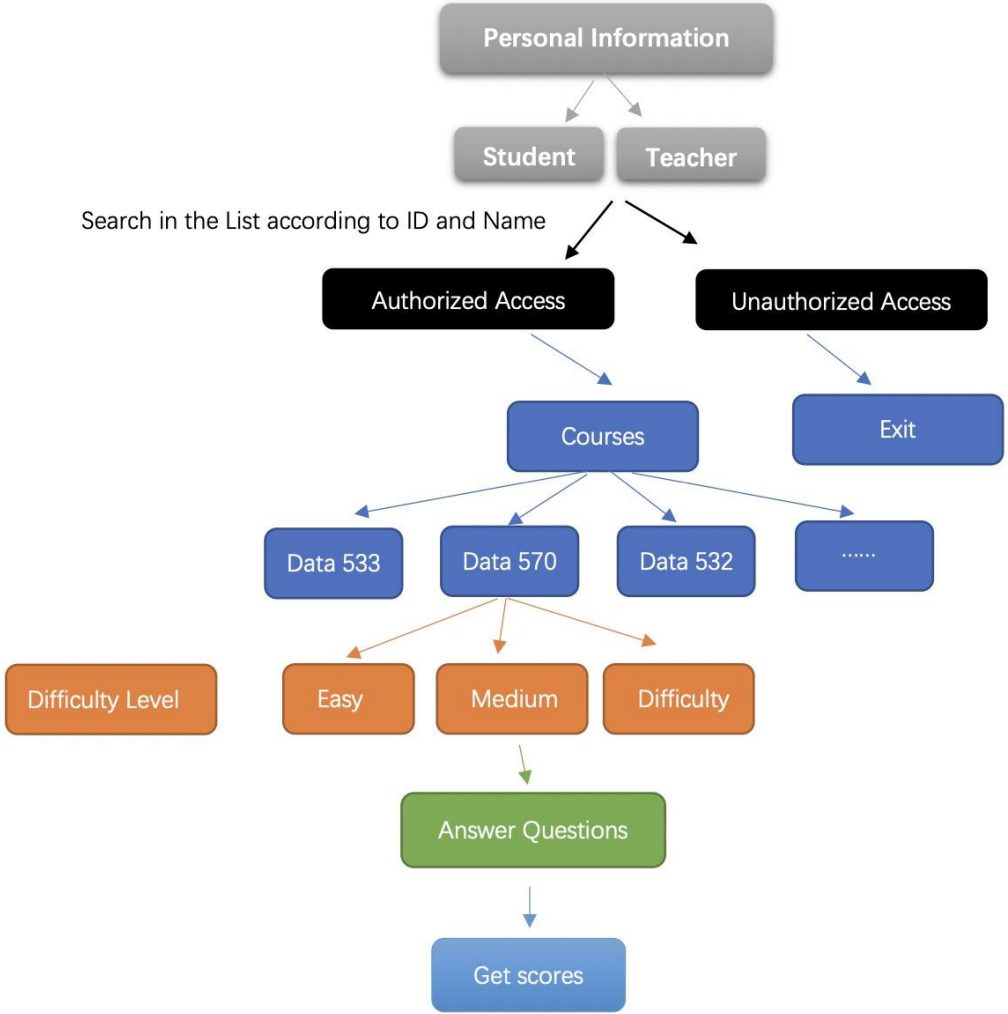
MDS Quiz Apps:

JUSTIN CHAN, LILI TANG

Motivation:


Personal Information
Student
Teacher
Search in the List according to ID and Name
Authorized Access
Unauthorized Access
Courses
Exit
Data 533
Data 570
Data 532
.....
Difficulty Level
Easy
Medium
Difficulty
Answer Questions
Get scores

Sub-Package 1

This package will include the personal information input and output.

Also, we will divide it into two categories, student and teacher.

Sub-Package 1 Module 1

Set functions about information-input, such as name, ID.

If a student, users need to input their major and degree;

If a teacher, users need to input their research direction and teaching length.

```
#information-input


def group():
        g=input("Are you a student or a teacher?")
        return g

def name():
        n=input("What's your name?")
        return n

def ID():
        I=input("What's your ID? ")
        return I

# a student
def major():
        n=input("What's your major? ")
        return n
def degree():
        n=input("Are you a undergraduate or postgraduate?")
        return n
# a teacher
def research():
        n=input("what is your research direction?")
        return n

def length():
        n=input("How many years since be a tearcher?")
        return n
```

Sub-Package 1 Module 2

Functions:information-output
Use the inheritance in this module.

Group common information: name and ID.

Additional information
Student: major and degree
Teacher: research and length)

Sub-Package 1
Module 2

```
class Group:
    def __init__(self, name, ID):
        self.name = name
        self.ID = ID

    def display(self):
        print('Name:{} ID:{} '.format(self.name, self.ID))


class Student(Group):
    def __init__(self, name, ID, major, degree):
        Group.__init__(self, name, ID)
        self.major=major
        self.degree=degree

    def display(self):
        Group.display(self)
        print('Major: {} Degree: {}'.format(self.major, self.degree))

class Teacher(Group):
    def __init__(self, name, ID, research,length):
        Group.__init__(self, name, ID)
        self.research=research
        self.length=length

    def display(self):
        Group.display(self)
        print('Research: {} Length: {}'.format(self.research, self.length))
```

Sub-Package 1 Module 2

Finally, we need to conclude whether it is a student or a teacher and input and output different type information

```
if  group()=="student":
    v1 = Student(name(), ID(), major(),degree());
    groups= [v1]
    for group in groups:
        group.display()

else:
    v1 = Teacher(name(), ID(), research(),length());
    groups= [v1]
    for group in groups:
        group.display()
```

Sub-Package 1 Module 2

Examples:

Input

```
Are you a student or a teacher? student
What's your name? Lily
What's your ID?  9130
What's your major?  Data Science
Are you a undergraduate or postgraduate? Postgraduate
```

Output

```
Name:Lily ID:9130
Major: Data Science Degree: Postgraduate
```

Input

```
Are you a student or a teacher? teacher
What's your name? Lily
What's your ID?  1111
what is your research direction? Machine Learning
How many years since be a tearcher? 10
```

Output

```
Name:Lily ID:1111
Research: Machine Learning Length: 10
```

Sub-Package 2 Module 1

This Module contain Class “Checking”, which contain all the necessary checking function such as:

1. **isEmpty(self) -** checking if there is more question to the question set. If there is, run function **next_question()** to prompt the user with new question and ask for their input.

2. **next_question(self) -** prompt the user with new question and ask for their input and validate user answer using **check_answer()**.

3. **check_answer(self, user_answer, correct_answer)** - checking if the user input matched the database answer and if not prompt the user with the correct answer.

4. **check_score(self)** - checking user score.

Sub-Package 2 Module 2

This Module contain ```class Question(Checking):``` which is inherited from Checking class, this class only have one function:

1. **__init__ (self, q_text, q_answer, q_diff)** - process database's question and allocate them into question, anser and difficulty.

Sub-Package 2 Module 3

This Module contain ```Course(Question,Checking):``` which is inherited from Checking class and Question class, this module are for appending database's questions, answer into separate array with different difficulty for the other class to access based on user inputs in the future. this class only have 4 function:

1. **append_text(self,question) -** for appending database question in to its own question array

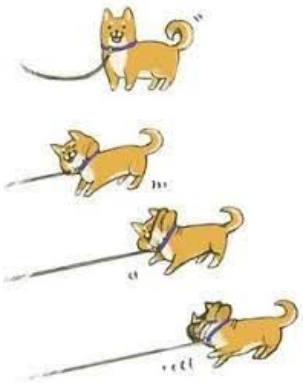
2. **append_answer(self,question) -** for appending database's answer in to its own answer array

3. **append_difficulty(self,question) -** for appending database's difficulty in to its own difficulty array

4. **append_courseq(self) -** for appending database question, answer according to the difficulty into 4 different question set array for user to access it after knowing the user difficulty that he/she want to be quiz on.

Demo:

```
What do you want to be Quiz on?  data541


What difficulty?  easy
*********************************************
*             Welcome to DATA541.           *
* There is 0 questions in total in DATA541.  *
*********************************************
Question 1. How many of the following statements are TRUE?(Answer in Number)
1) Git is a is distributed version control system
2) Git is designed to support parallel development
3) Git is a web-based repository Service
4) Git doesn't require a server
 3
Correct!!!


Question 2.  How many of the following statements are TRUE?(Answer in Number)
1) git branch NewBranch creates a new branch
2) git branch NewBranch does not check out to the NewBranch
3) git branch is tightly integrated with the git checkout and git merge commands
4) git branch -d NewBranch can be used to force delete the branch called NewBranch
 1
Wrong   :(


The correct answer is: 3

*************************************
*************************************

Final score: 1 out of 2.

*************************************
*************************************
```



Question?

Thanks!