AI- Assignment 5

Elective advisory/prediction system in Prolog for BTech or MTech students of IIITD with a small NLP interface in Python

This is an advisory system designed to help students with electives based on career choice and the prerequisite they have done. The input file for the prolog code is generated using a python interface.

- The code for the NLP interface is implemented in the file named NLP_interface.py.
- And the prolog code is implemented in 2020086.pl file.

NLP_interface.py:

1. To generate facts from User Input, I used two NLP algorithms,
    a. WordNetLemmatizer for career options and courses available and for the user Input of same.
    b. and PorterStemmer for the ssh/Interest option and user Input for the same.

    The reason behind the choice is that PorterStemmer stemming algorithm reduces the words to the root word. It converted acting to act, reading to read etc., but it can give Incorrect root values known as over-stemming and under-stemming. And we cannot risk incorrect classification in the case of career option and prerequisite done.
2. Taking user Input and matching them:
    a. **Career Choice:** If the user input matches with any of the updated values of the career options, a fact corresponding to the selected option is saved in the Input file.
    b. **Interest choice:** for the Interest choice, users' Input is passed through PorterStemmer steeming algorithm. If user input matches with one or more available ssh options, they are added to the list and then converted into facts and stored in the same input text file.
    c. **Prerequisite done:** For the Prerequisite, user Input is passed through WordNetLemmatizer. For matching the prerequisite, two methods are used:
        i. If the user entered the course code then directly add it to prev_courses_input_list.
        ii. Second is a score-based matching system between user input and available courses. If the user input and available courses have a score

greater than 0.5, then the user is asked to confirm the option, and the selected option is saved in the Input file.

2020086.pl

1. First, Read all the user Input from the input.txt file and convert them to facts.
2. Then, It prints the user option and the output generated on the console.

Screenshots:

1)

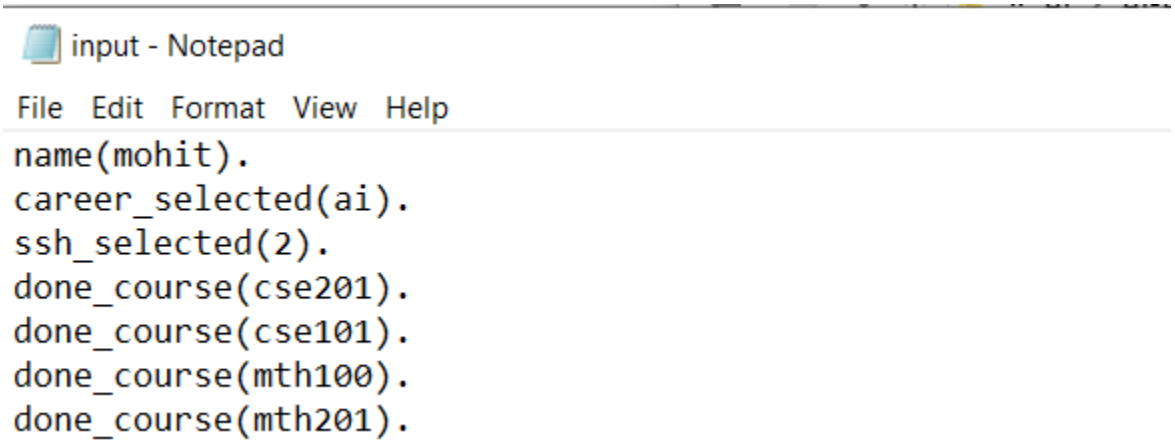
NLP_interface.py file

```
PS C:\Users\Sharm\Desktop\Courses\AI\AI-A5-Mohit-2020086> python -u "c:\Users\Sharm\Desktop\Cour
[nltk_data] Downloading package omw-1.4 to
[nltk_data]     C:\Users\Sharm\AppData\Roaming\nltk_data...
[nltk_data]   Package omw-1.4 is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\Sharm\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\Sharm\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
------------------------------------------------------------
Please enter your name: mohit
------------------------------------------------------------
Which career you want to pursue?
I want to pursue a career in Artificial Inteligence.
```

```
------------------------------------------------------------
What Are Your Interest Area?
I love to read books and I want to learn Acting.
------------------------------------------------------------
What are the courses that you have completed ?
In my first two years I completed basic cse course - cse101, cse201 and introductory maths courses
 on linear alebgra and probability.

------------------------------------------------------------
Please Verify the courses :
Completed: Linear Algebra(y or n) ? y
Completed: Probability and Statistics(y or n) ? y
------------------------------------------------------
Text file of facts named 'input.txt' generated
------------------------------------------------------
PS C:\Users\Sharm\Desktop\Courses\AI\AI-A5-Mohit-2020086>
```

input file generated



2020086.pl files output:

```
?- consult('2020086.pl').
true.

?- start.

_________Electives advisory / prediction system for BTech or MTech student of IIITD_________

Hi mohit!, Welcome, to Electives advisory / prediction system.
-----------------------------------------------------------
Carrer Option You Selected :  Artificial Intelligence

-----------------------------------------------------------
Area of Interest Choosen :
reading, acting, .
-----------------------------------------------------------
Pre-requisites Done:
mth201, Probability and Statistics
mth100, Linear Algebra
cse101, Introduction to Programming
cse201, Advanced Programming

-----------------------------------------------------------
Here are the recomended list of Electives according to your answers
Course Level  Course Code  Course Name
2             ssh214       Introduction To The Study Of Literature
2             ssh211       Theatre Appreciation
1             cse140       Introduction to Intelligent Systems
5             cse506       Data Mining
3             cse342       Statistical Machine Learning

-----------------------------------------------------------
Thanks for using our course prediction system
-----------------------------------------------------------
true
```

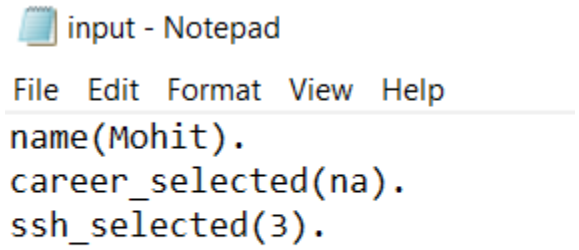
2)

NLP_interface.py file

```
-----------------------------------------------------------
Please enter your name: Mohit
-----------------------------------------------------------
Which career you want to pursue?
Haven't decided yet.
-----------------------------------------------------------
What Are Your Interest Area?
I love poetry.
-----------------------------------------------------------
What are the courses that you have completed ?
I haven't completed any course yet.


-----------------------------------------------------------
Please Verify the courses :
-----------------------------------------------------------
Text file of facts named 'input.txt' generated
-----------------------------------------------------------
```

Input file generated



2020086.pl files output:

```
?- consult('2020086.pl').
true.

?- start.

_________Electives advisory / prediction system for BTech or MTech student of IIITD_________

Hi _10152!, Welcome, to Electives advisory / prediction system.
------------------------------------------------------------
Carrer Option You Selected :  Not Decided

------------------------------------------------------------
Area of Interest Choosen :
poerty, .
------------------------------------------------------------
Looks like you haven't decided yet
No worries! Here are some Introductory courses to Help you.
1: Introduction to Intelligent Systems - cse140
2: Human Computer Interaction - des204
3: Fundamentals of Database Management System - cse202
4: Fundamentals of Database Management System - cse202
5: Econometrics I -eco221

------------------------------------------------------------
Thanks for using our course prediction system
------------------------------------------------------------
true
```

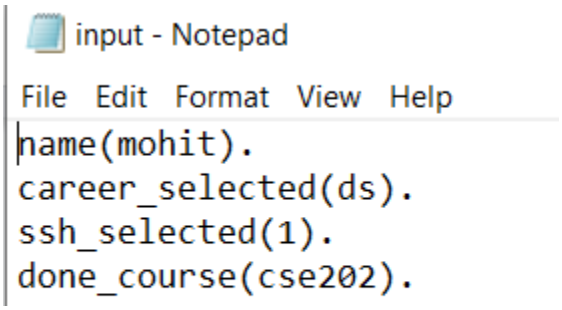
3)

NLP_interface.py file

```
------------------------------------------------------
Please enter your name: mohit
------------------------------------------------------
Which career you want to pursue?
I want to pursue Data analyst as a career.
------------------------------------------------------
What Are Your Interest Area?
Playing and acting.
------------------------------------------------------
What are the courses that you have completed ?
I have done one course on Database management.

------------------------------------------------------
Please Verify the courses :
Completed: Fundamentals of Database Management System(y or n) ? y
------------------------------------------------------
Text file of facts named 'input.txt' generated
------------------------------------------------------
PS C:\Users\Sharm\Desktop\Courses\AI\AI-A5-Mohit-2020086>
```

input file generated



2020086.pl files output:

```
[1]  ?- consult('2020086.pl').
true.

[1]  ?- start.

_________Electives advisory / prediction system for BTech or MTech student of IIITD________
__
Hi mohit!, Welcome, to Electives advisory / prediction system.
------------------------------------------------------------
Carrer Option You Selected :  Data Scientist

------------------------------------------------------------
Area of Interest Choosen :
acting, .
------------------------------------------------------------
Pre-requisites Done:
cse202, Fundamentals of Database Management System

------------------------------------------------------------
Here are the recomended list of Electives according to your answers
Course Level  Course Code  Course Name
2             ssh211       Theatre Appreciation
5             cse557       Big Data Analytics
6             cse606       Data Warehouse

------------------------------------------------------------
Thanks for using our course prediction system
------------------------------------------------------------
true
```

Source Code:

Python 2020086.pl:

```
import nltk
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize
from nltk.stem import PorterStemmer
from nltk.corpus import stopwords
from pyswip import Prolog
import string


option = {'ai':'Artificial Intelligence', 'qrs':'Quantum Researcher', 'ds':'Data Scientist'
        ,'uiux':'UI/UX designer', 'econ':'Economists', 'na':'Not Decided'}

option_ssh = {'1':'acting','2':'reading','3':'poetry','4':'history'}

prev_courses_available = {'ssh214':'Introduction To The Study Of Literature',
            'ssh215':'Indian Poetry Through the Ages',
            'ssh211':'Theatre Appreciation',
            'ssh212':'Nation and her Narratives',
            'cse643':'Aritificial Intelligence',
            'cse343':'Machine Learning',
            'cse140':'Introduction to Intelligent Systems',
            'cse556':'Natural Language Processing',
            'cse506':'Data Mining',
            'cse342':'Statistical Machine Learning',
            'ece525':'Quantum Mechanics',
            'ece545':'Photonics: Fundamentals & Applications',
            'ece524':'Quantum Materials and Devices',
            'cse622':'Introduction to Quantum Computing',
            'cse202':'Fundamentals of Database Management System',
            'cse557':'Big Data Analytics',
            'cse606':'Data Warehouse',
            'cse507':'Database System Implementation',
            'des130':'Prototyping Interactive Systems',
            'des101':'Design Drawing & Visualization',
            'des205':'Design of Interactive Systems',
            'des202':'Visual Design & Communication',
            'des201':'Design Processes and Perspectives',
            'des204':'Human Computer Interaction',
            'des524':'Ergonomics/Human factors for Design',
            'eco221':'Econometrics I',
            'eco201':'Macroeconomics',
            'eco301':'Microeconomics',
            'eco311':'Game Theory',
            'eco331':'Foundations of Finance',
            'cse102':'Data Structures & Algorithms',
            'cse201':'Advanced Programming',
            'cse101':'Introduction to Programming',
            'cse222':'Algorithm Design and Analysis',
```

```
        'mth100':'Linear Algebra',
        'mth201':'Probability and Statistics',
        'mth203':'Multivariate Calculus',
        'ece230':'Fields and Waves'}

carrer_option = {}
ssh_option = {}
prev_courses_option = {}

def setting_env():
    non_useful_words = set(stopwords.words('english'))
    nltk.download('omw-1.4')
    nltk.download('stopwords')
    nltk.download('wordnet')
    wl = WordNetLemmatizer()
    ps = PorterStemmer()

    for carrer in option.keys():
        carrer_up = option[carrer].lower()    #helps while compairing values
        carrer_up1 = ""
        for ch in carrer_up:
            if ch not in string.punctuation:
                carrer_up1 += ch
            else:
                carrer_up1 += " "
        carrer_up1 = wl.lemmatize(carrer_up1)
        token = word_tokenize(carrer_up1)
        carrer_option[carrer] = token

    for ssh in option_ssh.keys():
        ssh_up = option_ssh[ssh].lower()    #helps while compairing values
        ssh_up1 = ""
        for ch in ssh_up:
            if ch not in string.punctuation:
                ssh_up1 += ch
            else:
                ssh_up1 += " "
        ssh_up1 = ps.stem(ssh_up1)
        token = word_tokenize(ssh_up1)
        ssh_option[ssh] = token

    for course in prev_courses_available.keys():
        course_up = prev_courses_available[course].lower()    #helps while compairing values
        course_up1 = ""
        for ch in course_up:
            if ch not in string.punctuation:
                course_up1 += ch
            else:
                course_up1 += " "
        course_up1 = wl.lemmatize(course_up1)
        token = word_tokenize(course_up1)
        token_1 = []
        for i in token:
```

```
            if i not in non_useful_words:
                token_1.append(i)
        prev_courses_option[course] = token_1


def student_info():
    print('------------------------------------------------------------')
    name= str(input('Please enter your name: '))
    file_ptr = open("input.txt", 'w')
    file_ptr.write('name('+name+').\n')
    file_ptr.close()


def career_choice():
    non_useful_words = set(stopwords.words('english'))
    wl = WordNetLemmatizer()
    #taking a complete sentance as an input
    print('------------------------------------------------------------')
    print("Which career you want to pursue?")
    carrer_input = input()
    carrer_input_list = []
    wl = WordNetLemmatizer()

    carrer_input = carrer_input.lower()    #helps while compairing values
    carrer_input1 = ""
    for ch in carrer_input:
        if ch not in string.punctuation:
            carrer_input1 += ch
        else:
            carrer_input1 += " "
    carrer_input1 = wl.lemmatize(carrer_input1)
    carrer_input_list = word_tokenize(carrer_input1)

    count = 0
    carrer_choosen = ''
    for i in carrer_option:
        for j in carrer_input_list:
            if j not in non_useful_words and carrer_option[i].count(j):
                carrer_choosen = i
                count = count + 1
                break

    if count > 1:
        print('Only one carrer choice is allowed, Procceding with one of the input choices')
    # If it didn't match with any
    elif count == 0:
        carrer_choosen = 'na'
    file_ptr = open('input.txt', 'a')
    file_ptr.write('career_selected('+carrer_choosen+').\n')
    file_ptr.close()


def ssh_choice():
```

```
    non_useful_words = set(stopwords.words('english'))
    ps = PorterStemmer()
    print('------------------------------------------------------------')
    print("What Are Your Interest Area?")
    ssh_input = input()
    ssh_input_list = []

    ssh_input = ssh_input.lower()    #helps while compairing values
    ssh_input1 = ""
    for ch in ssh_input:
        if ch not in string.punctuation:
            ssh_input1 += ch
        else:
            ssh_input1 += " "
    ssh_input_list = word_tokenize(ssh_input1)
    for i in range(0, len(ssh_input_list)):
        ssh_input_list[i] = ps.stem(ssh_input_list[i])

    ssh_choosed = []
    for i in ssh_option:
        for j in ssh_input_list:
            if j not in non_useful_words and ssh_option[i].count(j):
                ssh_choosed.append(i)

    if len(ssh_choosed) == 0:
        ssh_choosed.append('5')

    file_ptr = open('input.txt', 'a')
    for i in ssh_choosed:
        file_ptr.write('ssh_selected('+i+').\n')
    file_ptr.close()


def prev_done_course():
    non_useful_words = set(stopwords.words('english'))
    wl = WordNetLemmatizer()
    print('------------------------------------------------------------')
    print("What are the courses that you have completed ? ")
    prev_courses_input = input()
    prev_courses_input_list = []

    prev_courses_input = prev_courses_input.lower()    #helps while compairing values
    prev_courses_input1 = ""
    for ch in prev_courses_input:
        if ch not in string.punctuation:
            prev_courses_input1 += ch
        else:
            prev_courses_input1 += " "
    prev_courses_input1 = wl.lemmatize(prev_courses_input1)
    prev_courses_input_list = word_tokenize(prev_courses_input1)

    course_done = []
```

```
    score = {}
    # if user added the course code
    for i in prev_courses_option:
        score[i] = 0
        if prev_courses_input_list.count(i):
            course_done.append(i)

    for i in prev_courses_option:
        for j in prev_courses_input_list:
            if j not in non_useful_words and prev_courses_option[i].count(j):
                score[i] += 1

    for i in score:
        score[i] = score[i]/len(prev_courses_option[i])

    print('\n-------------------------------------------------------------')
    print("Please Verify the courses : ")
    for i in score:
        if score[i] >= 0.5:
            option = str(input("Completed: "+prev_courses_available[i]+ "(y or n) ? "))
            if option == 'y':
                course_done.append(i)

    file_ptr = open('input.txt', 'a')
    for i in course_done:
        file_ptr.write('done_course('+i+').\n')
    file_ptr.close()

def main():
    setting_env()
    student_info()
    career_choice()
    ssh_choice()
    prev_done_course()

    # swipl = Prolog()
    # swipl.consult('2020086.pl')
    # list(swipl.query('start'))
    print('-------------------------------------------------------')
    print("Text file of facts named 'input.txt' generated")
    print('-------------------------------------------------------')

main()
```

Prolog 2020086.pl:

```
% Type start. to run the code

:-style_check(-singleton).
:-style_check(-discontiguous).
career(1,ai, 'Artificial Intelligence').
career(2,qrs, 'Quantum Researcher').
career(3,ds, 'Data Scientist').
career(4,uiux, 'UI/UX designer').
career(5,econ, 'Economist').
career(6,na, 'Not Decided').

ssh(1,acting).
ssh(2,reading).
ssh(3,poerty).
ssh(4,history).
ssh(5,non).

start:-
    nl,write('__________Electives advisory / prediction system for BTech or MTech student of IIITD__________'),nl,
    setting_env,
    read_input,
    student_info,
    career_choice,
    interest_course,
    nan,
    pre_req_done(List),
    print_choice_availble,
    exiting.


setting_env:-                       % helper funtion used to set all facts and rules
    retractall(name(_)),
    retractall(course_done_or_not()),
    retractall(course_choosen(_,_)),
    retractall(course(_,_,_,_)),
    retractall(course_to_print(_,_)),
    retractall(pre_req(_,_)),
    retractall(ssh_selected(_)),
    retractall(done_course(_)),
    retractall(career_selected(_)),
    define_course,
    define_prereq.

define_course:-
    retractall(course(_,_,_,_)),
    retractall(course_to_print(_,_)),
    assert(course(acting,'Theatre Appreciation',ssh211,2)),
    assert(course(reading,'Introduction To The Study Of Literature',ssh214,2)),
    assert(course(poerty,'Indian Poetry Through the Ages',ssh215,2)),
    assert(course(history,'Nation and her Narratives',ssh212,2)),
```

```
    assert(course(ai,'Introduction to Intelligent Systems',cse140,1)),
    assert(course(ai,'Aritificial Intelligence',cse643,6)),
    assert(course(ai,'Machine Learning',cse343,3)),
    assert(course(ai,'Natural Language Processing',cse556,5)),
    assert(course(ai,'Data Mining',cse506,5)),
    assert(course(ai,'Statistical Machine Learning',cse342,3)),

    assert(course(qrs,'Quantum Mechanics',ece525,5)),
    assert(course(qrs,'Photonics: Fundamentals & Applications',ece545,5)),
    assert(course(qrs,'Quantum Materials and Devices',ece524,5)),
    assert(course(qrs,'Introduction to Quantum Computing',cse622,6)),

    assert(course(ds,'Fundamentals of Database Management System',cse202,2)),
    assert(course(ds,'Big Data Analytics',cse557,5)),
    assert(course(ds,'Data Warehouse',cse606,6)),
    assert(course(ds,'Database System Implementation',cse507,5)),

    assert(course(uiux,'Prototyping Interactive Systems',des130,1)),
    assert(course(uiux,'Design Drawing & Visualization',des101,1)),
    assert(course(uiux,'Design of Interactive Systems',des205,2)),
    assert(course(uiux,'Visual Design & Communication',des202,2)),
    assert(course(uiux,'Design Processes and Perspectives',des201,2)),
    assert(course(uiux,'Human Computer Interaction',des204,2)),
    assert(course(uiux,'Ergonomics/Human factors for Design',des524,5)),

    assert(course(econ,'Econometrics I',eco221,2)),
    assert(course(econ,'Macroeconomics',eco201,2)),
    assert(course(econ,'Microeconomics',eco301,3)),
    assert(course(econ,'Game Theory',eco311,3)),
    assert(course(econ,'Foundations of Finance',eco331,3)),

    assert(course(cse,'Data Structures & Algorithms',cse102,1)),
    assert(course(cse,'Advanced Programming',cse201,2)),
    assert(course(cse,'Introduction to Programming',cse101,1)),
    assert(course(cse,'Algorithm Design and Analysis',cse222,2)),

    assert(course(mth,'Linear Algebra',mth100,1)),
    assert(course(mth,'Probability and Statistics',mth201,2)),
    assert(course(mth,'Multivariate Calculus',mth203,2)),
    assert(course(ece,'Fields and Waves',ece230,2)).

define_prereq:-
    retractall(pre_req(_,_)),
    assert(pre_req(cse643,cse102)),
    assert(pre_req(cse622,mth100)),
    assert(pre_req(cse202,cse102)),
    assert(pre_req(cse201,cse102)),
    assert(pre_req(cse201,cse101)),
    assert(pre_req(cse102,cse101)),
    assert(pre_req(cse222,cse102)),
    assert(pre_req(ece230,mth203)),
    assert(pre_req(des202,des101)),
```

```
    assert(pre_req(des201,des101)),
    assert(pre_req(cse343,mth100)),
    assert(pre_req(cse343,mth201)),
    assert(pre_req(cse343,cse101)),
    assert(pre_req(cse343,mth203)),
    assert(pre_req(cse556,cse101)),
    assert(pre_req(cse556,mth201)),
    assert(pre_req(cse556,cse222)),
    assert(pre_req(cse556,mth100)),
    assert(pre_req(cse506,cse101)),
    assert(pre_req(cse506,mth100)),
    assert(pre_req(cse506,mth201)),
    assert(pre_req(cse342,cse101)),
    assert(pre_req(cse342,mth201)),
    assert(pre_req(cse557,cse202)),
    assert(pre_req(cse606,cse202)),
    assert(pre_req(cse507,cse102)),
    assert(pre_req(cse507,cse202)),
    assert(pre_req(ece525,mth100)),
    assert(pre_req(cse622,mth100)),
    assert(pre_req(ece545,ece230)),
    assert(pre_req(eco221,mth201)).


read_file_l(InputStream,[]) :-
    at_end_of_stream(InputStream).

read_file_l(InputStream,[H|T]) :-
    \+ at_end_of_stream(InputStream),
    read(InputStream,H),
    read_file_l(InputStream,T).

define_fact([H|T]):-
    define_fact(T);assert(H).
define_fact([]).

read_input:-
    open('input.txt', read, InputStream),
    read_file_l(InputStream,List),
    close(InputStream),
    define_fact(List).

student_info:-
    name(Name),
    write('Hi '),write(Name),write('!, Welcome, to Electives advisory / prediction system.'),nl.

career_choice:-
    write('------------------------------------------------------------'),nl,
    write('Carrer Option You Selected :  '),
    career_selected(X),career(_,X,Y),write(Y),nl.

helper1([Ch|List]):-            % get choice for area of interest
    (ssh(Ch,Type),course(Type,X,Y,Z)->(assert(course_choosen(X,Y)),helper1(List));helper1(List)).
```

```
helper1([]).


interest_course:-
    findall(X, ssh_selected(X), List),
    nl,write('------------------------------------------------------------'),nl,
    write('Area of Interest Choosen :'),nl,
    forall(ssh_selected(X),(ssh(X,Y),write(Y),write(', '))),
    write('.'),nl,
    helper1(List).

helper2:-

career_selected(ai),retractall(course(qrs,_,_,_)),retractall(course(ds,_,_,_)),retractall(course(uiux,_,_,_)),retractall(course(econ,_,_,_))
;

career_selected(qrs),retractall(course(ai,_,_,_)),retractall(course(ds,_,_,_)),retractall(course(uiux,_,_,_)),retractall(course(econ,_,_,_))
;

career_selected(ds),retractall(course(qrs,_,_,_)),retractall(course(ai,_,_,_)),retractall(course(uiux,_,_,_)),retractall(course(econ,_,_,_))
;

career_selected(uiux),retractall(course(qrs,_,_,_)),retractall(course(ds,_,_,_)),retractall(course(ai,_,_,_)),retractall(course(econ,_,_,_))
;

career_selected(econ),retractall(course(qrs,_,_,_)),retractall(course(ds,_,_,_)),retractall(course(uiux,_,_,_)),retractall(course(ai,_,_,_))
,
    fail;true.


helper4([]).                % delete all the course and pre_req facts for the course that uses has completed
helper4([A|B]):-
    (retractall(pre_req(_,A));retractall(course(_,_,A,_))),helper4(B).


pre_req_done(List):-
    \+ career_selected(na),
    write('------------------------------------------------------------'),nl,
    write('Pre-requisites Done: '),nl,
    findall(X, done_course(X), List),
    forall(done_course(X),(course(A,B,X,D),write(X),write(', '),write(B),nl)),
    helper2,
    helper4(List),fail;true.

helper7:-
    pre_req(A,B),
    retractall(pre_req(A,B)),retractall(course(C,D,A,F)),helper7,fail;true.

helper8:-
    career_selected(Ch),course(Ch,A,B,C),retractall(course(Ch,A,B,C)),assert(course_choosen(A,B)).

helper9(X):-
    course_choosen(A,B),define_course,course(M,N,B,P),
```

```
    retract(course_choosen(A,B)),
    write(P),write("            "),write(B),write("      "),write(A),nl,
    Y is X + 1,
    helper9(Y),fail;true.

print_choice_availble:-
    \+ career_selected(na),
    nl,write('-------------------------------------------------------------'),nl,
    write('Here are the recomended list of Electives according to your answers'),nl,
    write('Course Level  Course Code  Course Name'),nl,
    helper7,
    helper8,
    helper9(1),
    fail;true.

nan:-
    career_selected(na),
    write('-------------------------------------------------------------'),nl,
    write('Looks like you haven\'t decided yet'),nl,
    write('No worries! Here are some Introductory courses to Help you.'),nl,
    write('1: Introduction to Intelligent Systems - cse140'),nl,
    write('2: Human Computer Interaction - des204'),nl,
    write('3: Fundamentals of Database Management System - cse202'),nl,
    write('4: Fundamentals of Database Management System - cse202'),nl,
    write('5: Econometrics I -eco221'),nl,
    fail;true.

exiting:-
    nl,write('-------------------------------------------------------------'),nl,
    write('Thanks for using our course prediction system'),nl,
    write('-------------------------------------------------------------'),nl.
```