

2021 Fall, Principles of Database Systems, Section B

Project Report

Qiuyu Chen

Net ID: qc872

Yichi Zhang

Net Id : yz8247

December 18, 2021

Content:

[illegible]

1. Summary

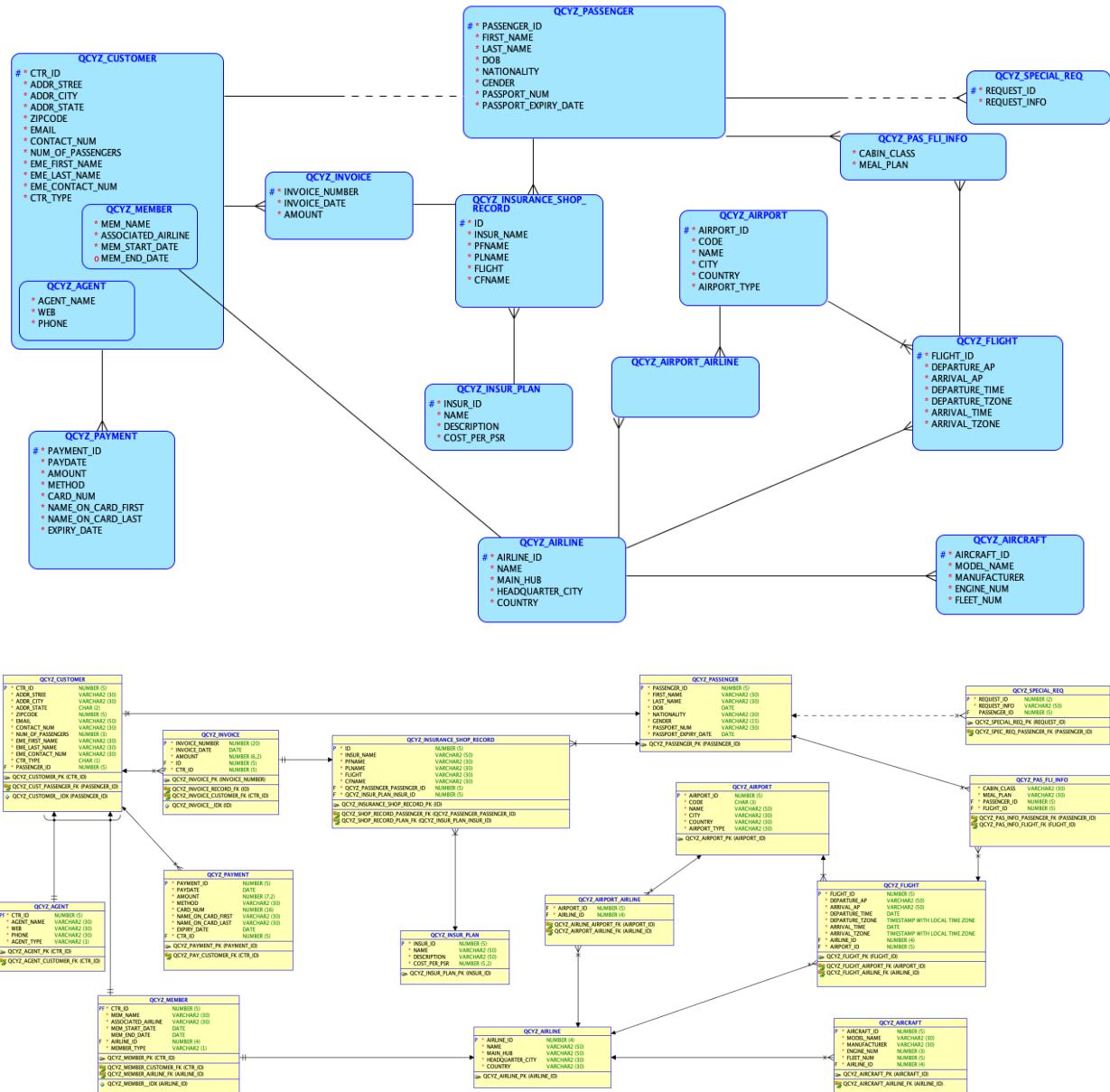
For an insurance company, it has a lot of customers, purchasing its insurance products. It is essential to record information about customers, purchase records, invoices, payments and offer a platform for customers to shop insurances conveniently.

In this project, a web application connected to a back-end database, has been designed and implemented for an insurance company.

This system enables customers to register and login in with their own accounts. After logging in, users can read, edit and delete their personal information. Customers can read the introductions of various insurance products of this company and purchase them. After purchasing, the corresponding invoices will be generated automatically in customers' personal pages. The system also shows the payment condition of each invoice. Customers can read, edit or delete all of their own purchasing records, invoices and payments. Besides supporting customers shopping insurances, this web application also helps the company to manage its business. A manager of the system from the insurance company is accessible to any customer's account information, purchase records, invoices and payment information with this system. Managers can also edit or delete this information or create new records. All important information will be stored in the database through the web user interface automatically. Therefore, this system really helps the company to manage all the data in an efficient way.

Model design:

In order the system can work better in practical application, we've improved our model design of project part I.



2. Software, Programming Language, and Database Used

In this project, we use HTML for front-end development. The framework PHP is used to connect our relational backend MySQL database. The software XAMPP is used for structuring

the testing environment. We also used the integrated development environment software Hbuilder and PhpStorm.

3. DDL file

```

11
12 CREATE TABLE qcyz_agent (
13     ctr_id      NUMBER(5) NOT NULL,
14     agent_name  VARCHAR2(30) NOT NULL,
15     web         VARCHAR2(30) NOT NULL,
16     phone      VARCHAR2(30) NOT NULL,
17     agent_type  VARCHAR2(1) NOT NULL
18 );
19
20 ALTER TABLE qcyz_agent
21     ADD CONSTRAINT ch_inh_qcyz_agent CHECK ( agent_type IN ( 'A' ) );
22
23 COMMENT ON COLUMN qcyz_agent.ctr_id IS
24     'CUSTOMER ID';
25
26 COMMENT ON COLUMN qcyz_agent.agent_name IS
27     'THE NAME OF AGENT';
28
29 COMMENT ON COLUMN qcyz_agent.web IS
30     'THE WEB ADDRESS OF BOOKING AGENT';
31
32 COMMENT ON COLUMN qcyz_agent.phone IS
33     'PHONE NUMBER';
34
35 ALTER TABLE qcyz_agent ADD CONSTRAINT qcyz_agent_pk PRIMARY KEY ( ctr_id );
36
37 CREATE TABLE qcyz_aircraft (
38     aircraft_id  NUMBER(5) NOT NULL,
39     model_name   VARCHAR2(30) NOT NULL,
40     manufacturer VARCHAR2(30) NOT NULL,
41     engine_num   NUMBER(3) NOT NULL,
42     fleet_num    NUMBER(5) NOT NULL,
43     airline_id   NUMBER(4) NOT NULL
44 );
45
46 COMMENT ON COLUMN qcyz_aircraft.aircraft_id IS
47     'AIRCRAFT ID';
48
49 COMMENT ON COLUMN qcyz_aircraft.model_name IS
50     'THE MODEL NAME OF THE AIRCRAFT';
51
52 COMMENT ON COLUMN qcyz_aircraft.manufacturer IS
53     'THE MANUFACTURER OF THE AIRCRAFT';
54
55 COMMENT ON COLUMN qcyz_aircraft.engine_num IS
56     'NUMBER OF ENGINES';
57
58 COMMENT ON COLUMN qcyz_aircraft.fleet_num IS
59     'NUMBER OF FLEETS';
60
61 ALTER TABLE qcyz_aircraft ADD CONSTRAINT qcyz_aircraft_pk PRIMARY KEY ( aircraft_id );
62

```

```
62
63 CREATE TABLE qcyz_airline (
64     airline_id      NUMBER(4) NOT NULL,
65     name             VARCHAR2(50) NOT NULL,
66     main_hub         VARCHAR2(50) NOT NULL,
67     headquarter_city VARCHAR2(30) NOT NULL,
68     country          VARCHAR2(30) NOT NULL
69 );
70
71 COMMENT ON COLUMN qcyz_airline.airline_id IS
72     'AIRLINE ID';
73
74 COMMENT ON COLUMN qcyz_airline.name IS
75     'NAME OF AIRLINE';
76
77 COMMENT ON COLUMN qcyz_airline.main_hub IS
78     'MAIN HUB OF AIRLINE';
79
80 COMMENT ON COLUMN qcyz_airline.headquarter_city IS
81     'HEADQUARTER CITY OF GIVEN AIRLINE';
82
83 COMMENT ON COLUMN qcyz_airline.country IS
84     'COUNTRY OF AIRLINE';
85
86 ALTER TABLE qcyz_airline ADD CONSTRAINT qcyz_airline_pk PRIMARY KEY ( airline_id );
87
88 CREATE TABLE qcyz_airport (
89     airport_id      NUMBER(5) NOT NULL,
90     code            CHAR(3) NOT NULL,
91     name            VARCHAR2(50) NOT NULL,
92     city            VARCHAR2(30) NOT NULL,
93     country         VARCHAR2(30) NOT NULL,
94     airport_type    VARCHAR2(30) NOT NULL
95 );
96
97 COMMENT ON COLUMN qcyz_airport.airport_id IS
98     'AIRPORT ID';
99
100 COMMENT ON COLUMN qcyz_airport.code IS
101     'AIRPORT CODE';
102
103 COMMENT ON COLUMN qcyz_airport.name IS
104     'NAME OF THE AIRPORT';
105
106 COMMENT ON COLUMN qcyz_airport.city IS
107     'CITY OF THE AIRPORT';
108
109 COMMENT ON COLUMN qcyz_airport.country IS
110     'COUNTRY OF THE AIRPORT';
111
112 COMMENT ON COLUMN qcyz_airport.airport_type IS
113     'THE TYPE OF AIRPORT';
114
```

```

114
115 ALTER TABLE qcyz_airport ADD CONSTRAINT qcyz_airport_pk PRIMARY KEY ( airport_id );
116
117 CREATE TABLE qcyz_airport_airline (
118     airport_id NUMBER(5) NOT NULL,
119     airline_id NUMBER(4) NOT NULL
120 );
121
122 CREATE TABLE qcyz_customer (
123     ctr_id          NUMBER(5) NOT NULL,
124     addr_stree      VARCHAR2(30) NOT NULL,
125     addr_city       VARCHAR2(30) NOT NULL,
126     addr_state      CHAR(2) NOT NULL,
127     zipcode         NUMBER(5) NOT NULL,
128     email           VARCHAR2(50) NOT NULL,
129     contact_num     VARCHAR2(30) NOT NULL,
130     num_of_passengers NUMBER(3) NOT NULL,
131     eme_first_name  VARCHAR2(30) NOT NULL,
132     eme_last_name   VARCHAR2(30) NOT NULL,
133     eme_contact_num VARCHAR2(30) NOT NULL,
134     ctr_type        CHAR(1) NOT NULL,
135     passenger_id    NUMBER(5) NOT NULL
136 );
137
138 ALTER TABLE qcyz_customer
139     ADD CONSTRAINT ch_inh_qcyz_customer CHECK ( ctr_type IN ( 'A', 'C', 'M' ) );
140
141 COMMENT ON COLUMN qcyz_customer.ctr_id IS
142     'CUSTOMER ID';
143
144 COMMENT ON COLUMN qcyz_customer.addr_stree IS
145     'THE STREET INFORMATION OF THE ADDRESS OF THE CUSTOMER';
146
147 COMMENT ON COLUMN qcyz_customer.addr_city IS
148     'THE ADDRESS OF CUSTOMER, CITY';
149
150 COMMENT ON COLUMN qcyz_customer.addr_state IS
151     'THE ADDRESS OF CUSTOMER, STATE';
152
153 COMMENT ON COLUMN qcyz_customer.zipcode IS
154     'ZIPCODE';
155
156 COMMENT ON COLUMN qcyz_customer.email IS
157     'EMAIL ADDRESS';
158
159 COMMENT ON COLUMN qcyz_customer.contact_num IS
160     'CONTACT NUMBER WITH COUNTRY CODE';
161
162 COMMENT ON COLUMN qcyz_customer.num_of_passengers IS
163     'THE NUMBER OF PASSENGERS.';
164

```

```

165 COMMENT ON COLUMN qcyz_customer.eme_first_name IS
166     'THE FIRST NAME OF EMERGENCY CONTACT PERSON';
167
168 COMMENT ON COLUMN qcyz_customer.eme_last_name IS
169     'THE LAST NAME OF EMERGENCY CONTACT PERSON';
170
171 COMMENT ON COLUMN qcyz_customer.eme_contact_num IS
172     'THE EMERGENCY CONTACT NUMBER WITH COUNTRY CODE';
173
174 COMMENT ON COLUMN qcyz_customer.ctr_type IS
175     'THE TYPE OF CUSTOMER';
176
177 CREATE UNIQUE INDEX qcyz_customer__idx ON
178     qcyz_customer (
179         passenger_id
180     ASC );
181
182 ALTER TABLE qcyz_customer ADD CONSTRAINT qcyz_customer_pk PRIMARY KEY ( ctr_id );
183
184 CREATE TABLE qcyz_flight (
185     flight_id      NUMBER(5) NOT NULL,
186     departure_ap   VARCHAR2(50) NOT NULL,
187     arrival_ap     VARCHAR2(50) NOT NULL,
188     departure_time DATE NOT NULL,
189     departure_tzone TIMESTAMP WITH LOCAL TIME ZONE NOT NULL,
190     arrival_time   DATE NOT NULL,
191     arrival_tzone  TIMESTAMP WITH LOCAL TIME ZONE NOT NULL,
192     airline_id     NUMBER(4) NOT NULL,
193     airport_id     NUMBER(5) NOT NULL
194 );
195
196 COMMENT ON COLUMN qcyz_flight.flight_id IS
197     'FLIGHT ID';
198
199 COMMENT ON COLUMN qcyz_flight.departure_ap IS
200     'THE DEPARTURE AIRPORT OF THE FLIGHT';
201
202 COMMENT ON COLUMN qcyz_flight.arrival_ap IS
203     'THE ARRIVAL AIRPORT OF THE FLIGHT';
204
205 COMMENT ON COLUMN qcyz_flight.departure_time IS
206     'DEPARTURE TIME';
207
208 COMMENT ON COLUMN qcyz_flight.departure_tzone IS
209     'DEPARTURE TIME ZONE';
210
211 COMMENT ON COLUMN qcyz_flight.arrival_time IS
212     'ARRIVAL TIME OF FLIGHT';
213
214 COMMENT ON COLUMN qcyz_flight.arrival_tzone IS
215     'ARRIVAL TIME ZONE';
216

```



```

217 ALTER TABLE qcyz_flight ADD CONSTRAINT qcyz_flight_pk PRIMARY KEY ( flight_id );
218
219 CREATE TABLE qcyz_insur_plan (
220     insur_id      NUMBER(5) NOT NULL,
221     name          VARCHAR2(50) NOT NULL,
222     description   VARCHAR2(50) NOT NULL,
223     cost_per_psr  NUMBER(5, 2) NOT NULL
224 );
225
226 COMMENT ON COLUMN qcyz_insur_plan.insur_id IS
227     'THE ID OF EACH INSURANCE';
228
229 COMMENT ON COLUMN qcyz_insur_plan.name IS
230     'THE INSURANCE PLAN NAME';
231
232 COMMENT ON COLUMN qcyz_insur_plan.description IS
233     'THE DESCRIPTION OF THE INSURANCE';
234
235 COMMENT ON COLUMN qcyz_insur_plan.cost_per_psr IS
236     'THE COST FOR PER PASSENGER';
237
238 ALTER TABLE qcyz_insur_plan ADD CONSTRAINT qcyz_insur_plan_pk PRIMARY KEY ( insur_id );
239
240 CREATE TABLE qcyz_insurance_shop_record (
241     id              NUMBER(5) NOT NULL,
242     insur_name      VARCHAR2(50) NOT NULL,
243     pfname          VARCHAR2(30) NOT NULL,
244     plname          VARCHAR2(30) NOT NULL,
245     flight          VARCHAR2(30) NOT NULL,
246     cfname          VARCHAR2(30) NOT NULL,
247     qcyz_passenger_passenger_id NUMBER(5) NOT NULL,
248     qcyz_insur_plan_insur_id  NUMBER(5) NOT NULL
249 );
250
251 COMMENT ON COLUMN qcyz_insurance_shop_record.id IS
252     'THE ID OF SHOP RECORD';
253
254 COMMENT ON COLUMN qcyz_insurance_shop_record.insur_name IS
255     'The Name of Insurance bought';
256
257 COMMENT ON COLUMN qcyz_insurance_shop_record.pfname IS
258     'First Name of Passenger of Bought Insurance.';
259
260 COMMENT ON COLUMN qcyz_insurance_shop_record.flight IS
261     'The Flight for Passenger';
262
263 COMMENT ON COLUMN qcyz_insurance_shop_record.cfname IS
264     'Customer First Name';
265
266 ALTER TABLE qcyz_insurance_shop_record ADD CONSTRAINT qcyz_insurance_shop_record_pk PRIMARY KEY (
    id );
267

```

```

268 CREATE TABLE qcyz_invoice (
269     invoice_number NUMBER(20) NOT NULL,
270     invoice_date    DATE NOT NULL,
271     amount          NUMBER(6, 2) NOT NULL,
272     id              NUMBER(5) NOT NULL,
273     ctr_id          NUMBER(5) NOT NULL
274 );
275
276 COMMENT ON COLUMN qcyz_invoice.invoice_number IS
277     'THE INVOICE ID';
278
279 COMMENT ON COLUMN qcyz_invoice.invoice_date IS
280     'THE DATE OF THE INVOICE ISSUED';
281
282 COMMENT ON COLUMN qcyz_invoice.amount IS
283     'THE TOTAL AMOUNT OF THE INVOICE';
284
285 CREATE UNIQUE INDEX qcyz_invoice__idx ON
286     qcyz_invoice (
287         id
288     ASC );
289
290 ALTER TABLE qcyz_invoice ADD CONSTRAINT qcyz_invoice_pk PRIMARY KEY ( invoice_number );
291
292 CREATE TABLE qcyz_member (
293     ctr_id          NUMBER(5) NOT NULL,
294     mem_name        VARCHAR2(30) NOT NULL,
295     associated_airline VARCHAR2(30) NOT NULL,
296     mem_start_date  DATE NOT NULL,
297     mem_end_date    DATE,
298     airline_id      NUMBER(4) NOT NULL,
299     member_type     VARCHAR2(1) NOT NULL
300 );
301
302 ALTER TABLE qcyz_member
303     ADD CONSTRAINT ch_inh_qcyz_member CHECK ( member_type IN ( 'M' ) );
304
305 COMMENT ON COLUMN qcyz_member.ctr_id IS
306     'CUSTOMER ID';
307
308 COMMENT ON COLUMN qcyz_member.mem_name IS
309     'MEMBERSHIP NAME';
310
311 COMMENT ON COLUMN qcyz_member.associated_airline IS
312     'ASOCIATED AIRLINE OF THE MEMBER';
313
314 COMMENT ON COLUMN qcyz_member.mem_start_date IS
315     'START DATE OF MEMBERSHIP';
316
317 COMMENT ON COLUMN qcyz_member.mem_end_date IS
318     'MEMBERSHIP END DATE(IF ANY)';
319

```

```

320 CREATE UNIQUE INDEX qcyz_member__idx ON
321     qcyz_member (
322         airline_id
323     ASC );
324
325 ALTER TABLE qcyz_member ADD CONSTRAINT qcyz_member_pk PRIMARY KEY ( ctr_id );
326
327 CREATE TABLE qcyz_pas_fli_info (
328     cabin_class VARCHAR2(30) NOT NULL,
329     meal_plan   VARCHAR2(30) NOT NULL,
330     passenger_id NUMBER(5) NOT NULL,
331     flight_id   NUMBER(5) NOT NULL
332 );
333
334 COMMENT ON COLUMN qcyz_pas_fli_info.cabin_class IS
335     'CABIN CLASS OF GIVENPASSENGER';
336
337 COMMENT ON COLUMN qcyz_pas_fli_info.meal_plan IS
338     'MEAL PLAN OF GIVEN PASSENGER';
339
340 CREATE TABLE qcyz_passenger (
341     passenger_id      NUMBER(5) NOT NULL,
342     first_name        VARCHAR2(30) NOT NULL,
343     last_name         VARCHAR2(30) NOT NULL,
344     dob               DATE NOT NULL,
345     nationality        VARCHAR2(30) NOT NULL,
346     gender            VARCHAR2(15) NOT NULL,
347     passport_num      VARCHAR2(30) NOT NULL,
348     passport_expiry_date DATE NOT NULL
349 );
350
351 COMMENT ON COLUMN qcyz_passenger.passenger_id IS
352     'ID OF EACH PASSENGER';
353
354 COMMENT ON COLUMN qcyz_passenger.first_name IS
355     'PASSENGER FIRST NAME';
356
357 COMMENT ON COLUMN qcyz_passenger.last_name IS
358     'PASSENGER LAST NAME';
359
360 COMMENT ON COLUMN qcyz_passenger.dob IS
361     'PASSENGER'S DATE OF BIRTH';
362
363 COMMENT ON COLUMN qcyz_passenger.nationality IS
364     'PASSENGER'S NATIONALITY';
365
366 COMMENT ON COLUMN qcyz_passenger.gender IS
367     'GENDER OF PASSENGER';
368
369 COMMENT ON COLUMN qcyz_passenger.passport_num IS
370     'PASSENGER'S PASSPORT NUMBER';
371

```

```

371
372 COMMENT ON COLUMN qcyz_passenger.passport_expiry_date IS
373     'EXPIRY DATE OF PASSENGER'S PASSPORT';
374
375 ALTER TABLE qcyz_passenger ADD CONSTRAINT qcyz_passenger_pk PRIMARY KEY ( passenger_id );
376
377 CREATE TABLE qcyz_payment (
378     payment_id      NUMBER(5) NOT NULL,
379     paydate         DATE NOT NULL,
380     amount          NUMBER(7, 2) NOT NULL,
381     method          VARCHAR2(30) NOT NULL,
382     card_num        NUMBER(16) NOT NULL,
383     name_on_card_first VARCHAR2(30) NOT NULL,
384     name_on_card_last VARCHAR2(30) NOT NULL,
385     expiry_date     DATE NOT NULL,
386     ctr_id          NUMBER(5) NOT NULL
387 );
388
389 COMMENT ON COLUMN qcyz_payment.payment_id IS
390     'THE PAYMENT ID';
391
392 COMMENT ON COLUMN qcyz_payment.paydate IS
393     'THE DATE OF PAYMENT';
394
395 COMMENT ON COLUMN qcyz_payment.amount IS
396     'THE AMOUNT PAID';
397
398 COMMENT ON COLUMN qcyz_payment.method IS
399     'THE PAYMENT METHOD, CREDIT CARD OR DEBIT CARD';
400
401 COMMENT ON COLUMN qcyz_payment.card_num IS
402     'THE CARD NUMBER USED TO PAY';
403
404 COMMENT ON COLUMN qcyz_payment.name_on_card_first IS
405     'THE FIRST NAME ON CARD';
406
407 COMMENT ON COLUMN qcyz_payment.name_on_card_last IS
408     'THE LAST NAME ON CARD';
409
410 COMMENT ON COLUMN qcyz_payment.expiry_date IS
411     'THE EXPIRE DATE OF GIVEN CARD';
412
413 ALTER TABLE qcyz_payment ADD CONSTRAINT qcyz_payment_pk PRIMARY KEY ( payment_id );
414
415 CREATE TABLE qcyz_special_req (
416     request_id      NUMBER(2) NOT NULL,
417     request_info    VARCHAR2(50) NOT NULL,
418     passenger_id    NUMBER(5)
419 );
420
421 COMMENT ON COLUMN qcyz_special_req.request_id IS
422     'REQUEST ID';

```

```

424 COMMENT ON COLUMN qcyz_special_req.request_info IS
425     'INFORMATION OF EACH REQUEST';
426
427 ALTER TABLE qcyz_special_req ADD CONSTRAINT qcyz_special_req_pk PRIMARY KEY ( request_id );
428
429 ALTER TABLE qcyz_agent
430     ADD CONSTRAINT qcyz_agent_customer_fk FOREIGN KEY ( ctr_id )
431     REFERENCES qcyz_customer ( ctr_id );
432
433 ALTER TABLE qcyz_aircraft
434     ADD CONSTRAINT qcyz_aircraft_airline_fk FOREIGN KEY ( airline_id )
435     REFERENCES qcyz_airline ( airline_id )
436     ON DELETE CASCADE;
437
438 ALTER TABLE qcyz_airport_airline
439     ADD CONSTRAINT qcyz_airline_airport_fk FOREIGN KEY ( airport_id )
440     REFERENCES qcyz_airport ( airport_id )
441     ON DELETE CASCADE;
442
443 ALTER TABLE qcyz_airport_airline
444     ADD CONSTRAINT qcyz_airport_airline_fk FOREIGN KEY ( airline_id )
445     REFERENCES qcyz_airline ( airline_id )
446     ON DELETE CASCADE;
447
448 ALTER TABLE qcyz_customer
449     ADD CONSTRAINT qcyz_cust_passenger_fk FOREIGN KEY ( passenger_id )
450     REFERENCES qcyz_passenger ( passenger_id )
451     ON DELETE CASCADE;
452
453 ALTER TABLE qcyz_flight
454     ADD CONSTRAINT qcyz_flight_airline_fk FOREIGN KEY ( airline_id )
455     REFERENCES qcyz_airline ( airline_id )
456     ON DELETE CASCADE;
457
458 ALTER TABLE qcyz_flight
459     ADD CONSTRAINT qcyz_flight_airport_fk FOREIGN KEY ( airport_id )
460     REFERENCES qcyz_airport ( airport_id );
461
462 ALTER TABLE qcyz_invoice
463     ADD CONSTRAINT qcyz_invoice_customer_fk FOREIGN KEY ( ctr_id )
464     REFERENCES qcyz_customer ( ctr_id )
465     ON DELETE CASCADE;
466
467 ALTER TABLE qcyz_invoice
468     ADD CONSTRAINT qcyz_invoice_record_fk FOREIGN KEY ( id )
469     REFERENCES qcyz_insurance_shop_record ( id );
470
471 ALTER TABLE qcyz_member
472     ADD CONSTRAINT qcyz_member_airline_fk FOREIGN KEY ( airline_id )
473     REFERENCES qcyz_airline ( airline_id );
474

```

```
475 ALTER TABLE qcyz_member
476     ADD CONSTRAINT qcyz_member_customer_fk FOREIGN KEY ( ctr_id )
477     REFERENCES qcyz_customer ( ctr_id );
478
479 ALTER TABLE qcyz_pas_fli_info
480     ADD CONSTRAINT qcyz_pas_info_flight_fk FOREIGN KEY ( flight_id )
481     REFERENCES qcyz_flight ( flight_id )
482     ON DELETE CASCADE;
483
484 ALTER TABLE qcyz_pas_fli_info
485     ADD CONSTRAINT qcyz_pas_info_passenger_fk FOREIGN KEY ( passenger_id )
486     REFERENCES qcyz_passenger ( passenger_id )
487     ON DELETE CASCADE;
488
489 ALTER TABLE qcyz_payment
490     ADD CONSTRAINT qcyz_pay_customer_fk FOREIGN KEY ( ctr_id )
491     REFERENCES qcyz_customer ( ctr_id )
492     ON DELETE CASCADE;
493
494 ALTER TABLE qcyz_insurance_shop_record
495     ADD CONSTRAINT qcyz_shop_record_passenger_fk FOREIGN KEY ( qcyz_passenger_passenger_id )
496     REFERENCES qcyz_passenger ( passenger_id );
497
498 ALTER TABLE qcyz_insurance_shop_record
499     ADD CONSTRAINT qcyz_shop_record_plan_fk FOREIGN KEY ( qcyz_insur_plan_insur_id )
500     REFERENCES qcyz_insur_plan ( insur_id );
501
502 ALTER TABLE qcyz_special_req
503     ADD CONSTRAINT qcyz_spec_req_passenger_fk FOREIGN KEY ( passenger_id )
504     REFERENCES qcyz_passenger ( passenger_id )
505     ON DELETE CASCADE;
506
```



```

507 CREATE OR REPLACE TRIGGER arc_fkarc_2_qcyz_agent BEFORE
508     INSERT OR UPDATE OF ctr_id ON qcyz_agent
509     FOR EACH ROW
510 DECLARE
511     d CHAR(1);
512 BEGIN
513     SELECT
514         a.ctr_type
515     INTO d
516     FROM
517         qcyz_customer a
518     WHERE
519         a.ctr_id = :new.ctr_id;
520
521     IF ( d IS NULL OR d <> 'A' ) THEN
522         raise_application_error(-20223,
523             'FK QCYZ_AGENT_CUSTOMER_FK in Table QCYZ_AGENT violates Arc
              constraint on Table QCYZ_CUSTOMER - discriminator column
              CTR_TYPE doesn't have value 'A''');
524     END IF;
525
526 EXCEPTION
527     WHEN no_data_found THEN
528         NULL;
529     WHEN OTHERS THEN
530         RAISE;
531 END;
532 /
533
534 CREATE OR REPLACE TRIGGER arc_fkarc_2_qcyz_member BEFORE
535     INSERT OR UPDATE OF ctr_id ON qcyz_member
536     FOR EACH ROW
537 DECLARE
538     d CHAR(1);
539 BEGIN
540     SELECT
541         a.ctr_type
542     INTO d
543     FROM
544         qcyz_customer a
545     WHERE
546         a.ctr_id = :new.ctr_id;
547
548     IF ( d IS NULL OR d <> 'M' ) THEN
549         raise_application_error(-20223,
550             'FK QCYZ_MEMBER_CUSTOMER_FK in Table QCYZ_MEMBER violates Arc
              constraint on Table QCYZ_CUSTOMER - discriminator column
              CTR_TYPE doesn't have value 'M''');
551     END IF;

```

```

553 EXCEPTION
554     WHEN no_data_found THEN
555         NULL;
556     WHEN OTHERS THEN
557         RAISE;
558 END;
559 /
560
561
562
563 -- Oracle SQL Developer Data Modeler Summary Report:
564 --
565 -- CREATE TABLE          15
566 -- CREATE INDEX            3
567 -- ALTER TABLE           33

```

4. List of tables

```
mysql> show tables;
+-----+
| Tables_in_insurance2 |
+-----+
| agent                |
| customer             |
| insurance            |
| invoice              |
| manager              |
| member               |
| passenger            |
| password_hash        |
| payment              |
| plan                 |
+-----+
```

Passenger Table:

passenger_id	first_name	last_name	DOB	nationality	gender	passport_num	passport_expiry_date
2	aa	aa	1990-01-01	aa	m	S12345678	2023-12-01
8	Sherlock	Holmes	1974-06-07	UK	Male	U12345678	2025-06-19
22	Kim	Chi	2000-11-23	Korea	Female	K12345678	2023-10-25
23	Baguette	France	1988-01-12	France	Male	F12345678	2023-10-17
26	Johnny	English	1990-01-01	UK	Male	R12345678	2022-07-06
30	Password	Test	2000-10-10	USA	Male	S66666666	2022-10-01
31	Yichi	Zhang	1999-02-18	China	Male	E22456565	2023-10-18
32	Qiuyu	Chen	1997-09-22	China	Male	E12345678	2024-01-01
33	Marry	Jane	1993-11-11	USA	Female	S45822145	2022-10-18
34	Woman	Female	2000-03-08	UK	Female	W12345678	2025-03-08

10 rows in set (0.01 sec)

Customer Table:

```
mysql> select * from customer;
```

customer_id	first_name	last_name	password	nationality	gender	addr_street	addr_city	addr_state	zipcode	email	contact_num	eme_first_name	eme_last_name	eme_contact_num	ctr_type	passenger_id
14	Password	Test	password	USA	Male	6 MetroTech Center	Brooklyn	NV	11201	ptest@nyu.edu	+16111111111	Test	Hash	+16222222222	C	30
15	Yichi	Zhang	Cq970922	China	Male	66 Street	New York	NV	11201	yich@nyu.edu	+16456786584	Zhang	Yichi	+16543765489	M	31
16	Qiuyu	Chen	Qc971224	China	Male	1565 W 11th Street	Brooklyn	NV	11204	qc972@nyu.edu	+16175131134	Hi	Yan	+16019279182942	C	32
17	Marry	Jane	00000000	USA	Female	Spider Street	Queens	NV	11439	MJ678@queens.edu	+16536536534	Peter	Parker	+16666666666	M	33
18	Woman	Female	123456	UK	Female	11 Street	Boston	MA	07201	Female12@yahoo.com	16176176177	Man	Male	+16516516511	A	34

5 rows in set (0.00 sec)

Member Table:


```
mysql> select * from member;
```

mem_name	associated_airline	mem_start_date	mem_end_date	customer_id
Delta Sky Club	Delta	2021-01-01	2022-01-01	15
TrueBlue	JetBlue	2019-02-19	2024-07-18	17

```
2 rows in set (0.01 sec)
```

Agent Table:

```
mysql> select * from agent;
```

agent_name	web	phone	customer_id
Booking	Booking.com	212-419-2618	18

```
1 row in set (0.01 sec)
```

Password Hash Table:

```
mysql> select * from password_hash;
```

customer_id	password	hash
14	password	\$2y\$10\$Lgb9LrxoXlCPG/q1hmcWIuORuzdrd01ZNrv.8IaWTVkc8zQX9b2MK
15	Cqy970922	\$2y\$10\$f..HGA0aqqm5qdzwhSqrCOKa76GCIIFrOqB0u/1lSvlrazuXXsKD
16	Qwerty1234	\$2y\$10\$ysMplHX3MWmE07WST7mZT05jZM8kOST3FPsmjt9azeY66YwVwYLeK
17	00000000	\$2y\$10\$BBdC/i2FKTdlIapUD01E2eSoEHVmC5iGRTfKE7XxhpLtTDZOHica
18	123456	\$2y\$10\$1wiJDdxhobKiSXizSyyPy.YGotv.iONl5nPxwfyVG/XbUyZG7MF.S

```
5 rows in set (0.00 sec)
```

Insurance Plan Table:

```
mysql> select * from plan;
```

id	name	descrip	cost
1	Trip Cancellation	Insurance for Trip Cancellation	5
2	Trip Interruption	Insurance for Trip Interruption	3
3	Medical Insurance	Insurance for Medical treatment	4
4	Baggage Insurance	Insurance for Baggage	10
5	Accidental Death Insurance	Insurance for Accidental Death	8
6	All-inclusive Insurance	Insurance for everything	20

```
6 rows in set (0.00 sec)
```

Insurance Shop Record Table:

```
mysql> select * from insurance;
```

id	insur_name	pfname	plname	flight	cfname	passenger_id	insurance_id
31	Baggage Insurance	Yichi	Zhang	AMD 122	Yichi	31	2
32	Accidental Death Insurance	Yichi	Zhang	AMD 122	Yichi	31	5
33	Medical Insurance	Yichi	Zhang	AMD 122	Yichi	31	3
34	Trip Interruption	Yichi	Zhang	AMD 122	Yichi	31	2
35	All-inclusive Insurance	Qiuyu	Chen	ABC 282	Qiuyu	32	6

5 rows in set (0.00 sec)

Invoice Table:

```
mysql> select * from invoice;
```

invoice_num	invoice_date	invoice_amount	customer_id	record_id
100000019	2021-12-16	3.00	15	31
100000020	2021-12-17	8.00	15	32
100000021	2021-12-17	4.00	15	33
100000022	2021-12-17	3.00	15	34
100000023	2021-12-18	20.00	16	35

5 rows in set (0.00 sec)

Payment Table:

```
mysql> select * from payment;
```

payment_id	pay_date	amount	method	card_num	name_on_card_first	name_on_card_last	expiry_date	customer_id
100000037	2021-12-17	15.50	Credit	7412589632587412	Yichi	Zhang	2024-06-01	15
100000038	2021-12-16	20.00	Credit	1234567812345678	Qiuyu	Chen	2023-05-24	16

2 rows in set (0.00 sec)

Manager Table:

```
mysql> select * from manager;
```

id	username	password
1	manager	manager

1 row in set (0.01 sec)

5. Screenshots of sessions

The main menu:

Welcome

[Menu](#)

Main Menu

- [Register](#)
- [Login](#)
- [Insurance Product Introduction](#)
- [Manager Login](#)

© 2021 SAFE FLY MANAGEMENT EXCELLENCE

The register page:

Welcome

[Menu](#)

Create customer

First Name:

Last Name:

Date of Birth:

Set Password:

Nationality:

Gender:

Passport Number:

Passport Expiry Date:

Address:

City:

State:

Zipcode:

Email:

Contact Number:

Emergency First Name:

Emergency Last Name:

Emergency Contact Number:

Member ? : ☐

Agent ? : ☐

© 2021 SAFE FLY MANAGEMENT EXCELLENCE

The insurance plan introduction page:

Welcome

[Menu](#)

Plans

plan_id	name	description	cost per person
1	Trip Cancellation	Insurance for Trip Cancellation	5
2	Trip Interruption	Insurance for Trip Interruption	3
3	Medical Insurance	Insurance for Medical treatment	4
4	Baggage Insurance	Insurance for Baggage	10
5	Accidental Death Insurance	Insurance for Accidental Death	8
6	All-inclusive Insurance	Insurance for everything	20

- [Shop Insurance](#)
- [Not A Customer? Register](#)

© 2021 SAFE FLY MANAGEMENT EXCELLENCE

The user customer page :

Welcome

[Menu](#)

Customer: Yichi Zhang

Nationality: China
Gender: Male
Address: 66 Street
City: New York
State: NY
Zipcode: 11201
Email: yichi@nyu.edu
Contact Number: +16456786584
Emergency Contact Name: Zhang Yichi
Emergency Contact Number: +16548765489

- [Edit](#)
- [Delete](#)
- [Add Passenger](#)

insurance name	passenger first name	passenger last name	flight		
Baggage Insurance	Yichi	Zhang	AMD 122	Edit	Delete
Accidental Death Insurance	Yichi	Zhang	AMD 122	Edit	Delete
Medical Insurance	Yichi	Zhang	AMD 122	Edit	Delete
Trip Interruption	Yichi	Zhang	AMD 122	Edit	Delete

- [Shop Insurance](#)

invoice number	invoice date	amount	
100000019	2021-12-16	3.00	View
100000020	2021-12-17	8.00	View
100000021	2021-12-17	4.00	View
100000022	2021-12-17	3.00	View

Payment Due: \$2.5

- [Make a Payment](#)

© 2021 SAFE FLY MANAGEMENT EXCELLENCE

A customer can view his or her own personal information, purchase records, invoices and payment balance on this page. Managers are accessible to the customer page of each customer.

The manager page:

Welcome

[Menu](#)

Manager Menu

- [View Customers](#)
- [View Passengers](#)
- [Shopping Records](#)
- [View Invoice](#)
- [Payment Record](#)

© 2021 SAFE FLY MANAGEMENT EXCELLENCE

Welcome

[Menu](#)

Customers

customer_id	first_name	last_name			
14	Password	Test	View	Edit	Delete
15	Yichi	Zhang	View	Edit	Delete
16	Qiuyu	Chen	View	Edit	Delete
17	Marry	Jane	View	Edit	Delete
18	Woman	Female	View	Edit	Delete

© 2021 SAFE FLY MANAGEMENT EXCELLENCE

6. Security features

6.1 Prevent SQL injection attacks

To prevent SQL injection attacks, we've built a validation function for user inputs.

Example: `validate_customer()` check validation of user's input while registering:

```
function validate_customer($customer): array
{
    $errors = [];

    // CHECK IF INPUT IS NULL
    if(is_blank($customer['first_name'])) {
        $errors[] = "First name cannot be blank.";
    }

    // menu_name
    if(is_blank($customer['last_name'])) {
        $errors[] = "Last name cannot be blank.";
    }

    if(is_blank($customer['password'])) {
        $errors[] = "Password cannot be blank.";
    }

    if(is_blank($customer['nationality'])) {
        $errors[] = "Nationality cannot be blank.";
    }

    if(is_blank($customer['gender'])) {
        $errors[] = "Gender cannot be blank.";
    }

    if(is_blank($customer['addr_stree'])) {
        $errors[] = "Address street cannot be blank.";
    }

    if(is_blank($customer['addr_city'])) {
        $errors[] = "Address city cannot be blank.";
    }

    if(is_blank($customer['addr_state'])) {
        $errors[] = "Address state cannot be blank.";
    }
}
```

```

    if(is_blank($customer['zipcode'])) {
        $errors[] = " Zipcode cannot be blank.";
    }

    // CHECK IF VALID ZIPCODE
    $zipcode_int = (int) $customer['zipcode'];
    if($zipcode_int < 10000 or $zipcode_int > 999999) {
        $errors[] = "Invalid zipcode.";
    }

    if(is_blank($customer['email'])) {
        $errors[] = "Email cannot be blank.";
    }

    // CHECK IF INPUT EMAIL IS VALID;
    if(!has_valid_email_format($customer['email'])) {
        $errors[] = "Invalid Email address.";
    }

    if(is_blank($customer['contact_num'])) {
        $errors[] = "Contact number cannot be blank.";
    }

    if(is_blank($customer['eme_first_name'])) {
        $errors[] = "Emergency first name cannot be blank.";
    }

    if(is_blank($customer['eme_last_name'])) {
        $errors[] = "Emergency last name cannot be blank.";
    }

    if(is_blank($customer['eme_contact_num'])) {
        $errors[] = "Emergency contact number cannot be blank.";
    }

    return $errors;
}



```

If the user makes a invalid input, the system would print out the message to let the user fix mistakes:

Create customer

Please fix the following errors:

- Nationality cannot be blank.
- Address street cannot be blank.
- Address city cannot be blank.
- Invalid Email address.

First Name:	<input type="text" value="Naruto"/>
Last Name:	<input type="text" value="Uzumaki"/>
Date of Birth:	<input type="text" value="10/10/2000"/> 
Set Password:	<input type="text" value="naruto"/>
Nationality:	<input type="text"/>
Gender:	<input type="text" value="Male"/>
Passport Number:	<input type="text" value="J12345678"/>
Passport Expiry Date:	<input type="text" value="07/13/2023"/> 
Address:	<input type="text"/>
City:	<input type="text"/>
State:	<input type="text" value="KO"/>
Zipcode:	<input type="text" value="24551"/>
Email:	<input type="text" value="naruto"/>
Contact Number:	<input type="text" value="+15459512354"/>
Emergency First Name:	<input type="text" value="Sasuke"/>
Emergency Last Name:	<input type="text" value="Uchiha"/>
Emergency Contact Number:	<input type="text" value="+17777777777"/>
Member ?:	<input type="checkbox"/>
Agent ?:	<input type="checkbox"/>

Create customer

6.2 Sensitive Data Encryption

All sensitive data should be encrypted before being stored in the database.

In this project, we encrypted users' passwords. We use an algorithm called PASSWORD_BCRYPT supported by PHP to hash users' passwords into a new string, which is the encrypted passwords and store the hashed string into a table called password_hash.

```
$hash = password_hash($customer['password'], algo: PASSWORD_BCRYPT);
$new_id = mysqli_insert_id($db);
```

The example hash after password_hash():

customer_id	password	hash
14	password	\$2y\$10\$Lgb9LrxoXlCPG/q1hmcWIuORuzdrd01ZNrv.8IaWTVkc8zQX9b2MK

When a registered customer is logging in, the system would verify the input password and the hash stored in the database with the specific customer id of the user customer.

```
$hash = get_hash($customer1);
if (password_verify($password, $hash)) {
    $customer2 = find_customer_by_password($password);
```

Only the input password passes the verification, the user can log in or the system would jump out a “Wrong Password” notification.

Wrong password!

7. Harvest and Reflection

We really learned a lot through this project.

The most important lesson we've learned is time management. We started to work on this project lately. At the same time, the final exams arrangement of other courses we are taking in

this semester changed unexpectedly. As a result, we failed to manage our time well and could not finish all the functions we expected to realize in this project timely.

We also learned how to coordinate project work with a teammate remotely. We finished most of the project work with communication online. This work mode is efficient, saving a lot of time for traveling to meet each other.

This project is the first time for us to build a web application by ourselves. During the project, we applied the knowledge taught in class into practical usage and studied web front-end and back-end by ourselves, which made us gain a lot.

8. Business analysis

Q1) Table joins with at least 3 tables in join

```
select c.customer_id, d.first_name, d.last_name,
c.invoice_num,c.invoice_amount, c.insur_name,c.pfname,c.plname
from
(select
a.insur_name,a.pfname,a.plname,b.invoice_num,b.invoice_amount,b.customer_id
from insurance a join invoice b
on a.id = b.record_id) c
join customer d on c.customer_id=d.customer_id
order by customer_id;
```

```
mysql> select c.customer_id, d.first_name, d.last_name, c.invoice_num, c.invoice_amount, c.insur_name, c.pfname, c.plname
-> from
-> (select a.insur_name, a.pfname, a.plname, b.invoice_num, b.invoice_amount, b.customer_id
-> from insurance a join invoice b
-> on a.id = b.record_id) c
-> join customer d on c.customer_id=d.customer_id
-> order by customer_id;
```

customer_id	first_name	last_name	invoice_num	invoice_amount	insur_name	pfname	plname
15	Yichi	Zhang	100000019	3.00	Baggage Insurance	Yichi	Zhang
15	Yichi	Zhang	100000020	8.00	Accidental Death Insurance	Yichi	Zhang
15	Yichi	Zhang	100000021	4.00	Medical Insurance	Yichi	Zhang
15	Yichi	Zhang	100000022	3.00	Trip Interruption	Yichi	Zhang
16	Qiuyu	Chen	100000023	20.00	All-inclusive Insurance	Qiuyu	Chen

5 rows in set (0.01 sec)

This query is used to show all customers, their shop records and the invoice information related to each purchase.

Q2) Multi-row subquery

```
select invoice_num, invoice_date
```

```
from invoice
```

```
where invoice_amount >= ALL(SELECT avg(invoice_amount) FROM invoice GROUP BY
customer_id);
```

```
mysql> select invoice_num, invoice_date
-> from invoice
-> where invoice_amount >= ALL(SELECT avg(invoice_amount) FROM invoice GROUP BY customer_id);
```

invoice_num	invoice_date
100000023	2021-12-18

1 row in set (0.01 sec)

This query is used to show the invoices whose amount is larger or equal to the highest average invoice amount of any specific customer.

Q3) Correlated subquery

To do the query, we add another record into the insurance table. This version of the insurance table is not used in other queries.

```
mysql> select * from insurance;
```

id	insur_name	pfname	plname	flight	cfname	passenger_id	insurance_id
31	Baggage Insurance	Yichi	Zhang	AMD 122	Yichi	31	2
32	Accidental Death Insurance	Yichi	Zhang	AMD 122	Yichi	31	5
33	Medical Insurance	Yichi	Zhang	AMD 122	Yichi	31	3
34	Trip Interruption	Yichi	Zhang	AMD 122	Yichi	31	2
35	All-inclusive Insurance	Qiuyu	Chen	ABC 282	Qiuyu	32	6
36	Medical Insurance	Johnny	English	MRB 111	Qiuyu	26	3

```
select pfname, plname from insurance
where pfname not in
(select first_name from customer where
customer.passenger_id = insurance.passenger_id);
```

```
mysql> select pfname, plname from insurance
-> where pfname not in
-> (select first_name from customer where
-> customer.passenger_id = insurance.passenger_id);
```

pfname	plname
Johnny	English

1 row in set (0.01 sec)

This query is used to show the name of the passenger(s) who has(have) insurance(s) and is(are) not customer(s) in the database.

Q4) SET operator query

```
SELECT id, insur_name, pfname, plname
FROM insurance WHERE passenger_id='31'
UNION
SELECT id, insur_name, pfname, plname
FROM insurance
WHERE id='2'
ORDER BY id;
```

```
mysql> SELECT id, insur_name, pfname, plname FROM insurance WHERE passenger_id='31' UNION SELECT id, insur_name, pfname, plname FROM insurance WHERE id='2' ORDER BY id;
```

id	insur_name	pfname	plname
31	Baggage Insurance	Yichi	Zhang
32	Accidental Death Insurance	Yichi	Zhang
33	Medical Insurance	Yichi	Zhang
34	Trip Interruption	Yichi	Zhang

4 rows in set (0.00 sec)

This query is used to view the shop insurance record of the passenger with passenger_id is 31 and the insurance with insurance_id=2 without duplicate record.

Q5) Query with in line view or WITH clause

```
with TOTAL as (select customer_id,sum(invoice_amount) totalamount from invoice
group by customer_id)

select a.customer_id, a.first_name, a.last_name, b.totalamount

from customer a, TOTAL b

where a.customer_id=b.customer_id;
```

```
mysql> with TOTAL as (select customer_id,sum(invoice_amount) totalamount from invoice group by customer_id)
-> select a.customer_id, a.first_name, a.last_name, b.totalamount
-> from customer a, TOTAL b
-> where a.customer_id=b.customer_id;
+-----+-----+-----+-----+
| customer_id | first_name | last_name | totalamount |
+-----+-----+-----+-----+
| 15 | Yichi | Zhang | 18.00 |
| 16 | Qiuyu | Chen | 20.00 |
+-----+-----+-----+-----+
2 rows in set (0.03 sec)
```

This query is used to view the total purchase amount of each customer.

Q6) TOP-N query

```
select invoice_num, invoice_amount

from invoice

order by invoice_amount DESC

limit 3;
```

```
mysql> Select invoice_num, invoice_amount
-> from invoice
-> order by invoice_amount DESC
-> Limit 3;
+-----+-----+
| invoice_num | invoice_amount |
+-----+-----+
| 100000023 | 20.00 |
| 100000020 | 8.00 |
| 100000021 | 4.00 |
+-----+-----+
3 rows in set (0.00 sec)
```

This query is used to view the invoices with the top 3 amounts.