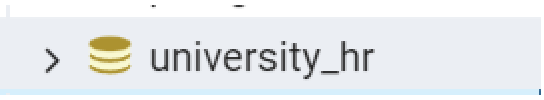
Databases-Week04

-Tao Nijia, He Yiyang, Liang Ruyi

BASIC TASKS

Task 1.

1. Consider you are creating a simple HR system for a university. Use DDL component of SQL and create a
new Database in your chosen DBMS (in class I demonstrated PosgreSQL with pgAdmin 4).
Mycode:
CREATE DATABASE university_hr;
My result



university_hr

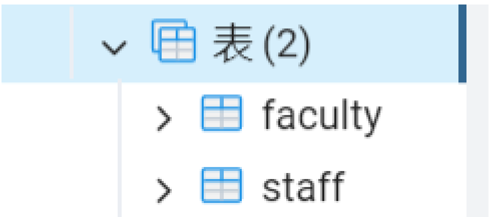
2. The database schema of that system are shown below (Faculty and Staff). Use the data types introduced
in the lecture and create the tables for the Staff and Faculty relations. Add constraints to ensure that the
facultyName is not NULL. Using appropriate data types create the tables for both of these relation. Make
sure that you create the PRIMARY and FOREIGN keys:
a. STAFF (staffId, staffName, staffDOB, *staffFaculty*)
b. FACULTY (facultyId, facultyName, NoOfStaff)

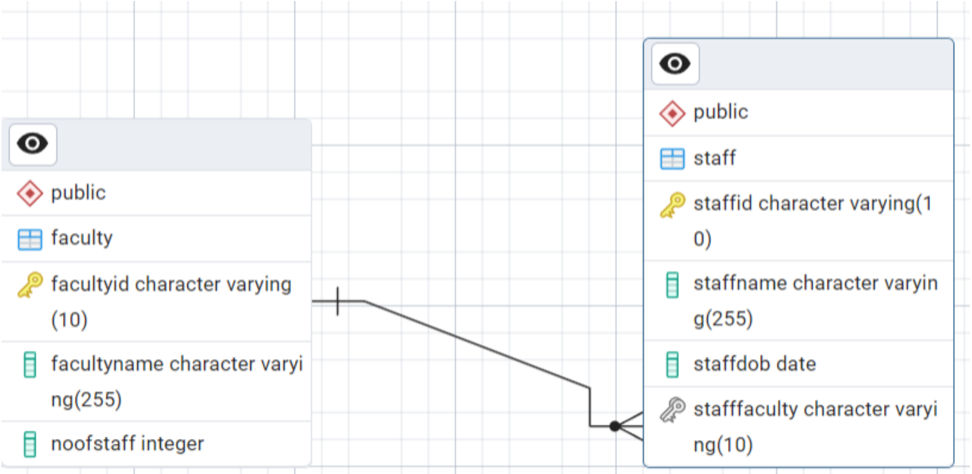
mycode:

```
CREATE TABLE Faculty (
    facultyId SERIAL PRIMARY KEY,
    facultyName VARCHAR(100) NOT NULL,
    NoOfStaff INT
);

CREATE TABLE Staff (
    staffId SERIAL PRIMARY KEY,
    staffName VARCHAR(100) NOT NULL,
    staffDOB DATE,
    staffFaculty INT,
    FOREIGN KEY (staffFaculty) REFERENCES Faculty(facultyId)
);
```

myresult:




public
faculty
facultyid character varying(10)
facultyname character varying(255)
noofstaff integer
public
staff
staffid character varying(10)
staffname character varying(255)
staffdob date
stafffaculty character varying(10)

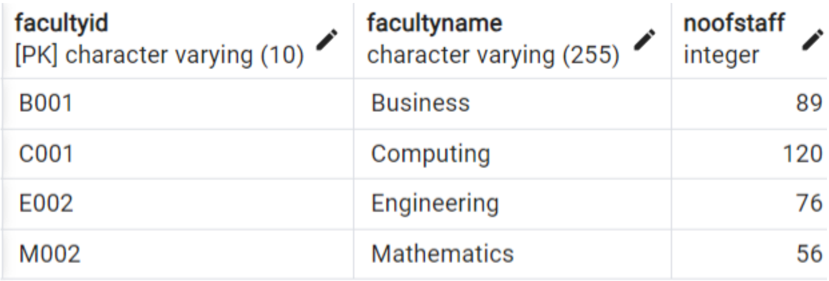
3. Insert the following data into the tables you have just created. Use Data Manipulation Language (DML)
component of SQL command to populate the tables created. Think about which data has to be populated
first and why

My code：

```
INSERT INTO Faculty (facultyName, NoOfStaff) VALUES
(’Engineering’, 70),
(’Arts’, 50),
(’Science’, 60);

INSERT INTO Staff (staffName, staffDOB, staffFaculty) VALUES
(’Alison Green’, ’1985-05-10’, 1),
(’Kieran West’, ’1978-03-22’, 2),
(’John Doe’, ’1982-07-15’, 1);
SELECT * FROM public.faculty
ORDER BY facultyid ASC
```

My result：



facultyid [PK] character varying (10)	facultyname character varying (255)	noofstaff integer
B001	Business	89
C001	Computing	120
E002	Engineering	76
M002	Mathematics	56

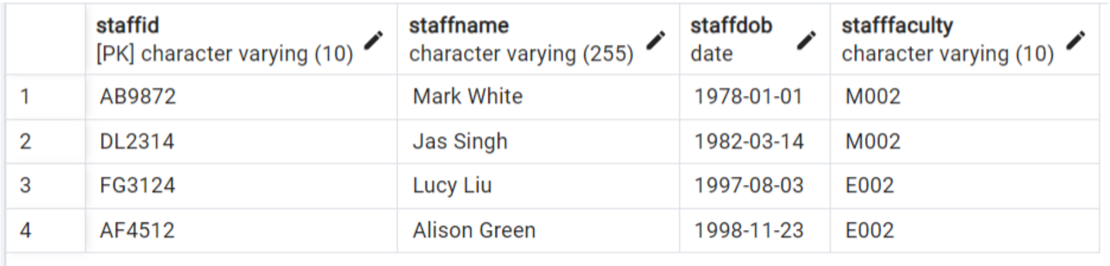
4. Code appropriate SQL command and find out the following

a. Obtain all data from the STAFF table

mycode:

```
SELECT * FROM Staff;
```

Myresult:



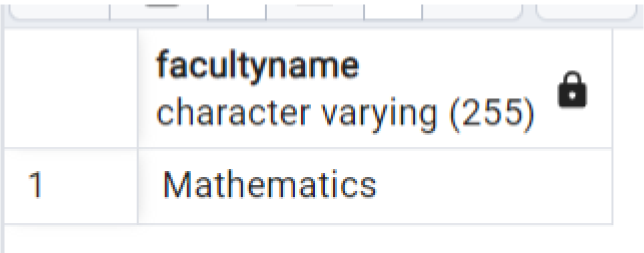
	staffid [PK] character varying (10)	staffname character varying (255)	staffdob date	stafffaculty character varying (10)
1	AB9872	Mark White	1978-01-01	M002
2	DL2314	Jas Singh	1982-03-14	M002
3	FG3124	Lucy Liu	1997-08-03	E002
4	AF4512	Alison Green	1998-11-23	E002

b. Find the facultyName where the number of staff is less than 75

mycode:

```
SELECT facultyName FROM Faculty WHERE NoOfStaff < 75;
```

Myresult:



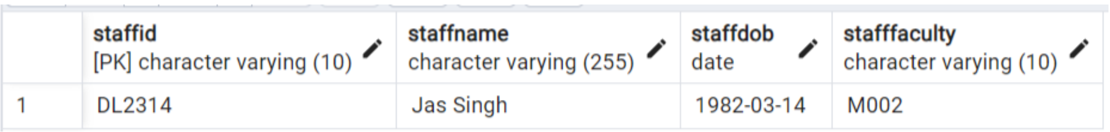
	facultyname character varying (255)
1	Mathematics

c. List all staff that were born in then 1980’ s

mycode:

```
SELECT * FROM Staff WHERE staffDOB BETWEEN ’1980-01-01’ AND ’1989-12-31’;
```

myresult:



	staffid [PK] character varying (10)	staffname character varying (255)	staffdob date	stafffaculty character varying (10)
1	DL2314	Jas Singh	1982-03-14	M002

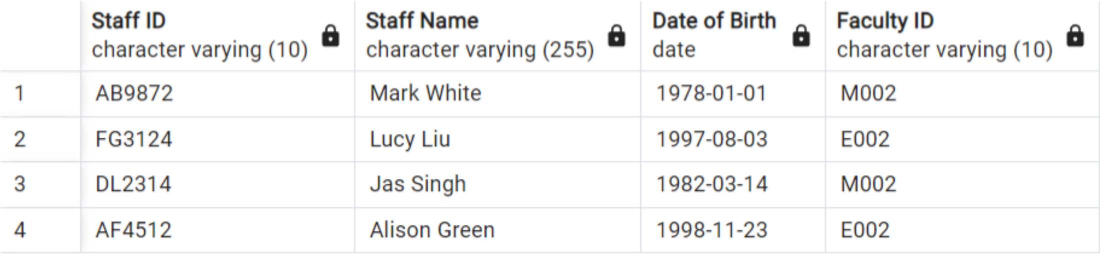
d. List all columns from the staff table, sort by their name (Z-A).

mycode:

```
 SELECT staffId AS ″Staff id″, staffName AS ″Staffname″, staffdob AS ″Date of Birth″, staffFaculty AS ″Faculty ID″
```

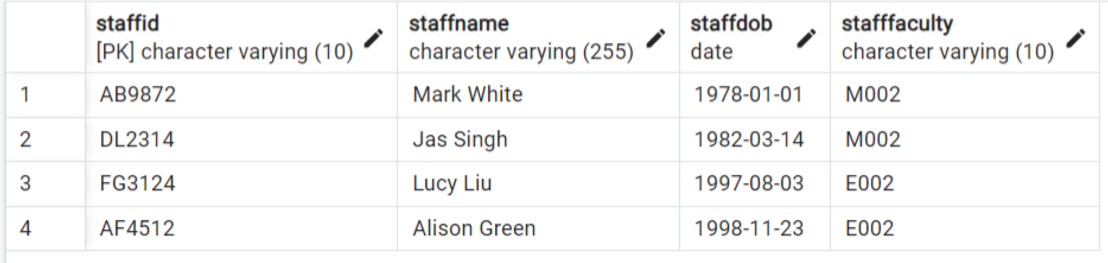
```
FROM Staff
ORDER BY staffName DESC;
```

myresult:



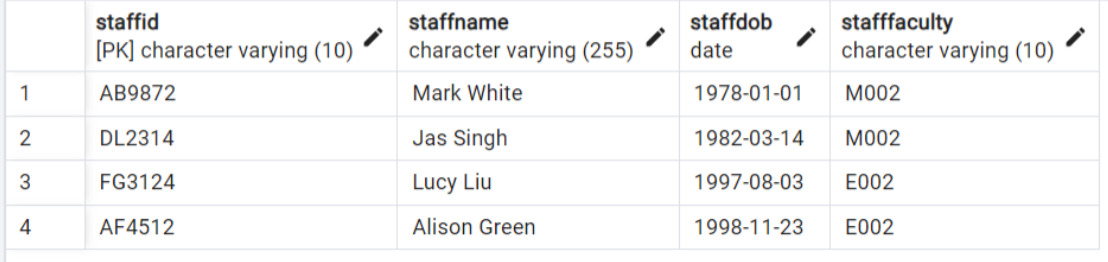
	Staff ID character varying (10)	Staff Name character varying (255)	Date of Birth date	Faculty ID character varying (10)
1	AB9872	Mark White	1978-01-01	M002
2	FG3124	Lucy Liu	1997-08-03	E002
3	DL2314	Jas Singh	1982-03-14	M002
4	AF4512	Alison Green	1998-11-23	E002

e. Alison Green has moved to a different faculty. She now works in Engineering. Write an
appropriate SQL statement to reflect this
mycode:
UPDATE Staff SET stafffaculty = ‘AF4512’ WHERE staffname = ’Alison Green’;
Myresult:



	staffid [PK] character varying (10)	staffname character varying (255)	staffdob date	stafffaculty character varying (10)
1	AB9872	Mark White	1978-01-01	M002
2	DL2314	Jas Singh	1982-03-14	M002
3	FG3124	Lucy Liu	1997-08-03	E002
4	AF4512	Alison Green	1998-11-23	E002

f. Kieran West has left the University. Write an appropriate SQL statement to remove his record
from the database
mycode:
DELETE FROM Staff WHERE staffName = ’Kieran West’;
Myresult:



	staffid [PK] character varying (10)	staffname character varying (255)	staffdob date	stafffaculty character varying (10)
1	AB9872	Mark White	1978-01-01	M002
2	DL2314	Jas Singh	1982-03-14	M002
3	FG3124	Lucy Liu	1997-08-03	E002
4	AF4512	Alison Green	1998-11-23	E002

Medium tasks(Liang Ruyi)
5. Consider you are creating a small database simulating a friendly bank. Use DDL component of SQL and
create a new Database in your chosen DBMS (in class I demonstrated PosgreSQL with pgAdmin 4). An

overview of the tables (adopted from Oracle) in the database are shown below:

mycode

```
CREATE DATABASE friendly_bank;
```

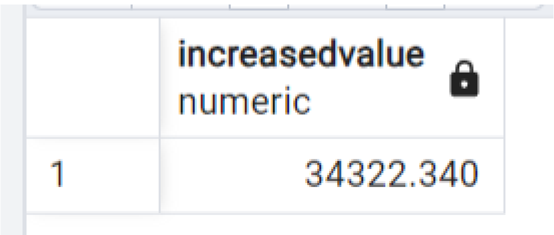
6. The questions below require you to write and execute SQL statements

a. Write a suitable query to display the result of increasing a value of 28,964 by 18.5%

mycode:

```
SELECT 28964 * 1.185 AS IncreasedValue;
```

myresult:



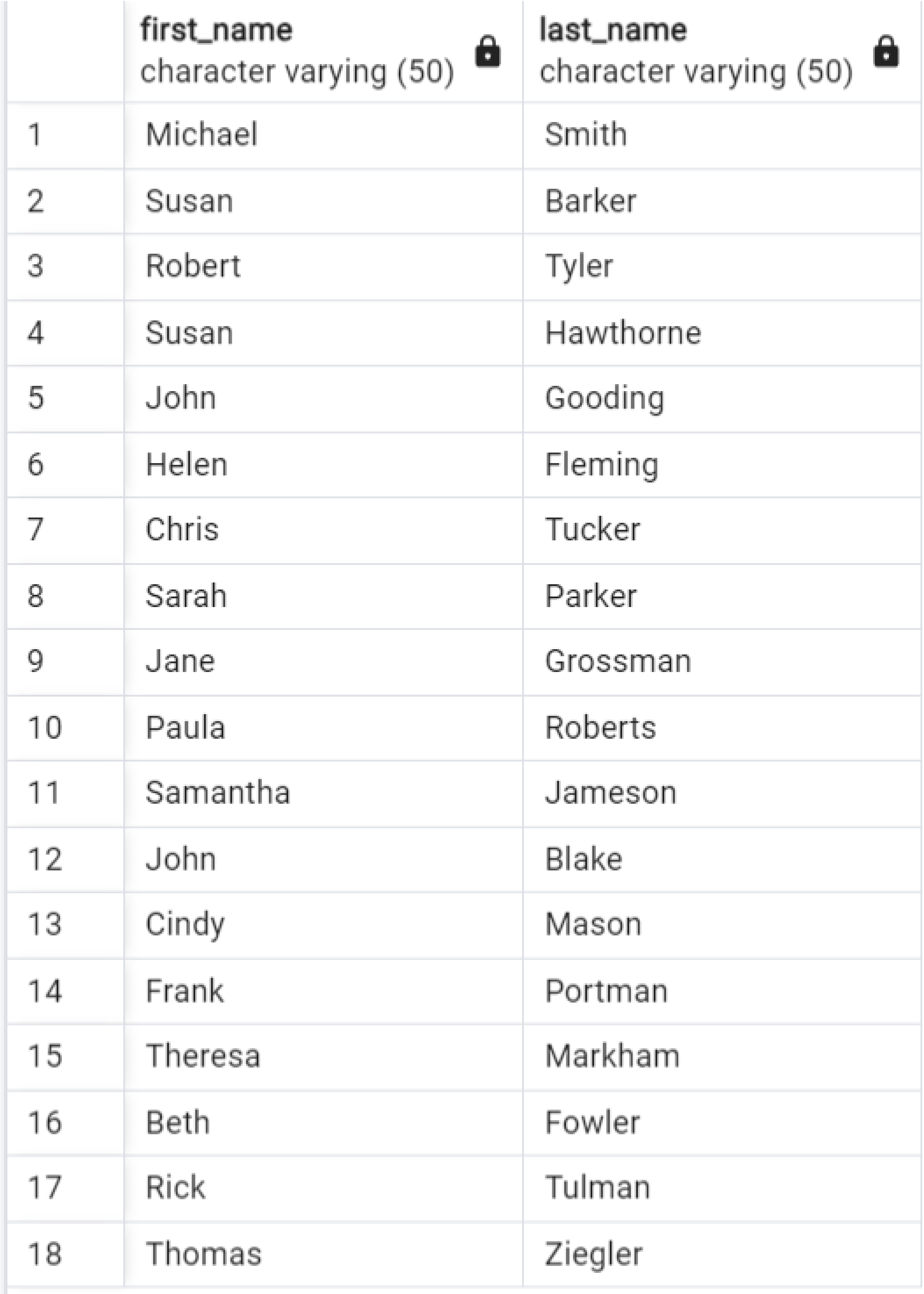
	increasedvalue numeric 🔒
1	34322.340

b. List the first and last name of all Employees

mycode:

```
SELECT first_name, last_name FROM employee;
```

Myresult:



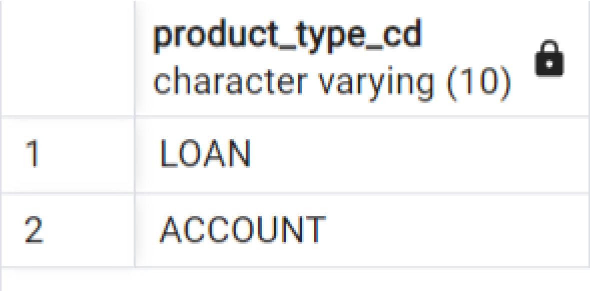
	first_name character varying (50) 🔒	last_name character varying (50) 🔒
1	Michael	Smith
2	Susan	Barker
3	Robert	Tyler
4	Susan	Hawthorne
5	John	Gooding
6	Helen	Fleming
7	Chris	Tucker
8	Sarah	Parker
9	Jane	Grossman
10	Paula	Roberts
11	Samantha	Jameson
12	John	Blake
13	Cindy	Mason
14	Frank	Portman
15	Theresa	Markham
16	Beth	Fowler
17	Rick	Tulman
18	Thomas	Ziegler

c. The PRODUCTS table lists products currently offered by the bank. There can be three types;
Account, Loan or Insurance. List the types currently offered by the bank. Your results should not
show duplicate records

mycode:

```
SELECT DISTINCT product_type_cd FROM PRODUCT;
```

myresult:



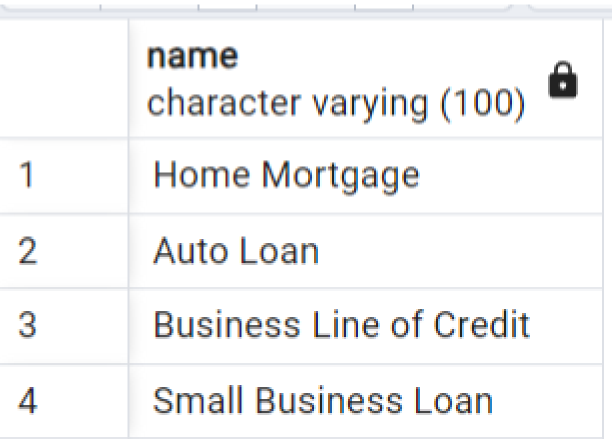
	product_type_cd character varying (10)
1	LOAN
2	ACCOUNT

d. List the different types of Loan that are offered

mycode:

```
SELECT name
FROM product
WHERE product_type_cd='LOAN';
```

myresult:



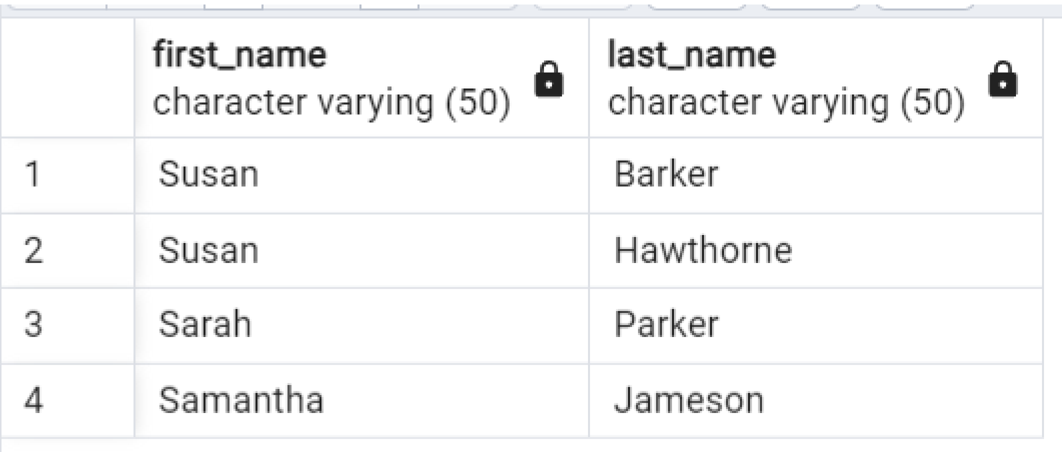
	name character varying (100)
1	Home Mortgage
2	Auto Loan
3	Business Line of Credit
4	Small Business Loan

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

mycode:

```
SELECT first_name,last_name
From employee
where first_name LIke 'S%';
```

myresult:



	first_name character varying (50)	last_name character varying (50)
1	Susan	Barker
2	Susan	Hawthorne
3	Sarah	Parker
4	Samantha	Jameson

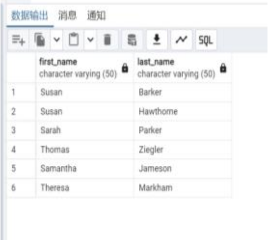
f. List all employees whose first name starts with the letter ‘S’ or

‘T’ and that also works in the

mycode:

```
 SELECT * FROM Employee WHERE (first_name LIKE ’S%’ OR first_name LIKE
’T%’) AND department = ’Operations’;
```

Myresult:



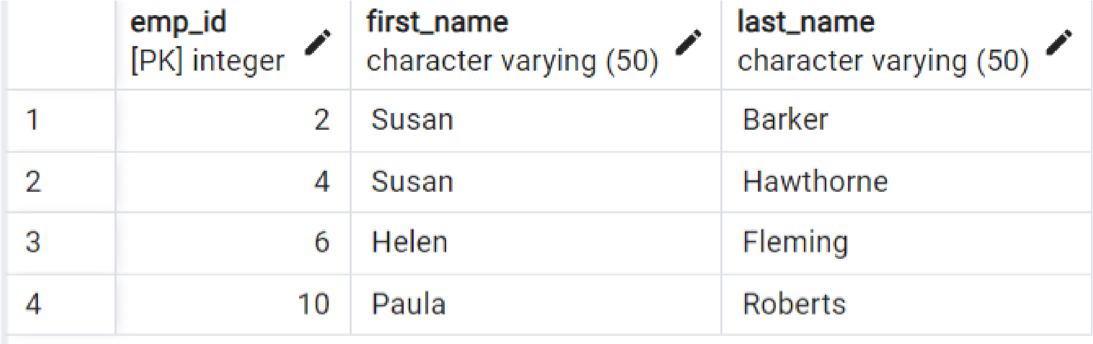
	first_name character varying (50)	last_name character varying (50)
1	Susan	Barker
2	Susan	Hawthorne
3	Sarah	Parker
4	Thomas	Ziegler
5	Samantha	Jameson
6	Theresa	Markham

g. Find all employees whose first name is either Susan, Helen or Paula. Your results should show
their employee ID, first name and surname

mycode:

```
SELECT emp_id, first_name, last_name FROM Employee WHERE first_name IN
(’Susan’, ’Helen’, ’Paula’);
```

myresult:



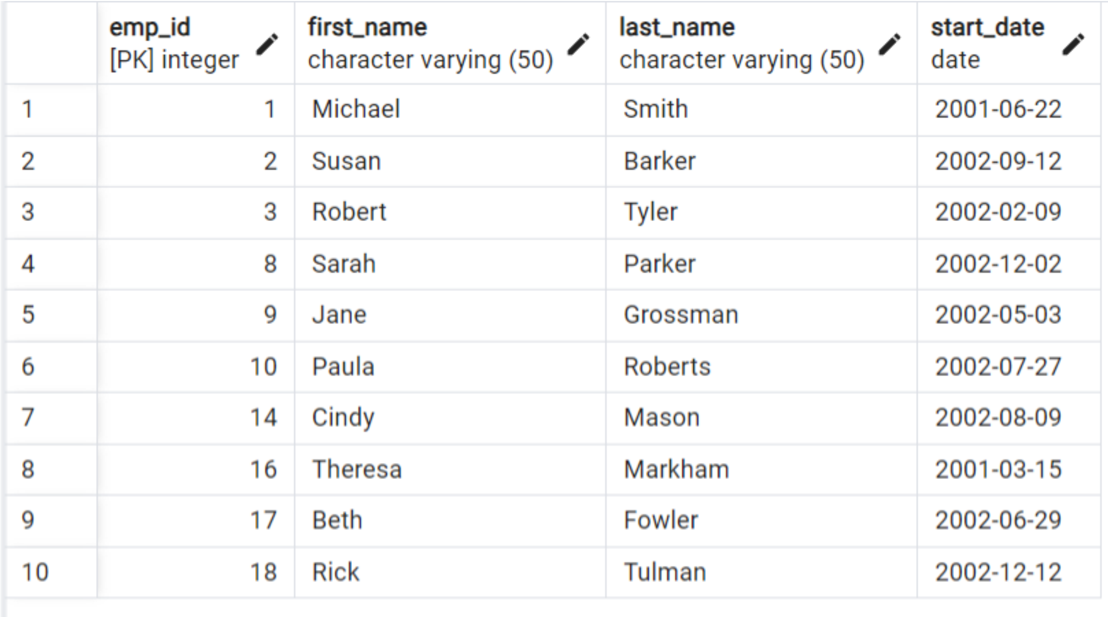
	emp_id [PK] integer	first_name character varying (50)	last_name character varying (50)
1	2	Susan	Barker
2	4	Susan	Hawthorne
3	6	Helen	Fleming
4	10	Paula	Roberts

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

mycode:

```
SELECT emp_id,first_name,last_name,start_date
From employee
where start_date BETWEEN ’2001-01-01’ AND ’2002-12-31’
```

myresult:



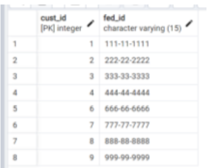
	emp_id [PK] integer	first_name character varying (50)	last_name character varying (50)	start_date date
1	1	Michael	Smith	2001-06-22
2	2	Susan	Barker	2002-09-12
3	3	Robert	Tyler	2002-02-09
4	8	Sarah	Parker	2002-12-02
5	9	Jane	Grossman	2002-05-03
6	10	Paula	Roberts	2002-07-27
7	14	Cindy	Mason	2002-08-09
8	16	Theresa	Markham	2001-03-15
9	17	Beth	Fowler	2002-06-29
10	18	Rick	Tulman	2002-12-12

i. The FED_ID column in the CUSTOMER table can be formatted in two ways (where n is a
number): a. nnn-nn-nnnn b. nn-nnnnnn List all customers that have a fed_id formatted the same
as shown in (a) above

mycode:

```
SELECT cust_id,fed_id
FROM customer
WHERE fed_id LIKE’___-__-____’;
```

myresult:



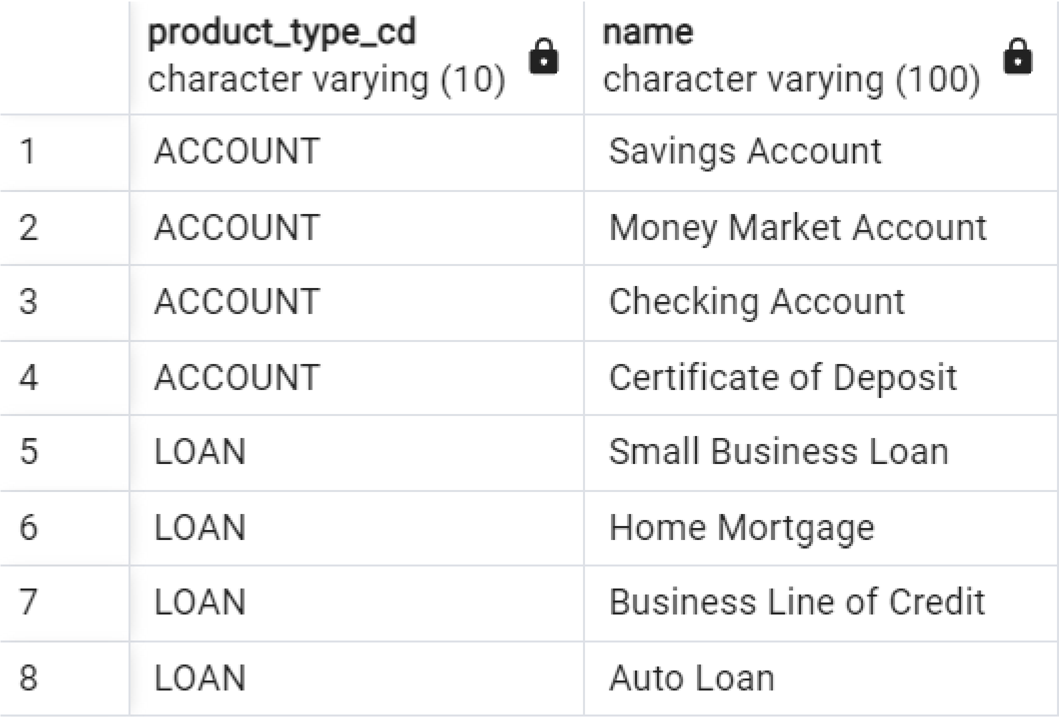
	cust_id [PK] integer	fed_id character varying (15)
1	1	111-11-1111
2	2	222-22-2222
3	3	333-33-3333
4	4	444-44-4444
5	6	666-66-6666
6	7	777-77-7777
7	8	888-88-8888
8	9	999-99-9999

j. Using the PRODUCTS table, list all product types and their names. The product type should be in
ascending alphabetical order, and the names in descending alphabetical order

mycode:

```
SELECT product_type_cd,name
FROM product
ORDER BY product_type_cd ASC,name DESC;
```

myresult:



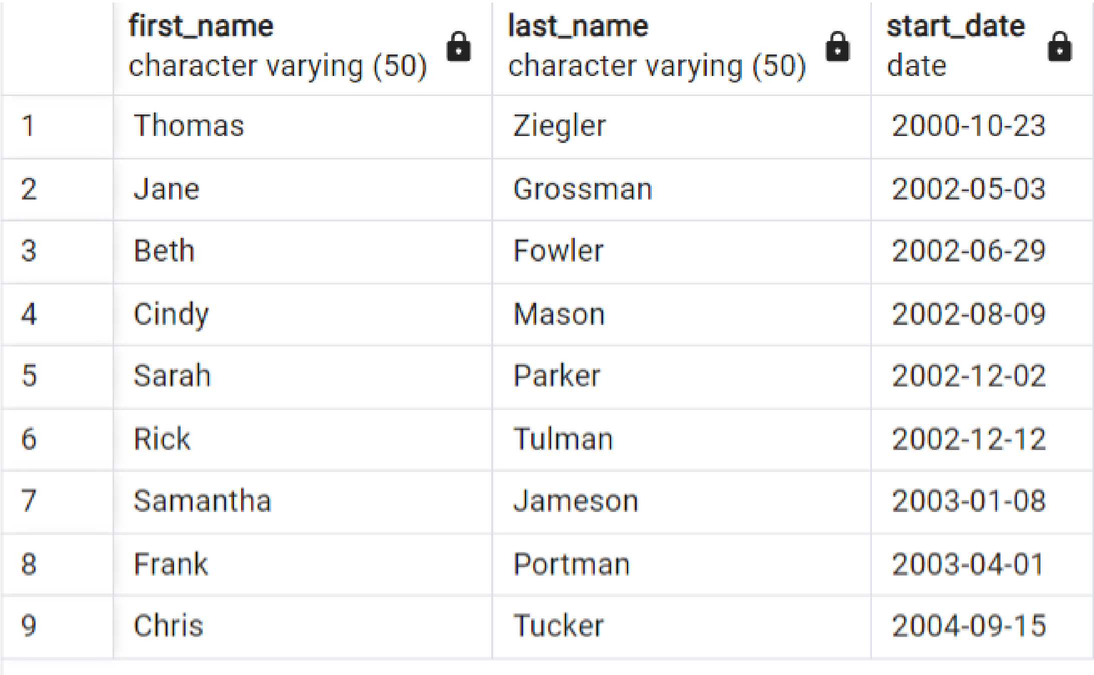
	product_type_cd character varying (10) 🔒	name character varying (100) 🔒
1	ACCOUNT	Savings Account
2	ACCOUNT	Money Market Account
3	ACCOUNT	Checking Account
4	ACCOUNT	Certificate of Deposit
5	LOAN	Small Business Loan
6	LOAN	Home Mortgage
7	LOAN	Business Line of Credit
8	LOAN	Auto Loan

k. List all employees whose position is a ‘Teller’. Sort them based on their start date, displaying the longest employed person first

mycode:

```
SELECT first_name,last_name,start_date
FROM employee
Where title='TELLER'
ORDER BY start_date;
```

myresult:

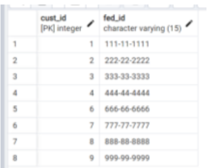


	first_name character varying (50) 🔒	last_name character varying (50) 🔒	start_date date 🔒
1	Thomas	Ziegler	2000-10-23
2	Jane	Grossman	2002-05-03
3	Beth	Fowler	2002-06-29
4	Cindy	Mason	2002-08-09
5	Sarah	Parker	2002-12-02
6	Rick	Tulman	2002-12-12
7	Samantha	Jameson	2003-01-08
8	Frank	Portman	2003-04-01
9	Chris	Tucker	2004-09-15

l. Select the account ID, product code, available and pending balances from the ACCOUNTS table for cust_ID=1. Write a statement to increase all balances in this customers account by 2%.
Display the updated data, listing the fields as before
mycode:

```
SELECT account_id, product_cd, avail_balance, pending_balance
FROM account
WHERE cust_id = 1;
UPDATE account
SET avail_balance = avail_balance * 1.02,
    pending_balance = pending_balance * 1.02
WHERE cust_id = 1;
SELECT account_id, product_cd, avail_balance, pending_balance
FROM account
WHERE cust_id = 1;
```

myresult:

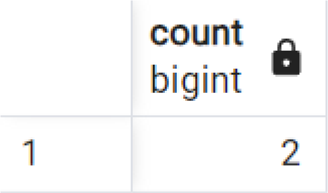


	cust_id [PK] integer	fed_id character varying (15)
1	1	111-11-1111
2	2	222-22-2222
3	3	333-33-3333
4	4	444-44-4444
5	6	666-66-6666
6	7	777-77-7777
7	8	888-88-8888
8	9	999-99-9999

m. There has been an error and all transactions that were made on the 30th July 2003 need to be removed. Write a suitable statement to remove the relevant records from the ACCOUNT_TRANSACTIONS table. How many records were affected?
Mycode:

```
DELETE FROM acc_transaction
WHERE txn_date = '2003-07-30';
SELECT COUNT(*)
FROM acc_transaction
WHERE txn_date = '2003-07-30';
```

Myresult:

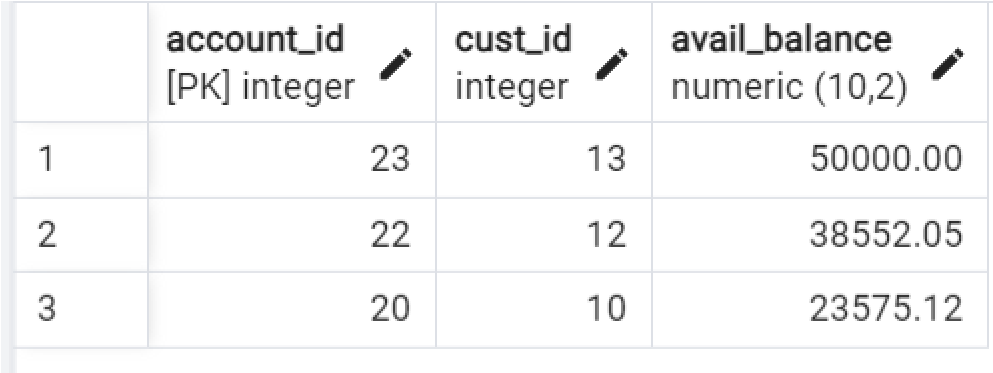


	count bigint
1	2

n. List the account number, customer number and available balance for all accounts with an available balance bigger than £10,000. Display the records in decreasing balance order mycode:

```
SELECT account_id, cust_id, avail_balance
FROM account
WHERE avail_balance > 10000
ORDER BY avail_balance DESC;
```

myresult:



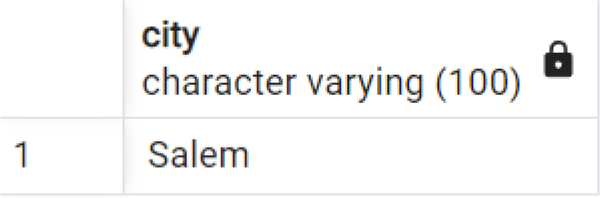
	account_id [PK] integer	cust_id integer	avail_balance numeric (10,2)
1	23	13	50000.00
2	22	12	38552.05
3	20	10	23575.12

o. List all city’ s based in the state of “NH” . The results should not contain multiple rows and should be sorted alphabetically

mycode:

```
SELECT DISTINCT city FROM branch WHERE state=’NH’ ORDER BY city;
```

myresult:



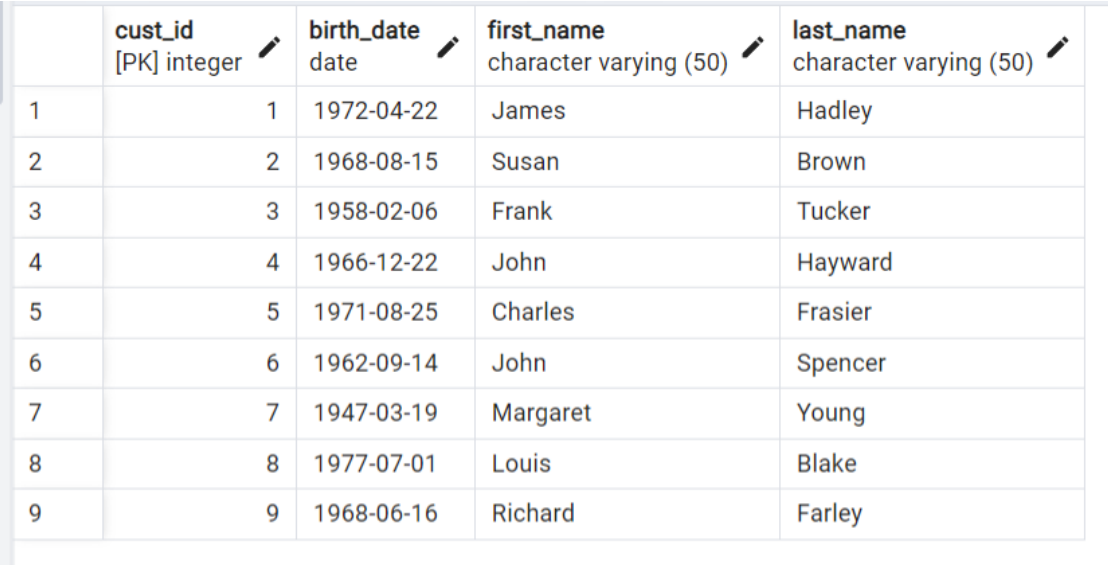
	city character varying (100)
1	Salem

p. The customer Susan Tingley has notified the bank of a name change. Her surname is now ‘Brown’ . Identify which table and column need updating and write the appropriate statement to amend the data

mycode:

```
SELECT * FROM public.individual ORDER BY cust_id ASC;
```

myresult:



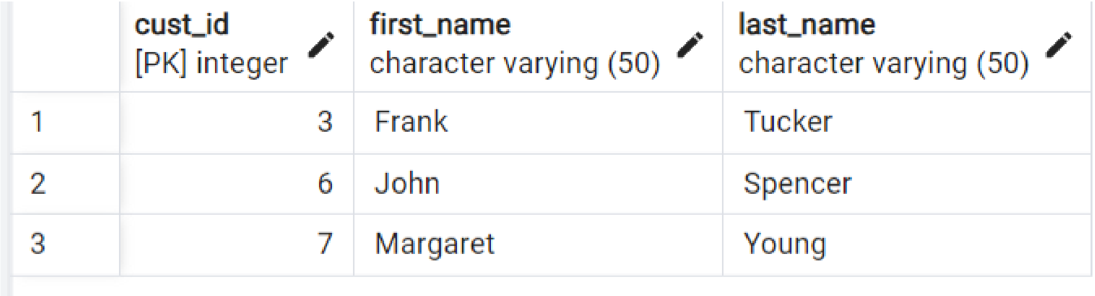
	cust_id [PK] integer	birth_date date	first_name character varying (50)	last_name character varying (50)
1	1	1972-04-22	James	Hadley
2	2	1968-08-15	Susan	Brown
3	3	1958-02-06	Frank	Tucker
4	4	1966-12-22	John	Hayward
5	5	1971-08-25	Charles	Frasier
6	6	1962-09-14	John	Spencer
7	7	1947-03-19	Margaret	Young
8	8	1977-07-01	Louis	Blake
9	9	1968-06-16	Richard	Farley

q. List all customers that were born before 1965 、

mycode：

```
SELECT cust_id， first_name， last_name
FROM individual
WHERE birth_date<’1965-01-01’;
```

myresult:



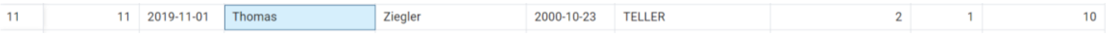
	cust_id [PK] integer	first_name character varying (50)	last_name character varying (50)
1	3	Frank	Tucker
2	6	John	Spencer
3	7	Margaret	Young

r. Thomas Ziegler left his job at the bank on 1st November 2019. Write an appropriate statement to
amend his Employee record

mycode:

```
SELECT * FROM public.employee ORDER BY emp_id ASC;
```

myresult:



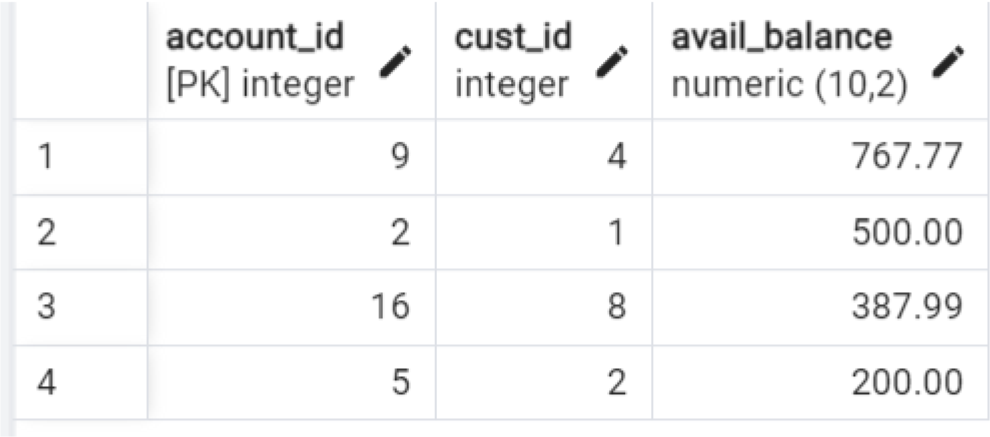
11	11	2019-11-01	Thomas	Ziegler	2000-10-23	TELLER	2	1	10

s. List all customers with a product code of ‘SAV’. Sort them by decreasing available balance

mycode:

```
SELECT account_id， cust_id， avail_balance
FROM account
WHERE product_cd=’SAV’
ORDER BY avail_balance DESC;
```

myresult:



	account_id [PK] integer	cust_id integer	avail_balance numeric (10,2)
1	9	4	767.77
2	2	1	500.00
3	16	8	387.99
4	5	2	200.00