Exercise 4

1. Constructor, Destructor & Vector

1-1. Description

- Create class 'Member' with the construction of

```
class Member{
 private:
  string name;
  int age;
  string department;
 public:
  Member(string name, int age, string department);
  //~Member();
  //Destructor is called implicitly when the object is out of the scope it is called, so when finishing one loop, C++ compiler implicitly calls Destructor.
  string getName();
  int getAge();
  string getDept();
  void print();
};
```

- Write your own code, that gets member information, can create or delete instance with message.
- names of Member are not like each other
- Contain Member instances inside vector

1-2. Input Description

- Enter name,age,department as many as you want
- Create instance immediately after you get a set of information for Member
- When QUIT is entered for name or department, or 0 is entered for age, then entering input is finished.
- After finishing input, the program prints out information of Members in turn, and deletes all Member Instances inside vector.
- Instance prints out messages when it is created and deleted

1-3. Input&Output Example

- Input part is marked by > in front.(Not in real implementation)

```
>Harry 37 Hogwarts
Harry is 37 years old.
>Viktor 41 Durmstrang
Viktor is 41 years old.
>Fleur 40 Beauxbatons
Fleur is 40 years old.
>QUIT 0 QUIT
Harry : Hogwarts, 37 years old.
Viktor : Durmstrang, 41 years old.
Fleur : Beauxbatons, 40 years old.
```

2. Inheritance

2-1. Description

- Create class Student that inherits Person class as below.

```
class Person{
 private://can use protected:
  string name;
 int age;
 string gender;//Male or Female
 //When inheritting class, if a member variable is private, you must access to variable with getter, setter in child class, otherwise, you can use protected scope.
 public:
 Person(...);
 ~Person();
 string getName();
 int getAge();
 string getGender();
 void introduce();
};
```

- Function void introduce() prints out the information of Person class

  ```
  NAME is AGE years old, and is GENDER
  ```

- class Student have its own member variable that saves school name.
- Function void introduce() of class Student prints out the information of Student class

  ```
  NAME is AGE years old, and is GENDER
  NAME is studying in SCHOOL
  ```

- names of Member are not like each other

2-2. Input Description

- At first line, input the number(N) of Person you would enter.
- From second line, for N lines, enter name, age, gender.
- Then, enter the number(M) of Students you would create from students.
- For M lines, enter the name, and school
- After all input is done, print out introduce() of all persons and students, following the order of person.

2-3. Input Example

```
3
HyunIl 24 Male
Byunghoon 24 Male
Jiwoo 24 Female
2
Jiwoo dcslab
HyunIl dcslab
```

2-4. Output Example

```
HyunIl is 24 years old, and is Male
Hyunil is studying in dcslab
Byunghoon is 24 years old, and is Male
Jiwoo is 24 years old, and is Female
Jiwoo is studying in dcslab
```