1.

COMP 5120 Assignment 3

Thomas Eden

1. a) $\pi_{name}(\sigma_{age>21}(User))$

   User age >21 = Jane (22)

   Select name of user = {Jane}

b) User $\bowtie$ Job $\bowtie$ City

   User NATURAL JOIN Job $\rightarrow$ User.job_id = Job.job_id

   User NATURAL JOIN City $\rightarrow$ User.city_id = City.city_id

   User

id	name	age	gender	job_id	city_id	job	city
1	John	20	Male	1	1	Software Engineer	New York City
2	Sara	20	Female	1	1	Software Engineer	New York City
3	Victor	21	Male	2	2	Accountant	Boston
4	Jane	22	Female	3	3	Dentist	Atlanta
5	Mason	21	Male	1	1	Software Engineer	New York City
6	Jason	20	Male	2	3	Accountant	Atlanta

c) $\sigma_{id>2 \; OR \; age<21}(User)$

   id>2 $\rightarrow$ {Victor, Jane, Mason, Jason}

   age<21 $\rightarrow$ {John, Sara, Jason}

   {John, Sara, Victor, Jane, Mason, Jason} All users

d) $\pi_{name, gender}(\sigma_{city="Boston"}(User \bowtie City))$

   Only Victor matches the city Boston.

   = {Victor, Male}

2. INSERT INTO Statements (Screenshots)

```sql
INSERT INTO orderparts.parts (`Pno`, `Pname`, `Qty`, `Price`, `Olevel`)
VALUES (1, 'Screw', 150, 17.00, 75),
    (2, 'Bolt', 100, 6.00, 50),
    (3, 'Wrench', 2, 30.00, 150),
    (4, 'Nut', 150, 3.00, 90),
    (5, 'Washer', 250, 22.00, 130);


INSERT INTO orderparts.customers (`Cno`, `Cname`, `Street`, `Zip`, `Phone`)
VALUES (111, 'John', '111 Hallmark Ave', 31001, '111-1111'),
    (222, 'Cynthia', '222 Lincoln Blvd', 32002, '222-2222'),
    (333, 'Maria', '333 Palm St', 33003, '333-3333'),
    (444, 'Edwin', '444 Elm St', 34004, '444-4444'),
    (555, 'Antonio', '555 Maple Ave', 35005, '555-5555');


INSERT INTO orderparts.employees (`Eno`, `Ename`, `Zip`, `Hdate`)
VALUES (1000, 'Lauren', 31001, '2018-10-27'),
    (2000, 'Ryan', 33003, '2020-05-13'),
    (3000, 'George', 37007, '2019-02-09');


INSERT INTO orderparts.zipcodes (`Zip`, `City`)
VALUES (31001, 'Atlanta'),
    (32002, 'Nashville'),
    (33003, 'Chicago'),
    (34004, 'Houston'),
    (35005, 'Philadelphia');


INSERT INTO orderparts.orders (`Ono`, `Cno`, `Eno`, `Received`, `Shipped`)
VALUES (1001, 111, 1000, '2025-01-03', '2025-01-10'),
    (2002, 222, 2000, '2025-01-06', '2025-01-12'),
    (3003, 333, 3000, '2025-01-15', '2025-01-26');


INSERT INTO orderparts.orderdetails (`Ono`, `Pno`, `Qty`)
VALUES (1001, 1, 150),
    (1001, 2, 100),
    (2002, 3, 2),
    (2002, 4, 150),
    (3003, 5, 250);
```

a) Edwin, Antonio

```sql
SELECT Cname
FROM Customers
WHERE Cno NOT IN (SELECT Cno FROM Orders);
```

b) John, Cynthia

```sql
SELECT DISTINCT Cname
FROM Customers
JOIN Orders ON Customers.Cno = Orders.Cno
JOIN OrderDetails ON Orders.Ono = OrderDetails.Ono
JOIN Parts ON OrderDetails.Pno = Parts.Pno
WHERE Price < 20.00;
```

c) John

```sql
SELECT DISTINCT Cname
FROM Customers
JOIN Orders ON Customers.Cno = Orders.Cno
JOIN Employees ON Orders.Eno = Employees.Eno
JOIN ZipCodes ON Employees.Zip = ZipCodes.Zip
WHERE ZipCodes.City = 'Atlanta';
```
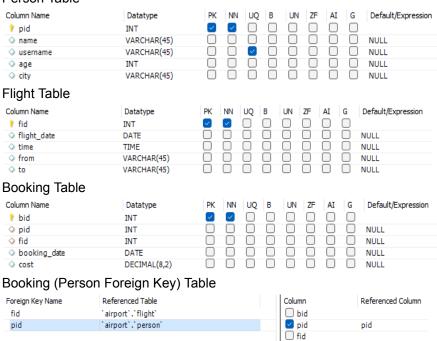
3.

Person Table

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
pid	INT	✓	✓							
name	VARCHAR(45)									NULL
username	VARCHAR(45)			✓						NULL
age	INT									NULL
city	VARCHAR(45)									NULL

Flight Table

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
fid	INT	✓	✓							
flight_date	DATE									NULL
time	TIME									NULL
from	VARCHAR(45)									NULL
to	VARCHAR(45)									NULL

Booking Table

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
bid	INT	✓	✓							
pid	INT									NULL
fid	INT									NULL
booking_date	DATE									NULL
cost	DECIMAL(8,2)									NULL

Booking (Person Foreign Key) Table

Foreign Key Name	Referenced Table		Column	Referenced Column
fid	`airport`.`flight`		☐ bid	
pid	`airport`.`person`		☑ pid	pid
			☐ fid	
			☐ booking_date	
			☐ cost	

Booking (Flight Foreign Key) Table

Foreign Key Name	Referenced Table
fid	`airport`.`flight`
pid	`airport`.`person`

	Column	Referenced Column
☐	bid	
☐	pid	
☑	fid	fid
☐	booking_date	
☐	cost	

Person INSERT INTO

```
INSERT INTO airport.person (`pid`, `name`, `username`, `age`, `city`)
VALUES (7, 'Alicia', 'alicia099', 25, 'Seattle'),
    (8, 'Mason', 'mason092', 33, 'Nashville'),
    (12, 'Perry', 'perry001', 24, 'Nashville'),
    (16, 'Lauren', 'lauren094', 30, 'Chicago');
```

Flight INSERT INTO

```
INSERT INTO airport.flight (`fid`, `flight_date`, `time`, `from`, `to`)
VALUES (101, '2020-01-01', '12:15:00', 'Seattle', 'Atlanta'),
    (123, '2020-01-02', '14:00:00', 'Nashville', 'Boston'),
    (333, '2020-01-02', '13:45:00', 'Nashville', 'Atlanta'),
    (456, '2020-01-03', '11:30:00', 'Chicago', 'Atlanta');
```

Booking INSERT INTO

```
INSERT INTO airport.booking (`bid`, `pid`, `fid`, `booking_date`, `cost`)
VALUES (11111, 7, 101, '2019-12-14', 450.00),
    (12345, 8, 123, '2019-11-24', 500.00),
    (22222, 12, 333, '2019-11-15', 250.00),
    (23456, 16, 456, '2019-12-07', 350.00);
```

a. All flights to Atlanta:

```
SELECT * FROM flight WHERE `to` = 'Atlanta';
```

	fid	flight_date	time	from	to
▶	101	2020-01-01	12:15:00	Seattle	Atlanta
	333	2020-01-02	13:45:00	Nashville	Atlanta
	456	2020-01-03	11:30:00	Chicago	Atlanta
*	NULL	NULL	NULL	NULL	NULL

b. All people who have bookings on at least one flight:

```
SELECT DISTINCT person.*
FROM person
JOIN booking ON person.pid = booking.pid;
```

	pid	name	username	age	city
▶	7	Alicia	alicia099	25	Seattle
	8	Mason	mason092	33	Nashville
	12	Perry	perry001	24	Nashville
	16	Lauren	lauren094	30	Chicago

c. Flights that are scheduled on 01/01/2020 or 01/02/2020:

```
SELECT * FROM flight
WHERE flight_date IN ('2020-01-01', '2020-01-02');
```

	fid	flight_date	time	from	to
▶	101	2020-01-01	12:15:00	Seattle	Atlanta
	123	2020-01-02	14:00:00	Nashville	Boston
	333	2020-01-02	13:45:00	Nashville	Atlanta
*	NULL	NULL	NULL	NULL	NULL

4.
a. Insertion Anomaly: You can't assign a new employee if they don't have an appointment.
Deletion Anomaly: Removing a patient will also remove the employee's info.
Modification Anomaly: Updating an employee or patient's info will need many changes to come into effect.
b. Separate tables for 3NF Normalization have been created below:

Patient Table

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
▮ patient_id	VARCHAR(45)	☑	☑	☐	☐	☐	☐	☐	☐
◇ patient_name	VARCHAR(45)	☐	☑	☐	☐	☐	☐	☐	☐

Employee Table

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
▮ emp_id	VARCHAR(45)	☑	☑	☐	☐	☐	☐	☐	☐
◇ emp_name	VARCHAR(45)	☐	☑	☐	☐	☐	☐	☐	☐

Appointment Table

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
▮ appointment_id	INT	☑	☑	☐	☐	☐	☐	☐	☐
◆ emp_id	VARCHAR(45)	☐	☑	☐	☐	☐	☐	☐	☐
◆ patient_id	VARCHAR(45)	☐	☑	☐	☐	☐	☐	☐	☐
◇ treatment_id	VARCHAR(45)	☐	☑	☐	☐	☐	☐	☐	☐
◇ date	DATE	☐	☑	☐	☐	☐	☐	☐	☐
◇ time	TIME	☐	☑	☐	☐	☐	☐	☐	☐

Appointment (Patient Foreign Key) Table

Foreign Key Name	Referenced Table		Column	Referenced Column
emp_id	`doctor_patient_appointment`.`employee`		☐ appointment_id	
patient_id	`doctor_patient_appointment`.`patient`		☐ emp_id	
			☑ patient_id	patient_id
			☐ treatment_id	
			☐ date	
			☐ time	

Appointment (Employee Foreign Key) Table

Foreign Key Name	Referenced Table		Column	Referenced Column
emp_id	`doctor_patient_appointment`.`employee`		☐ appointment_id	
patient_id	`doctor_patient_appointment`.`patient`		☑ emp_id	emp_id
			☐ patient_id	
			☐ treatment_id	
			☐ date	
			☐ time	