유저 정보입력 페이지

1. 입력받는 정보

- usr_name = "Ausk"

- usr_age = "26"

- usr_region_id = "15"

- usr_location = "서울시 마포구"

- usr_holiday_tp_nm = "주6일근무"

- usr_min_sal = "25000000"

- usr_edu_level = "석사"

- usr_career = "1"

- usr_license = "컴퓨터활용능력1급"

- ▼ 오른쪽은 예시로 적어놓은 거고 이정도 정보를 입력받을 수 있으면 될 것 같다.

- ▼ 입력받을 때 career 는 1:경력, 2:관계없음, 3:신입 세 개 중에 고를 수 있게 하면 될 것 같다.

2. 각각의 정보가 들어가는 위치

```
INSERT INTO user (user_id, name, age, region_id, location, holiday_tp_nm, min_sal)
VALUES ('2', 'Nasty', '28', '237', '경기도 구리시', '주5일근무', '25000000');

INSERT INTO user_spec (user_id, user_spec_id, edu_level)
VALUES ('2', '2', '박사')

INSERT INTO user_career (user_spec_id, career_id)
VALUES ('2', '1');

INSERT INTO user_license(user_spec_id, license_id)
VALUES ('2', '컴퓨터활용능력1급')
```

3. 테스트 파이썬 코드

```
from __future__ import print_function
from bs4 import BeautifulSoup
```

```
import urllib.request
import mysql.connector

cnx = mysql.connector.connect(host='project.catth3zniejo.ap-northeast-2.rds.amazonaws.com',
port='3306', database='JobRecommendSystem', user='admin', password='tkdghkd1!')
cursor = cnx.cursor()

# user
q_find_usr_id = "SELECT user_id FROM user"
cursor.execute(q_find_usr_id)
usr_id_t= cursor.fetchall()
num_usr_id = len(usr_id_t)
usr_cnt = num_usr_id + 1

# input information
usr_name = "Ausk"
usr_age = "26"
usr_region_id = "15"
usr_location = "서울시 마포구"
usr_holiday_tp_nm = "주6일근무"
usr_min_sal = "25000000"
usr_edu_level = "석사"
usr_career = "1"
usr_license = "컴퓨터활용능력1급"

# input user, user_spec, usr_career
q_find_usr_id = '''SELECT user_id FROM user WHERE name = %s and age = %s and region_id = %s
and location = %s and holiday_tp_nm = %s and min_sal = %s'''
cursor.execute(q_find_usr_id, (usr_name, usr_age, usr_region_id, usr_location, usr_holiday_tp_nm, usr_min_sal))
usr_id_t= cursor.fetchall()
if len(usr_id_t) <= 0 :
    q_add_usrTable = ('''INSERT INTO user (user_id, name, age, region_id, location,
    holiday_tp_nm, min_sal) VALUES (%s, %s, %s, %s, %s, %s, %s)''')
    data_add_usrTable = (usr_cnt, usr_name, usr_age, usr_region_id, usr_location, usr_holiday_tp_nm, usr_min_sal)
    cursor.execute(q_add_usrTable, data_add_usrTable)
    print("new user inserted : ", usr_cnt, usr_name)
    usr_id = usr_cnt
    usr_cnt += 1

    q_add_usr_specTable = ("INSERT INTO user_spec (user_id, user_spec_id, edu_level) VALUES (%s, %s, %s)")
    data_add_usr_specTable = (usr_id, usr_id, usr_edu_level)
    cursor.execute(q_add_usr_specTable, data_add_usr_specTable)
    print("new user_spec inserted : ", usr_id)
    usr_spec_id = usr_id

    q_add_usr_careerTable = ("INSERT INTO user_career (user_spec_id, career_id) VALUES (%s, %s)")
    data_add_usr_careerTable = (usr_spec_id, usr_career)
    cursor.execute(q_add_usr_careerTable, data_add_usr_careerTable)
    print("new user_career inserted : ", usr_spec_id, usr_career)
else :
    usr_id = usr_id_t[0][0]
    usr_spec_id = usr_id


# usr_license
q_find_license_id = "SELECT license_id FROM license"
cursor.execute(q_find_license_id)
license_id_t= cursor.fetchall()
num_license_id = len(license_id_t)
license_cnt = num_license_id + 1

q_find_license_id = "SELECT license_id FROM license WHERE license = %s"
cursor.execute(q_find_license_id, (usr_license,))2
license_id_t= cursor.fetchall()
if len(license_id_t) <= 0 :
    q_add_licenseTable = ("INSERT INTO license (license_id, license) VALUES (%s, %s)")
    data_add_licenseTable = (license_cnt, usr_license)
    cursor.execute(q_add_licenseTable, data_add_licenseTable)
    print("new license inserted : ", license_cnt, usr_license)
    usr_license_id = license_cnt
    license_cnt += 1
else :
    usr_license_id = license_id_t[0][0]

q_add_usr_licenseTable = ("INSERT INTO user_license (user_spec_id, license_id) VALUES (%s, %s)")
data_add_usr_licenseTable = (usr_spec_id, usr_license_id)
cursor.execute(q_add_usr_licenseTable, data_add_usr_licenseTable)
```

```
print("new user_license inserted : ", usr_spec_id, usr_license)


cnx.commit()
cursor.close()
cnx.close()
```