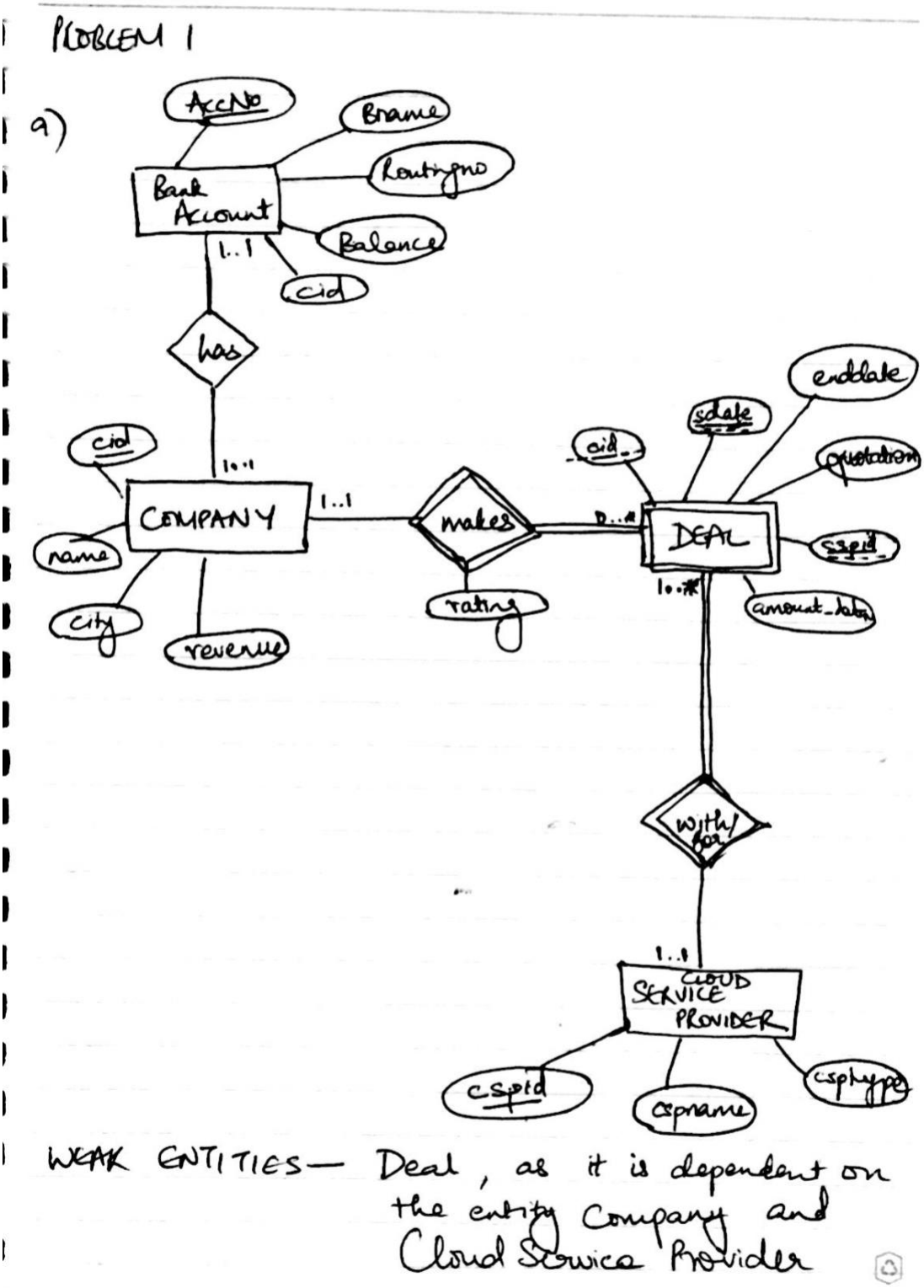
DB HOMEWORK #2

PROBLEM 1



PRIMARY KEYS –

Company - cid
Cloud Service Provider – Cspid
Bank Account – Acountno.
Deal – (cid, cspid, sdate)

FOREIGN KEYS –

cid - in Bank Account references Company
in Deal references Company

cspid - in Deal refers CloudServiceProvider

ASSUMPTIONS –

Each Company rates the deal between 1-5 and the deal startdate is the date the deal was made.

6) SCHEMA –

Company (cid , cname, ccity, crevenue)
BankAccount (AccNo , Bname , Routingno, Balance, cid)
CloudServiceProvider (cspid , cspname, csptype)
Deal (cid, cspid , sdate , enddate, quotation, Amount_data)

PRIMARY KEYS –

Company – cid
Cloud Service Provider – Cspid
Bank Account – Acountno.
Deal – (cid, cspid, sdate)

FOREIGN KEYS –

cid – in Bank Account references Company
in Deal references Company

cspid – in Deal refers CloudServiceProvider

HOMEWORK #2

Problem 1

c) 1.

```
SELECT CompanyId, sum(quotation) as total_spa
FROM Deal
GROUP BY CompanyId ;
```

2.

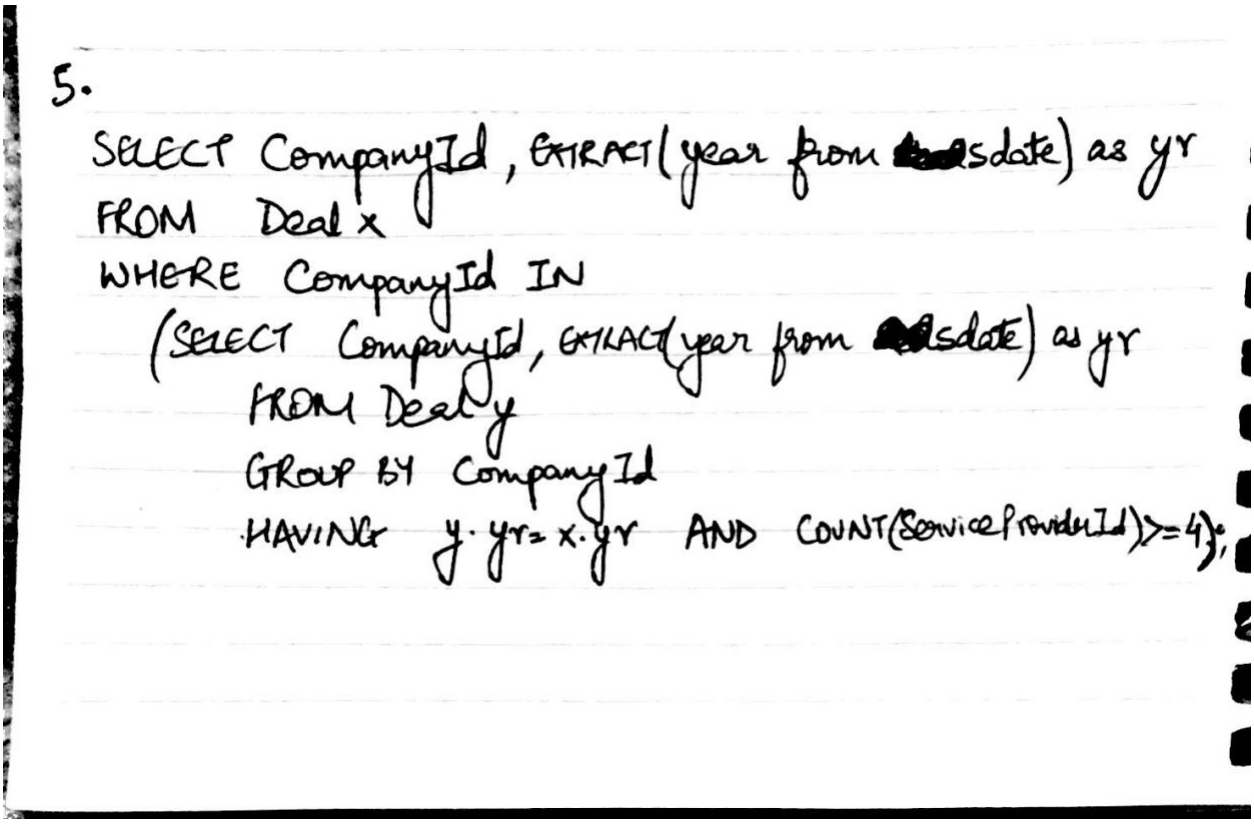
```
SELECT ServiceProviderId, sum(amount_data)
                                as total_data
FROM Deal
GROUP BY ServiceProviderId
HAVING EXTRACT(year FROM date)
                        = 2017;
```

3.

```
SELECT ServiceProviderId, avg(rating) as avg_rating
FROM Deal
GROUPBY ServiceProviderId
HAVING EXTRACT(year FROM date)=2016;
```

4.

```
SELECT CompanyId
FROM Deal
GROUP BY CompanyId
 HAVING  sum(quotation)/sum(amount_data)>=100
```



PROBLEM 2

a)

```
CREATE DATABASE leasing;

CREATE TABLE leasing.Customer (
   cid int NOT NULL,
   cname varchar(50) NOT NULL,
   cphone DOUBLE NOT NULL,
   ccity varchar(50) NOT NULL,
   PRIMARY KEY (cid)
   );

CREATE TABLE leasing.Landlord (
                lid int NOT NULL,
                lname varchar(50) NOT NULL,
                lphone DOUBLE NOT NULL,
                lcity varchar(50) NOT NULL,
                PRIMARY KEY (lid)
   );
```

```
CREATE TABLE leasing.Residence (
                rid int NOT NULL,
                rname varchar(50) NOT NULL,
                rstate varchar(50) NOT NULL,
                rcity varchar(50) NOT NULL,
                raddr varchar(50) NOT NULL,
                rtype varchar(50) NOT NULL,
                rarea varchar(50) NOT NULL,
                lid int NOT NULL,
                PRIMARY KEY (rid),
                FOREIGN KEY (lid) REFERENCES Landlord(lid)
  );

CREATE TABLE leasing.Leases (
                cid int NOT NULL,
                rid int NOT NULL,
    startdate datetime NOT NULL,
    enddate datetime NOT NULL,
                price DOUBLE NOT NULL,
                PRIMARY KEY (cid, rid, startdate),
    FOREIGN KEY (cid) REFERENCES Customer(cid),
                FOREIGN KEY (rid) REFERENCES Residence(rid)
  );

CREATE TABLE leasing.Rating (
                cid int NOT NULL,
                rid int NOT NULL,
    rtime datetime NOT NULL,
                score int NOT NULL CHECK (score>=1 AND score<=5),
                PRIMARY KEY (cid, rid, rtime),
                FOREIGN KEY (cid) REFERENCES Customer(cid),
                FOREIGN KEY (rid) REFERENCES Residence(rid)
  );
```

b)

1.

```
SELECT lid,rid
        FROM leasing.Landlord natural join leasing.Residence
   WHERE lcity='Brooklyn' AND rcity='Queens';
```

2.

```
SELECT count(r.rid),EXTRACT(Month from l.startdate) as month_in_date
        FROM leasing.Residence r join leasing.Leases l on r.rid=l.rid
   WHERE EXTRACT(Year from l.startdate)='2017' AND r.rcity='Chicago'
   GROUP BY month_in_date;
```

3.

```
SELECT rtype, COUNT(DISTINCT (lid)) AS cnt
                FROM leasing.Residence
                GROUP BY rtype
                HAVING cnt=max(cnt);
```

4.

```
SELECT c.*
        FROM leasing.Customer c JOIN leasing.Leases l ON c.cid=l.cid
   WHERE l.price = (Select max(price) from leasing.Leases);
```

5.

```
 SELECT avg(price), lid, lname
        FROM leasing.Leases natural join leasing.Residence natural join leasing.Landlord
   GROUP BY lid;
```

c)

1) $\pi_{lid, rid}$ $\sigma_{lcity = 'Brooklyn' \text{ and } rcity = 'Queens'}$ (Landlord ⋈ Residence)

2) $\pi_{count(Residence.rid),\ month\_in\_date}$ $G_{month\_in\_date}$ $\sigma_{extract(year\ from\ startdate) = 2019,\ Residence.rcity = 'Chicago'}$ (Residence ⋈ Leases)

3) $\pi_{rtype,\ count(distinct(lid))\ as\ cnt}$ G_{rtype} $\sigma_{cnt = max(cnt)}$ (Residence)

4) $\pi_{customer.cid,\ Customer.cname,\ Customer.phone,\ Customer.ccity}$ $\sigma_{price = (\pi_{price}\ \sigma_{max(price)}(Leases))}$ (Customer ⋈ Leases)