**Jeju-air Database**

**MYSQL Homework 3**

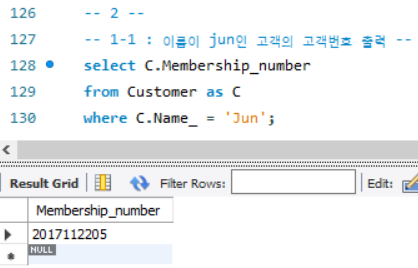
**2017112173 박동근**

**2017112205 신희준**

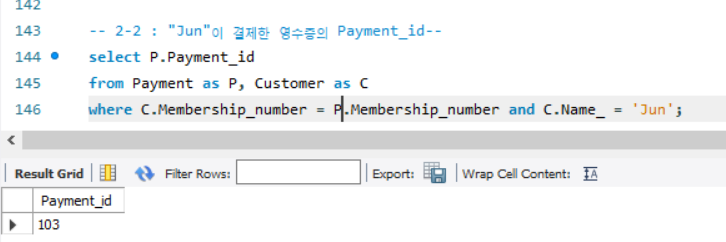
**1. Select four queries from the SQL queries in HW2. Any two queries can't belong to the same category(I - VI). All the queries should include one or more host variables.**

**1-1.show the four queries from HW2 you selected, including the host variables.**

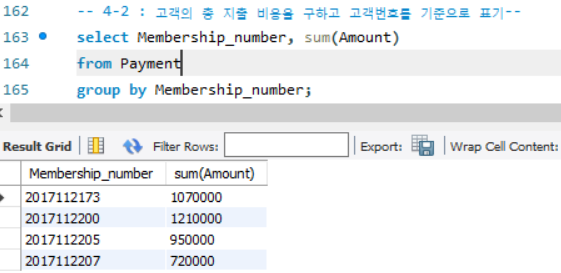
1)



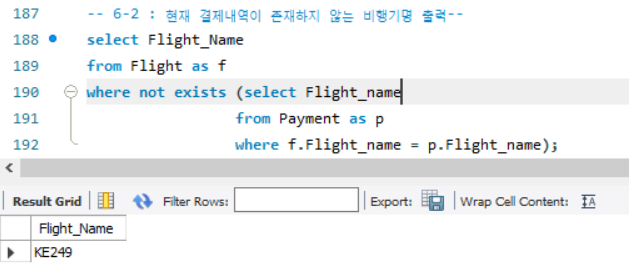
2)



3)



4)



**1-2.write a C (or Python etc) program which can process the four SQL queries.**

**You should include as many comments as possible to make it easier to be read**

(1) we call Membership\_number of Jun in database and save in variable select and print it.

sql = """select C.Membership\_number

from Customer as C

where C.Name\_ = 'Jun';"""

cursors.execute(sql)

select = list(cursors.fetchall())

db.commit()

print(select)

(2) we call payment\_id which jun paid it and save it in select 2 list.

sql2 = """select P.Payment\_id

from Payment as P, Customer as C

where C.Membership\_number = P.Membership\_number and C.Name\_ = 'Jun';"""

cursors.execute(sql2)

select2 = list(cursors.fetchall())

db.commit()

print(select2)

(3) We show the total cost each customer spent and Membership number in select3 list.

sql3 = """select Membership\_number, sum(Amount)

from Payment

group by Membership\_number;"""

cursors.execute(sql3)

select3 = list(cursors.fetchall())

db.commit()

print(select3)

(4) We show the flight name without the customer who booked the ticket.

sql4 = """select Flight\_Name

from Flight as f

where not exists (select Flight\_name

                 from Payment as p

                 where f.Flight\_name = p.Flight\_name);"""

cursors.execute(sql4)

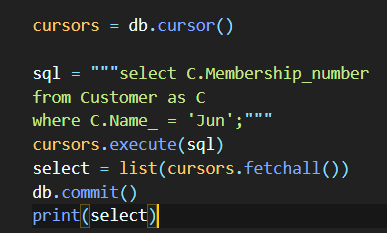
select4 = list(cursors.fetchall())

db.commit()

print(select4)

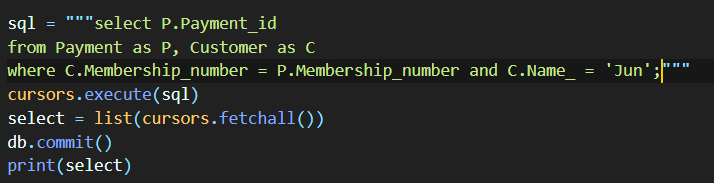
**1-3. show that your program work**

1)



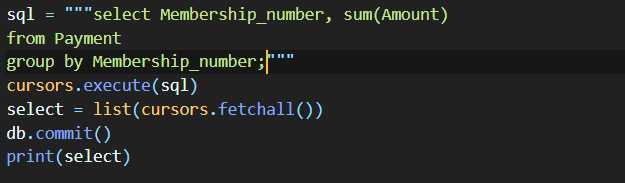


2)



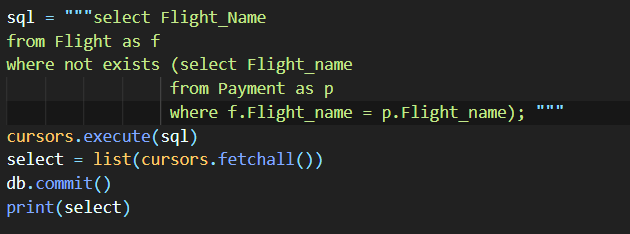


3)





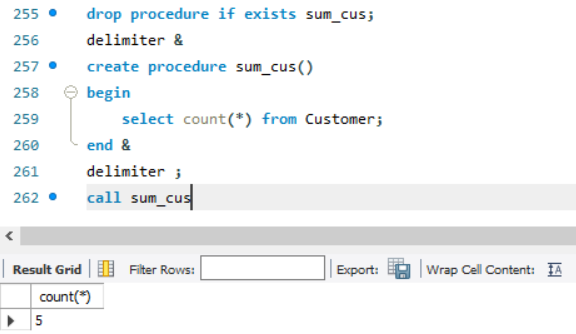
4)





**2. Implement the following**.

**2-1.define a procedure which contains an aggregate function.**



**2-2.define a function which returns a value**.

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

테이블이(가) 표시된 사진

자동 생성된 설명

**2-3. write a C (or Python etc) program which calls/uses both of the procedures/functions defined in 1)-2).**

텍스트, 화면, 스크린샷이(가) 표시된 사진

자동 생성된 설명

**2-4. show that your program works.**

1)

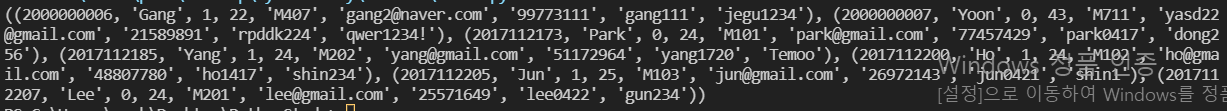
텍스트이(가) 표시된 사진

자동 생성된 설명 

2)

텍스트이(가) 표시된 사진

자동 생성된 설명



**3-1.describe what you are going to do with the use of web pages. Show the current web pages or you can draw ones yourself.**

텍스트이(가) 표시된 사진

자동 생성된 설명

Among the main functions of the page, we implemented pages with login and out, purchasing ticket, show reservation list and showing member information through Python. Customers can select what they want from the menu and use the necessary functions in the page.

2) implement a program using host language plus Mysql

import pymysql  
  
db = pymysql.connect(  
 host='localhost',  
 port=3306,  
 user='root',  
 password='dgu1234!',  
 db='jejuair',  
 charset='utf8'  
)  
cursors = db.cursor()  
sql = """select ID, Password\_   
from Customer"""  
cursors.execute(sql)  
select = list(cursors.fetchall())  
db.commit()  
  
flag = 1 # 로그인 여부 확인 플래그  
# id == park0417 pw == dong256  
while (1):  
 print("menu :(0) login (1) show my membership (2) book new flight (3)show my reservation (4) Logout")  
 i = int(input())  
 # 로그인 가능 여부 확인 후 로그인 기능 수행  
 if i == 0 and flag ==1:  
 id = input("ID : ")  
 pw = input("PW : ")  
 for i in range(len(select)):  
 if id == select[i][0] and pw == select[i][1]:  
 print("--LOG IN--")  
 flag = 0  
 if flag == 1:  
 print("--LOG IN FAIL--")  
 elif i==0 and flag ==0:  
 print("you already Login")  
 #1번 기능 수행  
 elif i == 1 and flag == 0:  
 print("--Show your Membership information--")  
 sql = """select \* from Customer"""  
 cursors.execute(sql)  
 select = list(cursors.fetchall())  
 db.commit()  
 #고객정보 print  
 for i in range(len(select)):  
 if id == select[i][7] and pw == select[i][8]:  
 print(select[i])  
 AM\_num = select[i][0]  
 #mileage 정보 print  
 sql = """select Available\_mileage from Mileage where Membership\_number = %s"""  
 cursors.execute(sql, AM\_num)  
 mileage = list(cursors.fetchall())  
 print("Available\_mileage : ", end="")  
 print(mileage[0][0])  
 #2번 기능 구현  
 elif i == 2 and flag == 0:  
 print("--Flight information--")  
 sql = """select \* from Flight"""  
 cursors.execute(sql)  
 select = list(cursors.fetchall())  
 db.commit()  
 for j in range(len(select)):  
 print(select[j])  
 print("--select Flight Name--")  
 f\_name = input()  
 for i in range(len(select)):  
 if select[i][0] == f\_name and select[i][6] >= 1:  
 print("--select 2 Additional Service--")#additional service 선택  
 add = []  
 add1 = int(input("More Baggage? : (1)Yes (0) No"))  
 add.append(add1)  
 add2 = int(input("Want meals? : (1)Yes (0) No"))  
 add.append(add2)  
 print("Additional Service(baggage, meal) : ", end="")  
 print(add)  
 print("--select 4 Pakage--")#Package 선택  
 pak = []  
 pak1 = int(input("Use Hotel? : (1)Yes (0) No"))  
 pak.append(pak1)  
 pak2 = int(input("Use Car\_rental service? : (1)Yes (0) No"))  
 pak.append(pak2)  
 pak3 = int(input("Use Pocket-Wifi? : (1)Yes (0) No"))  
 pak.append(pak3)  
 pak4 = int(input("Use Insurance? : (1)Yes (0) No"))  
 pak.append(pak4)  
 print("Package : ", end="")  
 print(pak)  
 total = (sum(add) \* 10000) + (sum(pak) \* 100000) + select[i][5] # 총 가격 도출  
 print("Total Cost : ", end="")  
 print(total)  
  
 print("--select Payment Type--") # 결제수단 선택 0 or 1  
 p\_type = int(input("(0) : Use only Credit (1): Use Mileage and Credit"))  
 if p\_type == 0: # 신용카드로만 결제  
 sql = """select Membership\_number from Customer where ID = %s and Password\_ = %s"""  
 cursors.execute(sql, (id, pw))  
 mem\_num = cursors.fetchall()  
 mem\_num = mem\_num[0][0] # 멤버쉽 번호 정보 받기  
  
 sql = """select Available\_mileage from Mileage where Membership\_number = %s"""  
 cursors.execute(sql, mem\_num)  
 AM = cursors.fetchall() # 받은 멤버쉽 넘버와 비교해서 보유 마일리지 값 변수에 저장  
 AM = AM[0][0]  
  
 sql = """UPDATE Mileage SET Available\_mileage = %s where Membership\_number = %s"""  
 AM = AM + total \* 0.01  
 cursors.execute(sql, (AM, mem\_num)) # 마일리지 정보 업데이트  
 db.commit()  
  
 # 새로운 결제정보 생성  
 cursors.execute("select count(\*) from payment;")  
 data = list(cursors.fetchall())  
 cnt = data[0][0]  
 new\_id = 101 + cnt  
 sql = """select Membership\_number from Customer where ID = %s and Password\_ = %s;"""  
 cursors.execute(sql, (id, pw))  
 mem\_num = cursors.fetchall()  
 mem\_num = mem\_num[0][0]  
 new\_data = (new\_id, mem\_num, f\_name, p\_type, total, AM)  
 sql = "INSERT INTO Payment values(%s,%s,%s,%s,%s,%s);"  
 cursors.execute(sql, new\_data)  
 db.commit()  
 print("--Payment is Completed--")  
  
 # 새로운 예약정보 생성  
 cursors.execute("select count(\*) from reservation;")  
 data = list(cursors.fetchall())  
 cnt = data[0][0]  
 new\_rid = 201 + cnt  
 new\_data = (new\_rid, mem\_num, new\_id, f\_name)  
 sql = "INSERT INTO reservation values(%s,%s,%s,%s)"  
 cursors.execute(sql, new\_data)  
 db.commit()  
  
 # 좌석 수 업데이트  
 sql = "update Flight set Remaining\_seat = Remaining\_seat -1 where Flight\_name = %s;"  
 cursors.execute(sql, f\_name)  
 db.commit()  
 break  
 elif p\_type == 1: # 마일리지도 이용한 결제  
 sql = """select Membership\_number from Customer where ID = %s and Password\_ = %s"""  
 cursors.execute(sql, (id, pw))  
 mem\_num = cursors.fetchall()  
 mem\_num = mem\_num[0][0] # 멤버쉽 번호 정보 받기  
  
 sql = """select Available\_mileage from Mileage where Membership\_number = %s"""  
 cursors.execute(sql, mem\_num)  
 AM = cursors.fetchall() # 받은 멤버쉽 넘버와 비교해서 보유 마일리지 값 변수에 저장  
 AM = AM[0][0]  
  
 total = total - AM # 총 가격에서 사용 가능 마일리지 차감  
 sql = """UPDATE Mileage SET Available\_mileage = %s where Membership\_number = %s"""  
 AM = total \* 0.01 # 마일리지 사용 이외 결제금액의 1%만큼 마일리지 적립  
 cursors.execute(sql, (AM, mem\_num))  
 db.commit()  
  
  
 print("--Payment is Completed--")  
 #새로운 결제정보 생성  
 cursors.execute("select count(\*) from payment;")  
 data = list(cursors.fetchall())  
 cnt = data[0][0]  
 new\_id = 101 + cnt  
 sql = """select Membership\_number from Customer where ID = %s and Password\_ = %s;"""  
 cursors.execute(sql, (id, pw))  
 mem\_num = cursors.fetchall()  
 mem\_num = mem\_num[0][0]  
 new\_data = (new\_id, mem\_num, f\_name, p\_type, total, AM)  
 sql = "INSERT INTO Payment values(%s,%s,%s,%s,%s,%s);"  
 cursors.execute(sql, new\_data)  
 db.commit()  
  
 #새로운 예약정보 생성  
 cursors.execute("select count(\*) from reservation;")  
 data = list(cursors.fetchall())  
 cnt = data[0][0]  
 new\_rid = 201 + cnt  
 new\_data = (new\_rid, mem\_num, new\_id, f\_name)  
 sql = "INSERT INTO reservation values(%s,%s,%s,%s)"  
 cursors.execute(sql, new\_data)  
 db.commit()  
  
 #좌석 수 업데이트  
 sql = "update Flight set Remaining\_seat = Remaining\_seat -1 where Flight\_name = %s;"  
 cursors.execute(sql, f\_name)  
 db.commit()  
 break  
 elif select[i][6] == 0:  
 print("No Remaining Seat") # 비행기의 잔여좍석이 없을 때  
 break  
 #Reservation 정보 출력  
 elif i == 3 and flag == 0:  
 sql = """select Membership\_number from Customer where ID = %s and Password\_ = %s"""  
 cursors.execute(sql, (id, pw))  
 mem\_num = cursors.fetchall()  
 mem\_num = mem\_num[0][0]  
  
 cursors.execute('select Membership\_number from Reservation;')  
 data = cursors.fetchall()  
 data\_list = list(data)  
 cond = 0  
 for i in range(0, len(data\_list)):  
 k = data\_list[i]  
 num = k[0]  
 if (mem\_num == num):  
 sql = """select \* from reservation where Membership\_number = %s"""  
 cursors.execute(sql, mem\_num)  
 data = cursors.fetchall()  
 cond = 1  
 print(data)  
 break  
 if cond == 0:  
 print("No reservation for you")  
 #로그아웃 후 페이지 종료  
 elif i == 4 and flag == 0:  
 print("--LOG OUT--")  
 break  
 #로그인이 안 되어있으면 요구  
 else:  
 print("--Please Log in--")

3) show that your program is working correctly.

1.show log\_in function first. If you choose any menu without login, program require login first. If you input wrong ID or PassWord, program show fail message. If all condition is clear, you can log in program.

텍스트이(가) 표시된 사진

자동 생성된 설명

2.Second function is showing customer’s information. You can see your information and available mileage in this function.

텍스트이(가) 표시된 사진

자동 생성된 설명

3.Third function is booking flight. If you choose this menu program show Flight information which you can choose. Select Flight by using Flight’s name, program show you additional service list and Pacakage list. After choose all of option program show you total cost, then you choose payment method(only credit or using mileage). After the payment process, update the changed DB information and return to the menu.

->Method 0(Use only Credit), you can see Reservation have been added

텍스트이(가) 표시된 사진

자동 생성된 설명



->Method 1(Use mileage and Credit), you can see on more Reservation have been added

텍스트이(가) 표시된 사진

자동 생성된 설명



4.Forth function is showing Reservation list of Customer. You can see Reservation list and Flight’s Remaining seat information is changed in pictures above.

5.Last function is Logout. If you finish your job, you choose this menu and program is finished.

텍스트이(가) 표시된 사진

자동 생성된 설명