

2132 Project Report

CSI2132 Databases

Group 44:

Binxuan Wu	300142301
Gary Gao	300124236
Yihao Wang	300059569

Part 1: The DBMS and the programming languages that you have used in your implementation of the application.

We use “PostgreSQL” JDBC and Java EE (Web application) and Tomcat 9.0.44 to implementation our application;

Part 2: Specific steps to guide someone to install your applications.

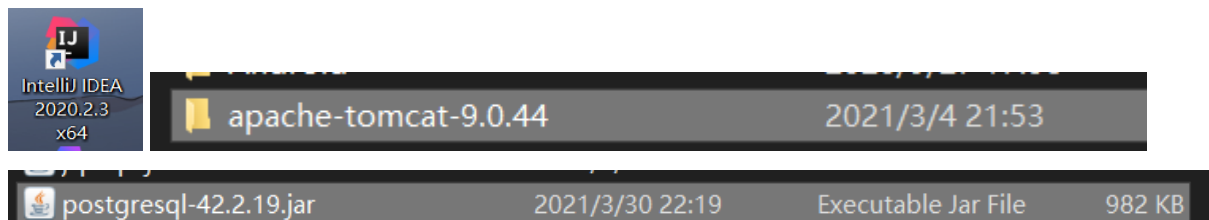
Part2: How to install our application

Requirement:

JetBrains IntelliJ IDEA, Postgresql JDBC driver, Tomcat 9.0.44

Step1:

Install IntelliJ IDEA, postgresql JDBC driver and Tomcat 9.0.44 into your computer



Step2:

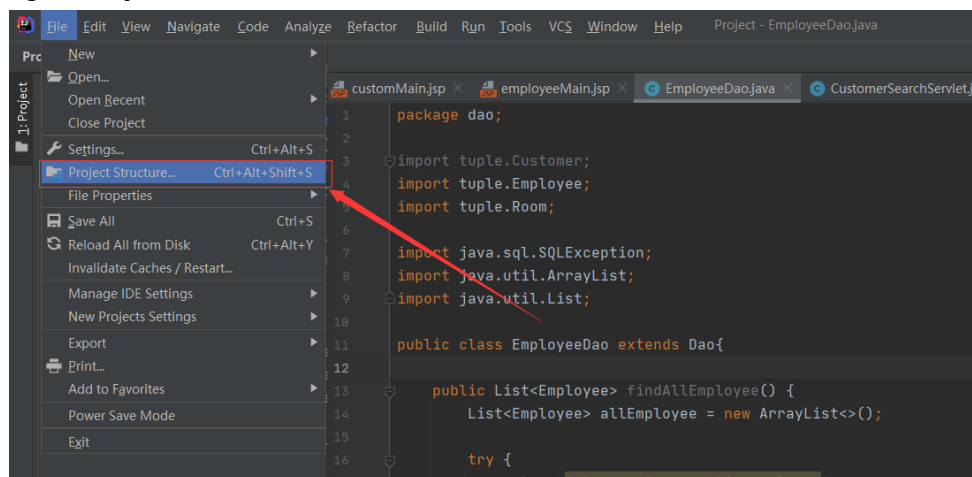
Add postgresql.jar into “apache-tomcat-9.0.44/lib”

Step3:

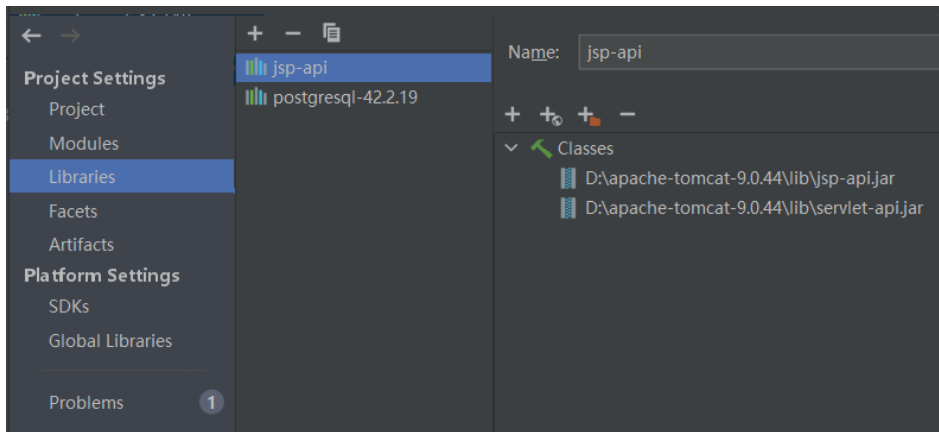
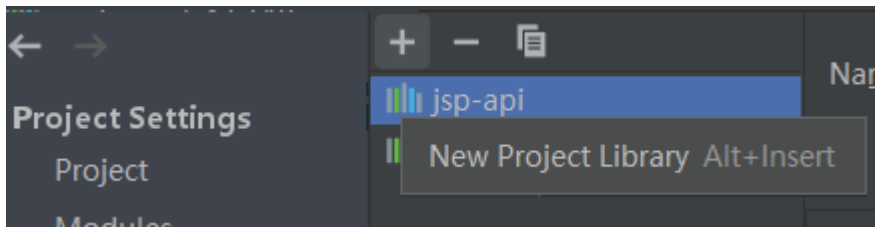
Open IntelliJ -> Open or Import -> Our application folder “Project”

Step4:

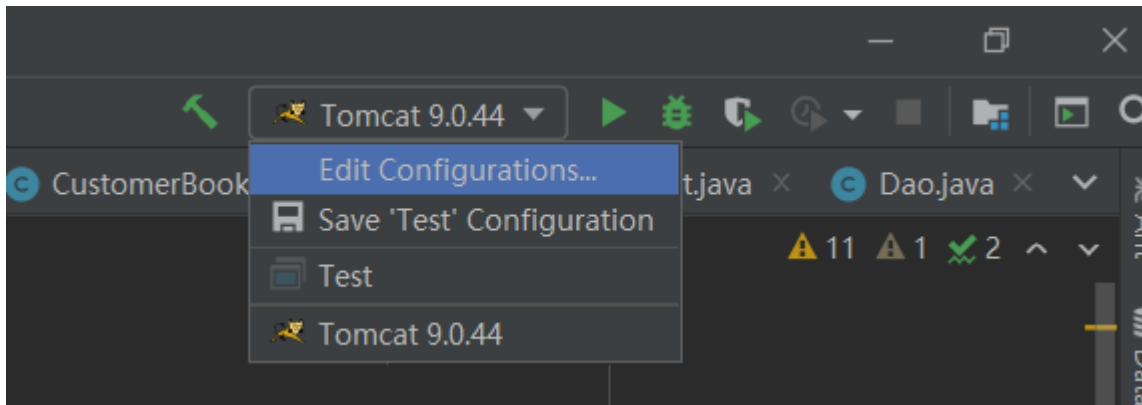
Open Project Structure

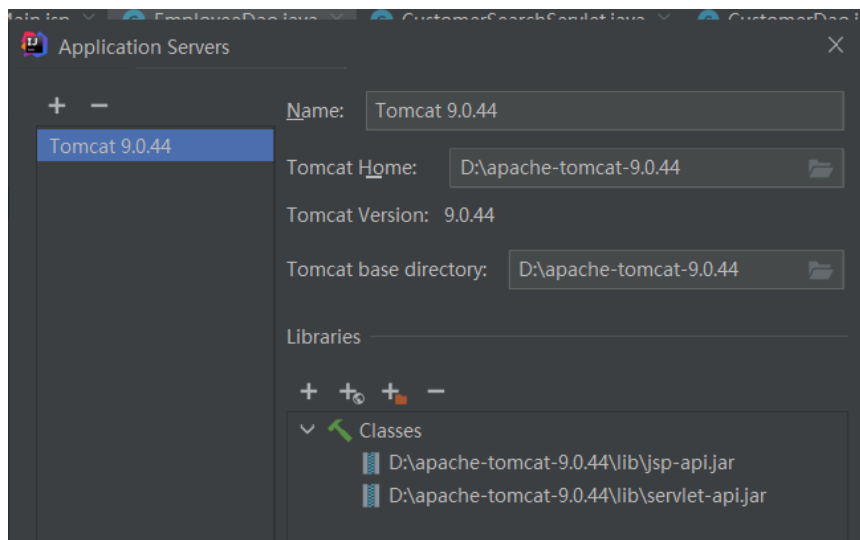
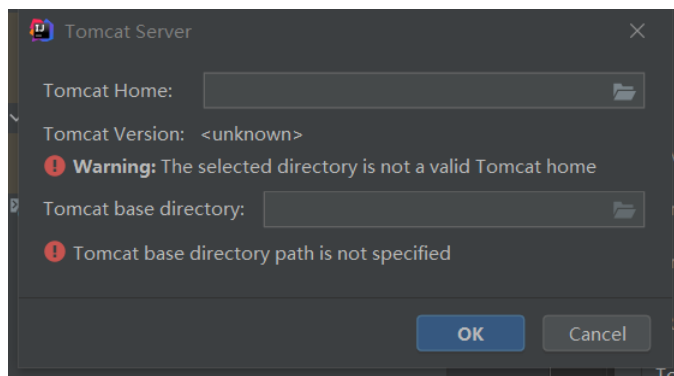
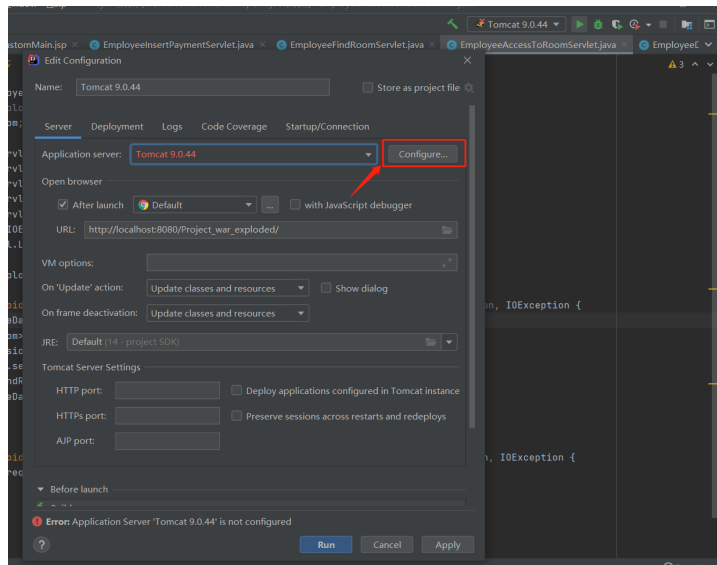


Step5: In Project Settings -> Libraries -> Choose “+” New project library -> Java -> In your “apache-tomcat-9.0.44/lib” Add “jsp-api.jar”, “servlet-api.jar”, And also Add “posetgresql-42.2.19.jar” as shown in follow.

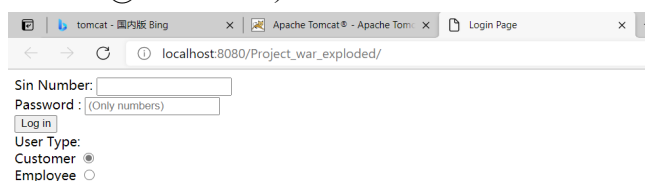


Step 6: At top-right -> Edit Configuration -> Choose “Configure...” (shown in screenshot)
 -> Set Tomcat Home and base directory as your Tomcat root (For me is “D:\apache-tomcat-9.0.44”).





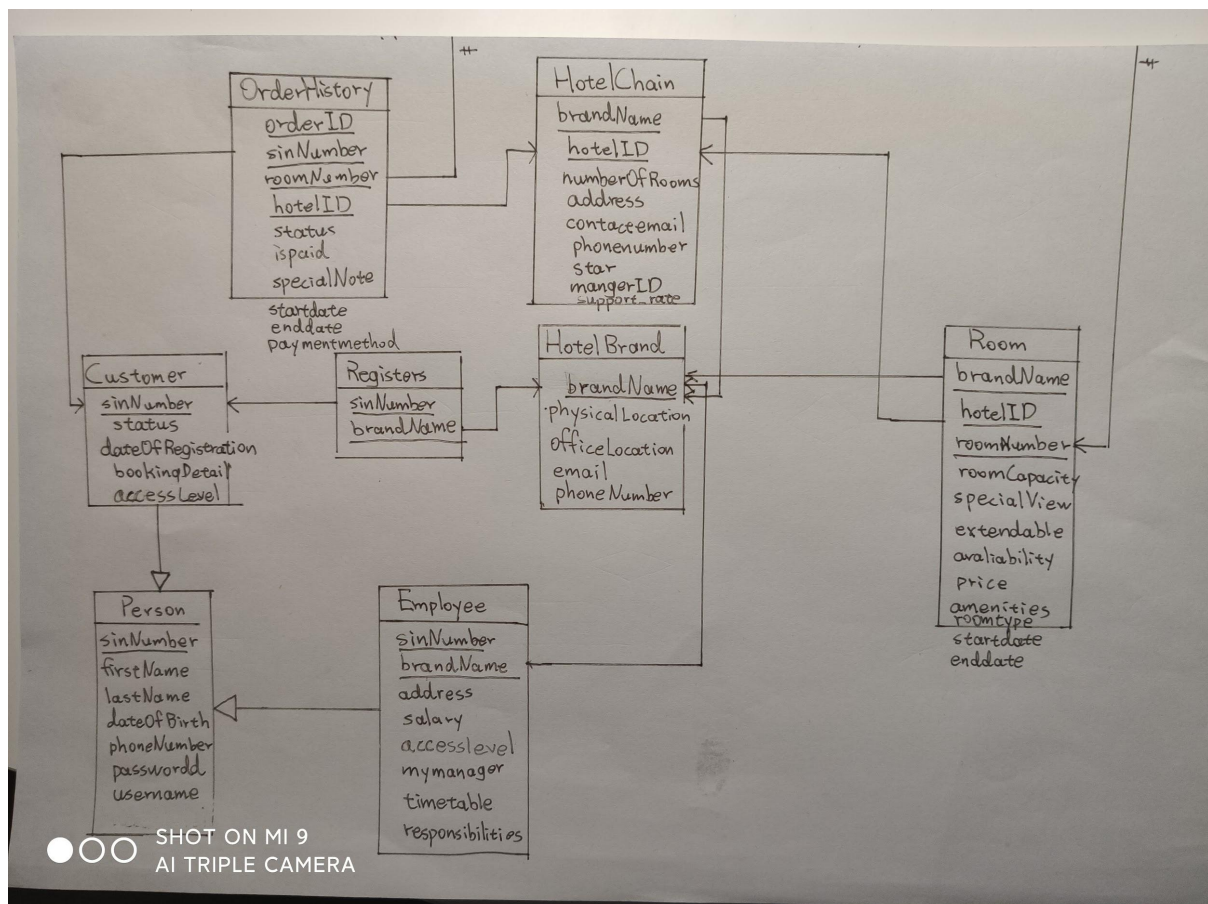
Step7: then you can click on the top-right run Tomcat 9.0.44 to open our application by your browser (If you have any problem with installing our application please contact me at bwu006@uottawa.ca)



Part3: The answers to points 4 to 8 defined above for the 2nd deliverable. You should also provide all necessary code (see points 2 and 3 below). You can provide a reference in your report to the code used in each question.

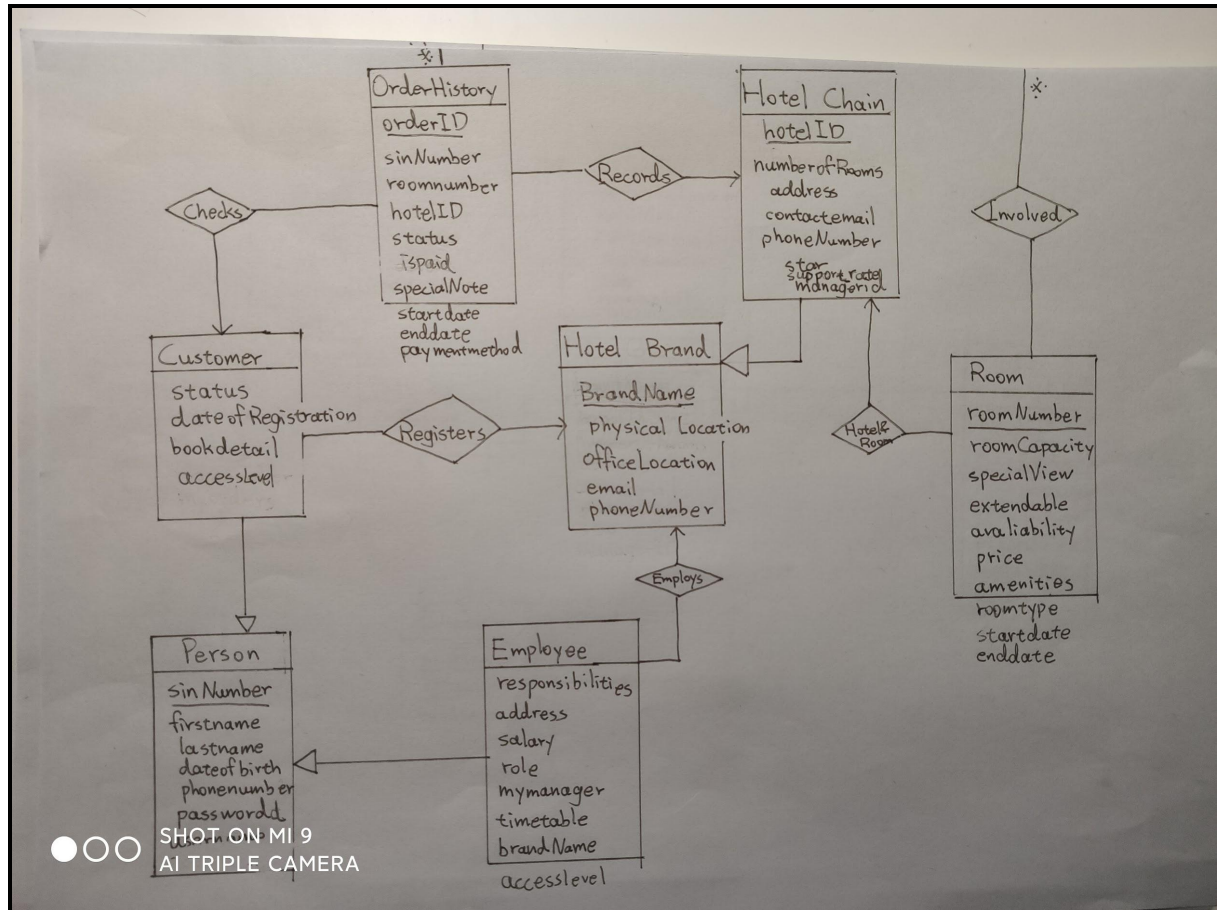
Note: The more detailed code can be found in our database 'group_b02_g44'

4. (3%) Implement the database according to your Relational model and the constraints that you have defined. Implement the relations in PostgreSQL and make sure you create indexes and constraints as appropriate. If you discover flaws that require changes to your E-R Diagram, make sure these changes are captured in your relational model.



A few changes were made to our er diagram and schema diagram, we removed a few redundant variables. And for one to many relations on a schema diagram, instead of creating a new table and storing the primary key for both ends, we chose to add some of the primary key of a table into the other table as a foreign key, and we remained the necessary ones.

Also, we added some very useful variables such as access level in table person, so when checking the orderhistory, according to the access level we can write java code of deciding which information to show and which not to show. We also removed unnecessary variables such as my orders in customer, since it can already be found in the table orderhistory.



For the process of building our database, our work can be referenced in our database, group_b02_g44, to conclude our work, we created tables with referential, primary, foreign and domain constraints, we inserted our data and lastly, we created trigger functions and triggers to validate our data.(a few examples can be found in question 5 and question 6). And also, all of our variables are type integer, text and date.

5. (2%) Create the necessary SQL modifications (use queries and triggers). Your database should allow insert, delete and update operations of data in your database according to the referential integrity constraints that you have defined. Give the SQL code for at least 2 queries and 2 triggers in your report.

```

1 create trigger check_hotel
2 before update on hotelchain
3 for each row
4 execute procedure check_hotel()
5

```

First Trigger: this trigger is used to check the new data the user about to update on the table hotelchain, it checks

the number of star, and if BrandName, address, hotelID is null. Also, there is a similar trigger that checks before insert on hotelchain.

```
1  create function check_hotel()
2  returns trigger as
3  $BODY$
4  BEGIN
5
6  if new.star>5 then
7  RAISE EXCEPTION 'star should not be higher than 5';
8  end if;
9
10 if new.star<0 then
11 RAISE EXCEPTION 'star should not be smaller than 0';
12 end if;
13
14 if new.BrandName is null then
15 RAISE EXCEPTION 'name cannot be null';
16 end if;
17
18
19 if new.address is null then
20 RAISE EXCEPTION 'address should not be null';
21 end if;
22
23 if new.hotelID is null then
24 RAISE EXCEPTION 'hotelID should not be null';
25 end if;
26
27 return new;
28 end
29 $BODY$ LANGUAGE plpgsql;
```

```
1  create trigger check_person_info
2  before update on person
3  for each row
4  execute procedure check_password_username()
```


Second Trigger: similar to the trigger above, this trigger is used to test the validity of the updating information of the table person, it checks if the password is null and the password should have a length of 9 digits, it checks the username, length of the sinnumber. **In addition**, there is a similar trigger that checks before insert on the table person.

```

1  create function check_password_username()
2  returns trigger as
3  $BODY$
4  BEGIN
5
6  if new.passwordd is null then
7  RAISE EXCEPTION 'password cannot be null';
8  end if;
9
10 if new.passwordd < 100000000 then
11 RAISE EXCEPTION 'password cannot be shorter than 8 digit';
12 end if;
13
14 if new.username is null then
15 RAISE EXCEPTION 'username cannot be null';
16 end if;
17
18 if new.sinnumber < 1000000000 then
19 RAISE EXCEPTION 'sin number should be at least 9 digit';
20 end if;
21
22 return new;
23 end
24 $BODY$ LANGUAGE plpgsql;

```

Referential Integrity Constraint 1:

This table Employee includes foreign key constraint and domain constraints. The variable sinNumber is a foreign key from table person and the variable brandname is a foreign key from the table hotelbrand and they have the update cascade and delete cascade on differently.

```

2  create table Employee(sinNumber integer, responsibilities text, address text,
3                          salary integer, role text, myManager text, timeTable text,
4                          Foreign key(sinNumber) references Person on delete cascade,
5                          brandName text, foreign key (brandName) references HotelBrand match simple
6                          on update cascade on delete cascade
7                          ,constraint salary_check check (salary > 0))

```

Referential Integrity Constraint2:

This table Orders includes foreign key sinNumber from the table person and a foreign key orderID from the table OrderHistory. Since sinnumber and orderID should never be updated they only have to delete cascade on.


```

1 create table Orders(sinNumber integer, orderID integer, status text,
2 constraint for1 foreign key (sinNumber) references person on delete cascade,
3 constraint for2 foreign key (orderID) references OrderHistory on delete cascade)

```

(This table was found useless since the table orderhistory includes all the information it needs, but the code used in this example is similar to the one we used to create orderhistory)

6. (2%) Populate your relations. Your database needs to be populated with data after creating it. Adding enough data will make your queries interesting. You should have at least 6 different hotel chains with at least 5 hotels each. The hotels of each chain should belong to at least 3 different star categories. You should also add at least 5 rooms with different capacities in each hotel and you should include several customers, employees, etc. Add at least 3 hotels from different hotel chains in Ottawa city.

We did populate our relations but unfortunately, we did not save these code for record, so I wrote these code for example.

For creating a new hotel brand:

```

insert into hotelbrand(brandname,physicallocation,officelocation,email,phonenumber)
values('the best hotel','Ottawa','Toronto','lovememore@gmail.com',6477777777);

```

For creating a new employee(person):

```

insert into person(sinnumber,firstname,lastname,dateofbirth,phonenumber,passwordd,username)
values(555555554,'Gary','TheGreat',200103098,752354,34328882,'adsjgsh');

insert into employee(sinnumber,brandname,address,salary,accesslevel,
mymanager,timetable,responsibilities)
values(555555554,'the best hotel','heaven road',8000,'manager','myself',
'monday to friday 8am to 8pm','to manage');

```

For creating a hotel from the above hotel brand:

```

insert into hotelchain(brandname,hotelid,numberofrooms,
address,contactemail,phonenumber,star,managerid,support_rate)
values('the best hotel',4463,5,'Ottwa road 462','callme@gmail.com',1234567890,5,555555554,100);

```

For inserting a room into the above hotel:

```
insert into room(brandname,hotelid,roomnumber,roomcapacity,specialview,extendable,availability
                ,price,amenities,roomtype,startdate,enddate)
values('the best hotel',4463,9001,3,'sea view','no','yes',700,'none','supreme',
      '01/03/2001','01/04/2001');
```

For creating a customer(person):

```
insert into person(sinnumber,firstname,lastname,dateofbirth,phonenummer,passwordd,username)
values(555555555,'Alex','TheGreat',200103012,7523984,34789245,'adsjfh');

insert into customer(sinnumber,status,dateofregistration,bookingdetail,accesslevel)
values(555555555,'active',20200405,'love sea view','customer');
```

For registering a customer(VIP) into a specific hotelbrand:

```
insert into registers(sinnumber,brandname)
values(555555555,'the best hotel');
```

For a customer booking a room:

```
insert into orderhistory(orderid,sinnumber,roomnumber,hotelid,status,
                        | specialnote,startdate,enddate)
values(10060,555555555,9001,4463,'active','hot water please','01/02/2021','02/02/2021');

update room set availability = 'booked' where roomnumber = 9001;
```

When an order is expired and change the room back to open:

```
update orderhistory set status to 'expired' where orderid =10060;

update room set availability = 'yes' where roomnumber = 9001;
```

,

7. (3%) Several users will need to make use of the database and each will require a special application during access. (Since it is not a web application development course, a simple command line interface will suffice but you can develop an application if you are able to). Database administrators will need to use SQL either through the command line or SQL Developer. The administrator should be able to insert/delete/update all information related to customers, employees, hotels and rooms. Customers need a web interface to search for rooms and to book rooms for specific dates. Hotel employees need a lookup application that allows a quick access to rooms available and booked. Employees will transform booked rooms into rented rooms when the customers arrive at the hotel. If a customer arrives at the hotel without a

booking the employee should be able to find a room and rent it without prior booking.
An employee should also be able to insert a customer payment for a renting.

8. (4%) You need to run a list of queries the database clients are interested in. Please provide the code and the results you obtained when running the queries in your database.

1. Give the details of all currently rented rooms in a specific hotel. Please display the columns as customer name, room type, room price, rental start date, room view, hotel chain. Sort by the room price in ascending order and rental start date in descending order.

Query Editor Query History

```
1 select firstname,lastname,datee,roomnumber,roomtype,price,specialview,brandname
2 from ((customer natural join person) join orderhistory using (sinnumber))join room using(roomnumber)
3 order by price asc,datee desc;
4
```

(note that the variable date was later changed to startdate and enddate.)

	firstname text	lastname text	datee text	roomnumber integer	roomtype text	price integer	specialview text	brandname text
1	yolo7	chod	21/03/30	104	supreme	50	mountain view	golden bay
2	ppp9	chod	21/03/30	103	supreme	100	mountain view	golden bay
3	jay9	chod	21/03/30	202	supreme	100	sea view	golden bay
4	ppp0	chod	21/03/30	202	supreme	100	sea view	golden bay
5	yolo6	chod	21/03/30	402	supreme	200	bridge view	golden bay
6	ppp7	chod	21/03/30	2106	supreme	230	city view	shinny beach
7	jay3	chod	21/03/30	6405	supreme	300	city view	hey boy and girl...
8	ppp1	chod	21/03/30	101	supreme	300	sea view	golden bay
9	sam3	chod	21/03/30	201	supreme	300	sea view	golden bay
10	sam9	chod	21/03/30	2601	supreme	300	city view	shinny beach
11	dude9	chod	21/03/30	3501	supreme	300	city view	the big brand
12	dude6	chod	21/03/30	3506	supreme	300	city view	the big brand
13	ppp0	chod	21/03/30	2606	supreme	330	city view	shinny beach
14	dude3	chod	21/03/30	2302	supreme	350	mountian view	shinny beach
15	pdp4	chod	21/03/30	4105	supreme	350	mountain view	championship h...
16	ppp7	chod	21/03/30	2105	supreme	410	city view	shinny beach
17	ppp2	chod	21/03/30	2105	supreme	410	city view	shinny beach
17	ppp2	chod	21/03/30	2105	supreme	410	city view	shinny beach
18	dude1	chod	21/03/30	2105	supreme	410	city view	shinny beach
19	pdp2	chod	21/03/30	2401	supreme	600	city view	shinny beach
20	dude5	chod	21/03/30	2501	supreme	600	city view	shinny beach
21	sam5	chod	21/03/30	303	supreme	700	sea view	golden bay
22	pdp4	chod	21/03/30	4103	supreme	1300	sea view	championship h...

2. Create a view named CustomerListView that gives the details of all the customers. Please, sort the customers by hotel chain.

	<div>sinnumber</div> <div>integer</div>	<div>firstname</div> <div>text</div>	<div>lastname</div> <div>text</div>	<div>dateofbirth</div> <div>text</div>	<div>phonenumber</div> <div>text</div>	<div>password</div> <div>integer</div>	<div>username</div> <div>text</div>	<div>accesslevel</div> <div>text</div>	<div>status</div> <div>text</div>	<div>dateofregistration</div> <div>text</div>	<div>bookingdetail</div> <div>text</div>	<div>brandname</div> <div>text</div>
1	2111111140	sam0	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
2	2111111149	sam9	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
3	2111111148	sam8	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
4	2111111147	sam7	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
5	2111111146	sam6	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
6	2111111145	sam5	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
7	2111111144	sam4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
8	2111111143	sam3	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
9	2111111142	sam2	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
10	2111111141	sam1	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	baby boss hotel
11	2111111169	yolo9	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
12	2111111160	dude9	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
13	2111111161	yolo1	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
14	2111111162	yolo2	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
15	2111111163	yolo3	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
16	2111111164	yolo4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
17	2111111165	yolo5	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
18	2111111166	yolo6	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
19	2111111167	yolo7	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
20	2111111168	yolo8	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	championship h..
21	2111111117	ppp0	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
22	2111111110	ppp1	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
23	2111111119	ppp9	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
24	2111111118	ppp8	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
25	2111111117	ppp7	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
26	2111111116	ppp6	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
27	2111111115	ppp5	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
28	2111111114	ppp4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay

	<div>sinnumber</div> <div>integer</div>	<div>firstname</div> <div>text</div>	<div>lastname</div> <div>text</div>	<div>dateofbirth</div> <div>text</div>	<div>phonenumber</div> <div>text</div>	<div>password</div> <div>integer</div>	<div>username</div> <div>text</div>	<div>accesslevel</div> <div>text</div>	<div>status</div> <div>text</div>	<div>dateofregistration</div> <div>text</div>	<div>bookingdetail</div> <div>text</div>	<div>brandname</div> <div>text</div>
28	2111111114	ppp4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
29	2111111113	ppp3	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
30	2111111112	ppp2	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	golden bay
31	2111111150	dude0	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
32	2111111151	dude1	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
33	2111111159	dude9	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
34	2111111158	dude8	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
35	2111111157	dude6	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
36	2111111156	dude6	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
37	2111111155	dude5	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
38	2111111154	dude4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
39	2111111153	dude3	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
40	2111111152	dude2	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	hey boy and girl..
41	2111111129	pdp4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
42	2111111120	pdp0	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
43	2111111121	pdp1	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
44	2111111122	pdp2	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
45	2111111123	pdp3	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
46	2111111124	pdp4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
47	2111111125	pdp4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
48	2111111126	pdp4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
49	2111111127	pdp4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
50	2111111128	pdp4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	shinny beach
51	2111111131	jay1	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
52	2111111130	jay0	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
53	2111111139	jay9	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
54	2111111138	jay8	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
55	2111111137	jay7	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
54	2111111138	jay8	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
55	2111111137	jay7	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
56	2111111136	jay6	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
57	2111111135	jay5	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
58	2111111134	jay4	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
59	2111111133	jay3	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand
60	2111111132	jay2	chod	11/09/03	7758521		123456789	fgtju	customer	[null]	[null]	the big brand

```

1 select sinnumber,firstname,lastname,dateofbirth,person.phonenumber,passwordd,username
2 ,accesslevel,status,dateofregistration,bookingdetail,brandname
3 from ((customer join person using (sinnumber))natural join registers )
4 join hotelbrand using (brandname)
5 order by brandname
6

```

3. Display the details of the cheapest hotel room of all hotel chains.

Query Editor

Query History

```
1 select brandname,hotelid,roomnumber,roomcapacity,specialview,extendable,avaliability,price,amenities,roomtype
2 from room
3 where
4 price = (select min(price)from room )
```

Data Output

Explain

Messages

Notifications

	brandname text	hotelid integer	roomnumber [PK] integer	roomcapacity integer	specialview text	extendable text	avaliability text	price integer	amenities text	roomtype text
1	golden bay	12	204	5	mountain view	no	yes	50	none	supreme
2	golden bay	11	104	5	mountain view	no	booked	50	none	supreme

4. List all the rooms in all hotels in Ottawa and sort them based on the hotel stars and price.

	roomnumber	address	price	roomcapacity	brandname	star
	integer	text	integer	integer	text	integer
1	405	Ottawa	70	4	golden bay	5
2	404	Ottawa	150	5	golden bay	5
3	402	Ottawa	200	2	golden bay	5
4	403	Ottawa	300	3	golden bay	5
5	401	Ottawa	500	1	golden bay	5
6	2605	Ottawa	210	5	shinny beach	4
7	2601	Ottawa	300	2	shinny beach	4
8	2606	Ottawa	330	3	shinny beach	4
9	2603	Ottawa	410	3	shinny beach	4
10	2602	Ottawa	450	1	shinny beach	4
11	2604	Ottawa	520	4	shinny beach	4
12	6304	Ottawa	200	5	hey boy and girls...	3
13	5304	Ottawa	200	5	baby boss hotel	3
14	4302	Ottawa	250	4	championship h...	3
15	6301	Ottawa	300	2	hey boy and girls...	3
16	5301	Ottawa	300	2	baby boss hotel	3
17	5305	Ottawa	300	6	baby boss hotel	3

	roomnumber integer	address text	price integer	roomcapacity integer	brandname text	star integer
17	5305	Ottawa	300	6	baby boss hotel	3
18	6305	Ottawa	300	6	hey boy and girls...	3
19	4301	Ottawa	300	2	championship h...	3
20	4305	Ottawa	350	1	championship h...	3
21	6302	Ottawa	600	3	hey boy and girls...	3
22	5302	Ottawa	600	3	baby boss hotel	3
23	5303	Ottawa	700	4	baby boss hotel	3
24	6303	Ottawa	700	4	hey boy and girls...	3
25	4303	Ottawa	1300	6	championship h...	3
26	4304	Ottawa	3300	3	championship h...	3
27	3302	Ottawa	200	3	the big brand	2
28	6204	Ottawa	200	5	hey boy and girls...	2
29	5204	Ottawa	200	5	baby boss hotel	2
30	4202	Ottawa	250	4	championship h...	2
31	3305	Ottawa	250	6	the big brand	2
32	505	Ottawa	300	5	golden bay	2
33	3301	Ottawa	300	2	the big brand	2

	roomnumber integer	address text	price integer	roomcapacity integer	brandname text	star integer
33	3301	Ottawa	300	2	the big brand	2
34	4201	Ottawa	300	2	championship h...	2
35	5201	Ottawa	300	2	baby boss hote'	2
36	5205	Ottawa	300	6	baby boss hotel	2
37	6201	Ottawa	300	2	hey boy and girls...	2
38	6205	Ottawa	300	6	hey boy and girls...	2
39	503	Ottawa	310	1	golden bay	2
40	4205	Ottawa	350	1	championship h...	2
41	3303	Ottawa	400	4	the big brand	2
42	502	Ottawa	400	2	golden bay	2
43	501	Ottawa	500	3	golden bay	2
44	504	Ottawa	550	6	golden bay	2
45	5202	Ottawa	600	3	baby boss hotel	2
46	3304	Ottawa	600	5	the big brand	2
47	6202	Ottawa	600	3	hey boy and girls...	2
48	6203	Ottawa	700	4	hey boy and girls...	2
49	5203	Ottawa	700	4	baby boss hotel	2

	roomnumber integer	address text	price integer	roomcapacity integer	brandname text	star integer
49	5203	Ottawa	700	4	baby boss hotel	2
50	4203	Ottawa	1300	6	championship h...	2
51	4204	Ottawa	3300	3	championship h...	2
52	104	Ottawa	50	5	golden bay	1
53	103	Ottawa	100	2	golden bay	1
54	102	Ottawa	100	2	golden bay	1
55	2104	Ottawa	120	2	shinny beach	1
56	105	Ottawa	150	4	golden bay	1
57	3102	Ottawa	200	3	the big brand	1
58	5104	Ottawa	200	5	baby boss hotel	1
59	6104	Ottawa	200	5	hey boy and girls...	1
60	3202	Ottawa	200	3	the big brand	1
61	2505	Ottawa	210	5	shinny beach	1
62	2106	Ottawa	230	3	shinny beach	1
63	3105	Ottawa	250	6	the big brand	1
64	4102	Ottawa	250	4	championship h...	1
65	2102	Ottawa	250	1	shinnv beach	1

	roomnumber integer	address text	price integer	roomcapacity integer	brandname text	star integer
65	2102	Ottawa	250	1	shinny beach	1
66	3205	Ottawa	250	6	the big brand	1
67	5105	Ottawa	300	6	baby boss hotel	1
68	5101	Ottawa	300	2	baby boss hotel	1
69	6101	Ottawa	300	2	hey boy and girls...	1
70	3101	Ottawa	300	2	the big brand	1
71	6105	Ottawa	300	6	hey boy and girls...	1
72	4101	Ottawa	300	2	championship h...	1
73	3201	Ottawa	300	2	the big brand	1
74	101	Ottawa	300	2	golden bay	1
75	2506	Ottawa	330	3	shinny beach	1
76	4105	Ottawa	350	1	championship h...	1
77	3103	Ottawa	400	4	the big brand	1
78	3203	Ottawa	400	4	the big brand	1
79	2105	Ottawa	410	6	shinny beach	1
80	2503	Ottawa	410	3	shinny beach	1
81	2502	Ottawa	450	1	shinny beach	1

	roomnumber integer	address text	price integer	roomcapacity integer	brandname text	star integer
77	3103	Ottawa	400	4	the big brand	1
78	3203	Ottawa	400	4	the big brand	1
79	2105	Ottawa	410	6	shinny beach	1
80	2503	Ottawa	410	3	shinny beach	1
81	2502	Ottawa	450	1	shinny beach	1
82	2504	Ottawa	520	4	shinny beach	1
83	3104	Ottawa	600	5	the big brand	1
84	6102	Ottawa	600	3	hey boy and girls...	1
85	5102	Ottawa	600	3	baby boss hotel	1
86	3204	Ottawa	600	5	the big brand	1
87	2501	Ottawa	600	2	shinny beach	1
88	2101	Ottawa	600	3	shinny beach	1
89	2103	Ottawa	610	4	shinny beach	1
90	6103	Ottawa	700	4	hey boy and girls...	1
91	5103	Ottawa	700	4	baby boss hotel	1
92	4103	Ottawa	1300	6	championship h...	1
93	4104	Ottawa	3300	3	championship h...	1

[Query Editor](#) [Query History](#)

```

1 select roomnumber,address,price,roomcapacity,room.brandname,star
2 from hotelchain join room using(hotelid)
3 where address = 'Ottawa'
4 order by star desc, price asc

```

5. List all the details of all rooms rented on the 10th day of a month of your choice. Ensure to insert dates in your table that correspond to that month in order to run your query.

Since our data type of variable date was in text, for this query, we changed it to integer for convenience. The following was our query to change it. We also added a few new orders for our result.

```

1 alter table orderhitory drop column date;
2 alter table orderhistory add column date integer;
3
4 update orderhistory
5 set datee =20010331;
6

```

Data Output

Explain

Messages

Notifications

Read-only column

	roomnumber integer	orderid integer	sinnumber integer	hotelid integer	status text	specialnote text	datee integer	brandname text	hotelid integer
1	2302	20011	211111153	3	not active	[null]	20010310	shinny beach	3
2	201	10006	211111143	12	active	[null]	20210410	golden bay	12
3	2501	10010	211111155	5	active	[null]	20210610	shinny beach	5
4	3501	20010	211111159	114	not active	[null]	20210410	the big brand	114

brandname text	hotelid integer	roomcapacity integer	specialview text	extendable text	avaliability text	price integer	amenities text	roomtype text
shinny beach	3	1	mountian view	no	yes	350	none	supreme
golden bay	12	1	sea view	no	booked	300	none	supreme
shinny beach	5	2	city view	no	booked	600	none	supreme
the big brand	114	2	city view	no	yes	300	none	supreme

Query Editor

Query History

1

select * from orderhistory join room using(roomnumber)where mod(datee,100) =10

6. Update the phone number of a customer.

We chose the first customer in our list to update her phone number, it was 7758521

Query Editor	Query History
1	select sinnumber,firstname,phonenumber from customer join person using(sinnumber)

Data Output	Explain	Messages	Notifications
sinnumber integer	firstname text	phonenumber text	
1	111111111	Helly	7758521

After the update.

Query Editor

Query History

```
1 update person set phonenumber = 1234567890 where sinnumber =111111111;
2 select sinnumber,firstname,phonenumber from customer join person using(sinnumber)
3 where sinnumber =111111111;
```

Data Output

Explain

Messages

Notifications

	<div>sinnumber</div> <div>integer</div>	<div>firstname</div> <div>text</div>	<div>phonenumber</div> <div>text</div>	
1	111111111	Helly	1234567890	

7. Which category hotels (1 star to 5 star) are most preferred by the customers?
 Since we do not have a rating system, we will add in a support rate into our hotel system.

Query Editor	Query History
<pre> 1 alter table hotelchain 2 add column support_rate integer </pre>	

We updated the support rate one by one for every single hotel

<pre> 1 update hotelchain 2 set support_rate = 100 where hotelid = 1; 3 select hotelid from hotelchain; </pre>

Query Editor

Query History

1

2

3

4

5

6

7

8

9

```

select avg(support_rate )as average ,star  from hotelchain group by (star)
order by average desc limit 1

```

Data Output

Explain

Messages

Notifications

	average numeric	star integer
1	68.00000000	1

The result is one star hotels is most welcomed by customers

8. Find the second highest salary from the employee table.

Query Editor

Query History

1

2

3

4

5

6

7

8

```
select salary,firstname,lastname,sinnumber from employee join person using(sinnumber)
order by salary desc
offset 1 limit 1
```

Data Output

Explain

Messages

Notifications

salary

integer

firstname

text

lastname

text

sinnumber

integer

1

13330

jax

chod

111111133

What each person did:

Binxuan Wu 300142301: Write part 7 to design the application. Also, I help my partner to figure out some problems in other parts.

Gary Gao 300124236: Wrote part 4,5,6,8 and implemented part of the report, also created our postgresql database. Adjusted our er diagram and schema diagram. Also, according to the need of building the app, make changes in our database.

Yihao Wang 300059569: With the help of Gary, completed question 8 and inserted the screenshots of the queries for the Q8 in the report. Also, I did some of the employee part of question 7 and helped with the writing of the steps to install our application.