 100, '2003-07-30', '2003-07-30', 'CDT', 22, NULL, NULL);
INSERT 0 21

```

6. The questions below require you to write and execute SQL statements

a. Increase a value of 28,964 by 18.5%

```
SELECT 28964 * 1.185 AS Increased_Value;
```



```
bank_management=# SELECT 28964 * 1.185 AS Increased_Value;
increased_value
-----
      34322.340
(1 row)
```

b. List the first and last name of all Employees

SELECT FIRST_NAME, LAST_NAME FROM EMPLOYEE;

```
bank_management=# SELECT FIRST_NAME, LAST_NAME FROM EMPLOYEE;
first_name | last_name
-----+-----
Michael    | Smith
Susan      | Barker
Robert     | Tyler
Susan      | Hawthorne
John       | Gooding
Helen      | Fleming
Chris      | Tucker
Sarah      | Parker
Jane       | Grossman
Paula      | Roberts
Thomas     | Ziegler
Samantha   | Jameson
John       | Blake
Cindy      | Mason
Frank      | Portman
Theresa     | Markham
Beth       | Fowler
Rick       | Tulman
(18 rows)
```

c. List the types currently offered by the bank (no duplicates)

SELECT DISTINCT product_type_cd FROM PRODUCT;

```
bank_management=# SELECT DISTINCT product_type_cd FROM PRODUCT;
product_type_cd
-----
ACCOUNT
LOAN
(2 rows)
```

d. List the different types of Loan that are offered

```
SELECT DISTINCT PRODUCT_TYPE_CD FROM PRODUCT WHERE  
PRODUCT_TYPE_CD = 'Loan';
```

```
bank_management=# SELECT DISTINCT PRODUCT_TYPE_CD FROM PRODUCT WHERE PRODUCT_TYPE_CD = 'Loan';  
product_type_cd  
-----  
(0 rows)
```

e. List all employees whose first name starts with the letter 'S'

```
SELECT FIRST_NAME, LAST_NAME FROM EMPLOYEE WHERE FIRST_NAME LIKE  
'S%';
```

```
bank_management=# SELECT FIRST_NAME, LAST_NAME FROM EMPLOYEE WHERE FIRST_NAME LIKE 'S%';  
first_name | last_name  
-----+-----  
Susan      | Barker  
Susan      | Hawthorne  
Sarah      | Parker  
Samantha   | Jameson  
(4 rows)
```

f. List employees whose first name starts with 'S' or 'T' and work in the 'Operations' department

```
SELECT FIRST_NAME, LAST_NAME FROM EMPLOYEE  
  
WHERE (FIRST_NAME LIKE 'S%' OR FIRST_NAME LIKE 'T%')  
  
AND DEPT_ID = (SELECT DEPT_ID FROM DEPARTMENT WHERE NAME =  
'Operations');
```

```
bank_management=# SELECT FIRST_NAME, LAST_NAME FROM EMPLOYEE
bank_management=# WHERE (FIRST_NAME LIKE 'S%' OR FIRST_NAME LIKE 'T%')
bank_management=# AND DEPT_ID = (SELECT DEPT_ID FROM DEPARTMENT WHERE NAME = 'Operations');
 first_name | last_name
-----+-----
 Susan      | Hawthorne
 Sarah       | Parker
 Thomas     | Ziegler
 Samantha   | Jameson
 Theresa    | Markham
(5 rows)
```

g. Find employees whose first name is either Susan, Helen or Paula

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME FROM EMPLOYEE
```

```
WHERE FIRST_NAME IN ('Susan', 'Helen', 'Paula');
```

```
bank_management=# SELECT EMP_ID, FIRST_NAME, LAST_NAME FROM EMPLOYEE
bank_management=# WHERE FIRST_NAME IN ('Susan', 'Helen', 'Paula');
 emp_id | first_name | last_name
-----+-----+-----
      2 | Susan      | Barker
      4 | Susan      | Hawthorne
      6 | Helen      | Fleming
     10 | Paula      | Roberts
(4 rows)
```

h. Find employees with a start date after 1st January 2001 and before 31st December 2002

```
SELECT * FROM EMPLOYEE
```

```
WHERE START_DATE > '2001-01-01' AND START_DATE < '2002-12-31';
```

```
bank_management=# SELECT * FROM EMPLOYEE
bank_management=# WHERE START_DATE > '2001-01-01' AND START_DATE < '2002-12-31';
 emp_id | end_date | first_name | last_name | start_date | title | assigned_branch_id | dept_id | superior_emp_id
-----+-----+-----+-----+-----+-----+-----+-----+-----
      1 |          | Michael    | Smith     | 2001-06-22 | President | 1 | 3 | 
      2 |          | Susan      | Barker    | 2002-09-12 | Vice President | 1 | 3 | 1
      3 |          | Robert     | Tyler     | 2002-02-09 | Treasurer | 1 | 3 | 1
      8 |          | Sarah      | Parker    | 2002-12-02 | Teller    | 1 | 1 | 6
      9 |          | Jane       | Grossman  | 2002-05-03 | Teller    | 1 | 1 | 6
     10 |          | Paula      | Roberts   | 2002-07-27 | Head Teller | 2 | 1 | 4
     14 |          | Cindy      | Mason     | 2002-08-09 | Teller    | 3 | 1 | 13
     16 |          | Theresa    | Markham   | 2001-03-15 | Head Teller | 4 | 1 | 4
     17 |          | Beth       | Fowler    | 2002-06-29 | Teller    | 4 | 1 | 16
     18 |          | Rick       | Tulman    | 2002-12-12 | Teller    | 4 | 1 | 16
(10 rows)
```

i. List customers with FED_ID formatted as nnn-nn-nnnn

```
SELECT * FROM CUSTOMER
```

```
WHERE FED_ID ~ '^\\d{3}-\\d{2}-\\d{4}$';
```

```
bank_management=# SELECT * FROM CUSTOMER
bank_management=# WHERE FED_ID ~ '^\\d{3}-\\d{2}-\\d{4}$';
```

cust_id	address	city	cust_type_cd	fed_id	postal_code	state
1	47 Mockingbird Ln	Lynnfield	I	111-11-1111	1940	MA
2	372 Clearwater Blvd	Woburn	I	222-22-2222	1801	MA
3	18 Jessup Rd	Quincy	I	333-33-3333	2169	MA
4	12 Buchanan Ln	Waltham	I	444-44-4444	2451	MA
5	2341 Main St	Salem	I	555-55-5555	3079	NH
6	12 Blaylock Ln	Waltham	I	666-66-6666	2451	MA
7	29 Admiral Ln	Wilmington	I	777-77-7777	1887	MA
8	472 Freedom Rd	Salem	I	888-88-8888	3079	NH
9	29 Maple St	Newton	I	999-99-9999	2458	MA

(9 rows)

j. List all product types and names in specified order

```
SELECT PRODUCT_TYPE_CD, NAME FROM PRODUCT
```

```
ORDER BY PRODUCT_TYPE_CD ASC, NAME DESC;
```

```
bank_management=# SELECT PRODUCT_TYPE_CD, NAME FROM PRODUCT
bank_management=# ORDER BY PRODUCT_TYPE_CD ASC, NAME DESC;
```

product_type_cd	name
ACCOUNT	savings account
ACCOUNT	money market account
ACCOUNT	checking account
ACCOUNT	certificate of deposit
LOAN	small business loan
LOAN	home mortgage
LOAN	business line of credit
LOAN	auto loan

(8 rows)

k. List all employees whose position is 'Teller', sorted by start date

```
SELECT * FROM EMPLOYEE
```

WHERE TITLE = 'Teller'

ORDER BY START_DATE;

```
bank_management=# SELECT * FROM EMPLOYEE
bank_management=# WHERE TITLE = 'Teller'
bank_management=# ORDER BY START_DATE;
```

emp_id	end_date	first_name	last_name	start_date	title	assigned_branch_id	dept_id	superior_emp_id
11		Thomas	Ziegler	2000-10-23	Teller	2	1	10
9		Jane	Grossman	2002-05-03	Teller	1	1	6
17		Beth	Fowler	2002-06-29	Teller	4	1	16
14		Cindy	Mason	2002-08-09	Teller	3	1	13
8		Sarah	Parker	2002-12-02	Teller	1	1	6
18		Rick	Tulman	2002-12-12	Teller	4	1	16
12		Samantha	Jameson	2003-01-08	Teller	2	1	10
15		Frank	Portman	2003-04-01	Teller	3	1	13
7		Chris	Tucker	2004-09-15	Teller	1	1	6

(9 rows)

I. Select account details for cust_ID=1 and increase all balances by
2%

UPDATE ACCOUNT

SET AVAIL_BALANCE = AVAIL_BALANCE * 1.02,

PENDING_BALANCE = PENDING_BALANCE * 1.02

WHERE CUST_ID = 1;

```
bank_management=# UPDATE ACCOUNT
bank_management=# SET AVAIL_BALANCE = AVAIL_BALANCE * 1.02,
bank_management=#     PENDING_BALANCE = PENDING_BALANCE * 1.02
bank_management=# WHERE CUST_ID = 1;
UPDATE 3
bank_management=#
```

SELECT ACCOUNT_ID, PRODUCT_CD, AVAIL_BALANCE, PENDING_BALANCE

FROM ACCOUNT

WHERE CUST_ID = 1;

```
bank_management=# SELECT ACCOUNT_ID, PRODUCT_CD, AVAIL_BALANCE, PENDING_BALANCE
bank_management=# FROM ACCOUNT
bank_management=# WHERE CUST_ID = 1;
 account_id | product_cd | avail_balance | pending_balance
-----
          1 | CHK       |      1078.91 |      1078.91
          2 | SAV       |       510.00 |       510.00
          3 | CD        |      3060.00 |      3060.00
(3 rows)
```

m. Remove transactions made on 30th July 2003

DELETE FROM ACC_TRANSACTION

WHERE TXN_DATE = '2003-07-30';

```
bank_management=# DELETE FROM ACC_TRANSACTION WHERE TXN_DATE = '2003-07-30';
DELETE 2
bank_management=#
```

SELECT COUNT(*) FROM ACC_TRANSACTION

WHERE TXN_DATE = '2003-07-30';

```
bank_management=# SELECT COUNT(*) FROM ACC_TRANSACTION
bank_management=# WHERE TXN_DATE = '2003-07-30';
 count
-----
      0
(1 row)
```

n. List accounts with an available balance bigger than £10,000

SELECT ACCOUNT_ID, CUST_ID, AVAIL_BALANCE FROM ACCOUNT

WHERE AVAIL_BALANCE > 10000

ORDER BY AVAIL_BALANCE DESC;

```
bank_management=# SELECT ACCOUNT_ID, CUST_ID, AVAIL_BALANCE FROM ACCOUNT
bank_management=# WHERE AVAIL_BALANCE > 10000
bank_management=# ORDER BY AVAIL_BALANCE DESC;
 account_id | cust_id | avail_balance
-----+-----+-----
          23 |       13 |      50000.00
          22 |       12 |      38552.05
          20 |       10 |      23575.12
(3 rows)
```

o. List all cities in the state of “NH” without duplicates, sorted alphabetically

SELECT DISTINCT CITY FROM BRANCH

WHERE STATE = 'NH'

ORDER BY CITY;

```
bank_management=# SELECT DISTINCT CITY FROM BRANCH
bank_management=# WHERE STATE = 'NH'
bank_management=# ORDER BY CITY;
 city
-----
 Salem
(1 row)
```

p. Update Susan Tingley's surname to 'Brown'

UPDATE INDIVIDUAL

SET LAST_NAME = 'Brown'

WHERE FIRST_NAME = 'Susan' AND LAST_NAME = 'Tingley';

```
bank_management=# UPDATE INDIVIDUAL
bank_management=# SET LAST_NAME = 'Brown'
bank_management=# WHERE FIRST_NAME = 'Susan' AND LAST_NAME = 'Tingley';
UPDATE 1
bank_management=#
```

q. List customers born before 1965

```
SELECT * FROM INDIVIDUAL
```

```
WHERE BIRTH_DATE < '1965-01-01';
```

```
bank_management=# SELECT * FROM INDIVIDUAL
bank_management=# WHERE BIRTH_DATE < '1965-01-01';
 cust_id | birth_date | first_name | last_name
-----+-----+-----+-----
      3 | 1958-02-06 | Frank     | Tucker
      6 | 1962-09-14 | John      | Spencer
      7 | 1947-03-19 | Margaret  | Young
(3 rows)
```

r. Update Thomas Ziegler's employee record (left job on 1st November 2019)

```
UPDATE EMPLOYEE
```

```
SET END_DATE = '2019-11-01'
```

```
WHERE FIRST_NAME = 'Thomas' AND LAST_NAME = 'Ziegler';
```

```
bank_management=# UPDATE EMPLOYEE
bank_management=# SET END_DATE = '2019-11-01'
bank_management=# WHERE FIRST_NAME = 'Thomas' AND LAST_NAME = 'Ziegler';
UPDATE 1
bank_management=#
```

s. List customers with a product code of 'SAV' sorted by decreasing available balance


```
SELECT * FROM ACCOUNT

WHERE PRODUCT_CD = 'SAV'

ORDER BY AVAIL_BALANCE DESC;
```

```
bank_management=# SELECT * FROM ACCOUNT
bank_management=# WHERE PRODUCT_CD = 'SAV'
bank_management=# ORDER BY AVAIL_BALANCE DESC;
```

account_id	avail_balance	close_date	last_activity_date	open_date	pending_balance	status	cust_id	open_branch_id	open_emp_id	product_cd
9	767.77		2004-10-24	2000-01-15	767.77	ACTIVE	4	1	8	SAV
2	510.00		2004-12-19	2000-01-15	510.00	ACTIVE	1	2	10	SAV
16	387.99		2004-10-12	2001-05-23	387.99	ACTIVE	3	4	13	SAV
5	200.00		2004-12-11	2001-03-12	200.00	ACTIVE	2	2	11	SAV

(4 rows)