

# CSDS341\_Project\_accessingMySQL

Luke Zhang (rxz330)

2022-07-21

## Contents

<b>Preliminaries</b>	<b>1</b>
Load Packages . . . . .	1
Establish Connection to mySQL . . . . .	1
<b>Queries &amp; SQL</b>	<b>2</b>
Easy Queries & SQL . . . . .	2

## Preliminaries

### Load Packages

The connection between R and MySQL will require the following packages.

```
library(RMySQL)
library(RODBC)
library(knitr)
library(tidyverse)
```

### Establish Connection to mySQL

Before starting query the database, we need to set up a link about our connection. We stored our connection in variable `sql_link`.

```
# Establish Connection to MySQL
sql_link <- dbConnect(RMySQL::MySQL(),
                      dbname='airline',
                      host='localhost',
                      port=3306,
                      user='root',
                      password='Lukezrh@010408')

# List All the Tables in the Schema
dbListTables(sql_link)
```

```
## [1] "aeroplanes_belong" "airlines" "airports"
## [4] "assign" "crew" "flights_ariv_dept"
## [7] "hub" "schedules" "serve"
## [10] "tickets_book_for" "travelers"
```

## Queries & SQL

### Easy Queries & SQL

Query 01: Find the first name(`first_name`), last name(`last_name`), date of birth(`dob`), and the credits(`credits`) of all male travelers who are U.S. citizen.

```
sql_01 <- "SELECT first_name, last_name, dob, credits
          FROM   travelers
          WHERE  citizenship = 'US'
          AND
          gender = 'M'
          AND
          dob < '1985-01-01'"
```

```
result_01 <- dbSendQuery(sql_link, sql_01)
```

```
output_01 <- fetch(result_01)
```

```
output_01 %>% kable()
```

first_name	last_name	dob	credits
Adrian	Rosales	1982-05-05	6269
Zachary	Shipley	1982-09-17	1776
Sean	Kim	1981-12-07	4915
Jose	Villatoro	1981-11-13	1645
Daniel	Ramirez	1968-10-14	6937
Turki	al-Ahsan	1965-01-01	8458
Andrew	Macy	1968-09-20	9060
Muhaajir	el-Noorani	1949-04-04	5074
Sterling	Kirkland	1961-03-06	4279
Darius	Stukes	1952-03-30	9913
Deven	Granderson	1951-04-25	5676

Query 02: Find the passport number(`passport_no`) and first name(`first_name`) of all Chinese Traveler who has already booked at least one Plane Ticket. The result table should also include `citizenship` as a check.

```
sql_02 <- "SELECT passport_no, first_name, citizenship
          FROM   Travelers
          WHERE  (Citizenship = 'CN')
          AND
          (user_id IN (SELECT   traveler_id
                        FROM     Tickets_book_for
                        GROUP BY  traveler_id
                        HAVING    count(ticket_id) >= 1))"
```

```
result_02 <- dbSendQuery(sql_link, sql_02)
```

```
output_02 <- fetch(result_02)
```

```
output_02 %>% kable()
```

passport_no	first_name	citizenship
UU836955458	Jesus	CN
IB430482154	Brandon	CN
OQ380375182	Haseena	CN
LM812170185	John	CN
EB949426590	Paul	CN
TL625875287	Jordan	CN
NJ572174743	Natelle Rae	CN
TT276365122	Brandon	CN
GA510923640	Fatima	CN
RX309492405	Sidney	CN
KE075171969	Kristina	CN
BH026711943	Keisha	CN
CB965376246	Haley	CN
OC970861821	Mariah	CN
IC641671105	Lauren	CN
FX285795793	Daisy	CN
VB903190894	Sierra	CN
BQ269116750	Timothy	CN
FM884378913	Paris	CN
LV473086926	Fat'hi	CN
DQ891369571	Shajee'a	CN
PI055048751	Macennia	CN
LM354095793	Shaakir	CN
BX613479733	Adham	CN

Query 02: Find the flight number of all flights that operated by China Eastern Airlines and has a capacity of 30 seats in first class cabin. % query starts here

```
sql_03 <- "SELECT flight_no, dept_iata_code, ariv_iata_code
FROM   Flights_ariv_dept NATURAL JOIN Aeroplanes_belong
WHERE  (Aeroplanes_belong.fir_capacity = 30)
AND    (regis_no IN (
        SELECT regis_no
        FROM   Aeroplanes_belong NATURAL JOIN Airlines
        WHERE  Airlines.company_name = 'China Eastern Airlines'))"
```

```
result_03 <- dbSendQuery(sql_link, sql_03)
```

```
output_03 <- fetch(result_03)
```

```
output_03 %>% kable()
```

flight_no	dept_iata_code	ariv_iata_code
MU1332	ORF	SJC
MU3212	VBG	KEF

flight_no	dept_iata_code	ariv_iata_code
MU3762	GSP	CEB
MU470	GSP	ISE
MU5879	NNG	BDL
MU6141	DAY	RND
MU7451	FOC	AFW
MU8560	KWL	CAE
MU8990	PHL	SDQ
MU9025	HSV	OTP
MU9871	CAE	ITM
MU2680	ORF	DJJ
MU4422	SHE	AYT
MU4868	VBG	ADA
MU4936	DAY	SJC
MU636	LIT	CRK
MU7946	TYN	ERI
MU8048	RDU	TSE
MU8188	GSP	BLA
MU8497	LIT	MCI
MU8723	MBS	DCA
MU9773	PHL	TPA