Account Table

1. Convert the Date attribute into a yyyy-mm-dd by adding 24 in year

format in Excel or SQL

- 1993 -> 2017
- 1994 -> 2018
- 1995 -> 2019
- 1996 -> 2020
- 1997 -> 2021

Ans: First alter the table's date column datatype length to accommodate changes in its format

alter table account

modify column date varchar(10);

then, Converting the Date attribute into a yyyy-mm-dd by adding 24 in year

update account

set date = date_format(date_add(date, interval 24 year), '%Y-%m-%d');

OUTPUT:

account_id	district_id	frequency	date
576	55	POPLATEK MESICNE	2017-01-01
3818	74	POPLATEK MESICNE	2017-01-01
704	55	POPLATEK MESICNE	2017-01-01
2378	16	POPLATEK MESICNE	2017-01-01
2632	24	POPLATEK MESICNE	2017-01-02
1972	77	POPLATEK MESICNE	2017-01-02
1539	1	POPLATEK PO OBRATU	2017-01-03
793	47	POPLATEK MESICNE	2017-01-03

2. Replace in frequency attribute “POPLATEK MESICNE” AS Monthly Issuance, “POPLATEKTYDNE” AS Weekly Issuance, and “POPLATEK POBRATU” AS Issuance After a Transaction in Excel or create a case statement in SQL.

Ans:

```
update account
set frequency = case
                    when frequency='POPLATEK MESICNE' then 'Monthly Issuance'
                    when frequency = 'POPLATEKTYDNE' then 'Weekly Issuance'
                    else 'Issuance After a Transaction'
                end;
```

OUTPUT:

account_id	district_id	frequency	date
576	55	Monthly Issuance	2017-01-01
3818	74	Monthly Issuance	2017-01-01
704	55	Monthly Issuance	2017-01-01
2378	16	Monthly Issuance	2017-01-01
2632	24	Monthly Issuance	2017-01-02
1972	77	Monthly Issuance	2017-01-02
1539	1	Issuance After a Transaction	2017-01-03
793	47	Monthly Issuance	2017-01-03

3. Create a Custom Column Card_Assigned and assign below :

● Silver -> Monthly issuance

● Diamond - weekly issuance

● Gold - Issuance after a transaction

Ans: First add the custom column in the table:

alter table account

add Card_Assigned varchar(20);

then, insert data in it as per the question

update account

set Card_Assigned = case

when frequency = 'Monthly issuance' then 'Silver'

when frequency = 'weekly issuance' then 'Diamond'

when frequency = 'Issuance after a transaction' then 'Gold'

end;

OUTPUT:

account_id	district_id	frequency	date	Card_Assigned
576	55	Monthly Issuance	2017-01-01	Silver
3818	74	Monthly Issuance	2017-01-01	Silver
704	55	Monthly Issuance	2017-01-01	Silver
2378	16	Monthly Issuance	2017-01-01	Silver
2632	24	Monthly Issuance	2017-01-02	Silver
1972	77	Monthly Issuance	2017-01-02	Silver
1539	1	Issuance After a Transaction	2017-01-03	Gold
793	47	Monthly Issuance	2017-01-03	Silver

CARD Table

1. Replace type attribute value “junior” as Sliver, “Classic” as Gold,

And “Gold” as Diamond by using replace in Excel or by using update

in SQL.

Ans:

```
update card
set `type` = case
                when `type` = 'junior' then 'Sliver'
                when `type` = 'Classic' then 'Gold'
                when `type` = 'Gold' then 'Diamond'
            end;
```

output:

card_id	disp_id	type	issued
1005	9285	Gold	931107 00:00:00
104	588	Gold	940119 00:00:00
747	4915	Gold	940205 00:00:00
70	439	Gold	940208 00:00:00
577	3687	Gold	940215 00:00:00
377	2429	Gold	940303 00:00:00
721	4680	Sliver	940405 00:00:00
437	2762	Gold	940601 00:00:00

2. Convert issued attribute into yyyy-mm-dd adding 23 in year.

ANS:

First used `**STR_TO_DATE**` to parse the exsting date string, then use `**DATE_FORMAT()**` function to convert it into the desired formet

```
update card
set issued = date_format(date_add(str_to_date(issued, '%y%m%d %H:%i:%s'), interval 23 year),'%Y-%m-%d');
```

OUTPUT:

card_id	disp_id	type	issued
1005	9285	Gold	2016-11-07
104	588	Gold	2017-01-19
747	4915	Gold	2017-02-05
70	439	Gold	2017-02-08
577	3687	Gold	2017-02-15
377	2429	Gold	2017-03-03
721	4680	Sliver	2017-04-05
437	2762	Gold	2017-06-01

DISTRICT Table

1. Change all column names and delete the attributes a12

Ans:

alter table district

drop column A12

then,

alter table district

change A1 districtCode int,

change A2 districtName varchar(20),

change A3 region varchar(20),

change A4 noOfInhabitants int,

change A5 Muncipalities1 int,

change A6 Muncipalities2 int,

change A7 Muncipalities3 int,

change A8 Muncipalities4 int,

change A9 NoOfCities int,

change A10 ratioUrbanInhabitant decimal(10, 2),

change A11 avgSalary int,

change A13 Unemployment96 decimal(10,2),

change A14 entrepreneurs int,

change A15 crimes95 int,

change A16 crimes96 int

;

OUTPUT:

Field Types

#	Field	Schema	Table	Type	Character Set	Displa
1	districtCode	q1	district	INT	binary	
2	districtName	q1	district	VARCHAR	utf8mb4	
3	region	q1	district	VARCHAR	utf8mb4	
4	noOfInhabitants	q1	district	INT	binary	
5	Muncipalities1	q1	district	INT	binary	
6	Muncipalities2	q1	district	INT	binary	
7	Muncipalities3	q1	district	INT	binary	
8	Muncipalities4	q1	district	INT	binary	
9	NoOfCities	q1	district	INT	binary	
10	ratioUrbanInhabitant	q1	district	DECIMAL	binary	
11	avgSalary	q1	district	INT	binary	
12	Unemployment96	q1	district	DECIMAL	binary	
13	entrepreneurs	q1	district	INT	binary	
14	crimes95	q1	district	INT	binary	
15	crimes96	q1	district	INT	binary	

LOAN Table

1. Convert the Date Attribute into yyyy-mm-dd format adding 23 in year.

Ans:

```
alter table loan
modify column `date` date;
```

then,

```
update loan
set `date` = date_add(`date`, interval 23 year);
```

2. Convert Status Attribute value “A” as Contract Finished, “B” as Loan Not Paid, “C” as Running Contract, and “D” Client in debt.

Ans:

```
update loan
set `status` = case
when `status` = 'A' then 'Contract Finished'
when `status` = 'B' then ' Loan Not Paid'
when `status` = 'C' then 'Running Contract'
when `status` = 'D' then 'Client in debt'
end;
```

In th Transactions Table do the following , whosoever count is highest sort it in

descending order and change the year from 2022,2021,2020 and so on

--DATA TRANSFORMATION

/*

2021 -> 2017

2020 -> 2018

2019 -> 2019 -- NO CHANGE

2018 -> 2020

2017 -> 2021

2016 -> 2022

UPDATE TRANSACTIONS

SET BANK = 'Sky Bank' WHERE BANK IS NULL AND YEAR(DATE) =

2022;

UPDATE TRANSACTIONS

SET BANK = 'DBS Bank' WHERE BANK IS NULL AND YEAR(DATE) =

2021;

UPDATE TRANSACTIONS

SET BANK = 'Northern Bank' WHERE BANK IS NULL AND YEAR(DATE)

= 2019;

UPDATE TRANSACTIONS

SET BANK = 'Southern Bank' WHERE BANK IS NULL AND YEAR(DATE)

= 2018;

Ans: In the Transactions Table, whosoever count is highest sorting it in

descending order:

select *, count(*) as transactions from trans group by account_id order by transactions desc;