

Person類別範例4

類別繼承

簡介

Person類別是用以對「真實世界(real world)」中的「人」進行「抽象對映(mapping abstraction)」的定義。

Data Members

- [-] firstname : string
- [-] lastname : string

Member Functions

- [+] set_firstname(string fn) : void -> setter for firstname
- [+] get_firstname():string -> getter for firstname
- [+] set_lastname(string fn) : void -> setter for lastname
- [+] get_lastname():string -> getter for lastname
- [+] showInfo(): void -> print out its information

Constructor and Destructor

- Person() -> set default values of firstname and lastname as "unknown"
- Person(string fn, string ln) -> set firstname and lastname as fn and ln
- ~Person() -> print out that it has been deleted!

Student類別範例

簡介

Student是繼承自Person類別的類別，用以對「真實世界(real world)」中的「學生」進行「抽象對映(mapping abstraction)」的定義。但比起Person類別，Student類別又更為特別一些，它還多了一個名為ID的data member及相關的setter與getter:

Data Members

- [-] firstname : string
- [-] lastname : string
- [-] ID : string

Member Functions

- [+] set_firstname(string fn) : void -> setter for firstname
- [+] get_firstname():string -> getter for firstname
- [+] set_lastname(string fn) : void -> setter for lastname
- [+] get_lastname():string -> getter for lastname
- [+] showInfo(): void -> print out its information
- [+] set_ID(string id): void -> setter for ID
- [+] get_ID(): string -> getter for Id

Constructor and Destructor

None

Source Code

```
#include <iostream>
using namespace std;

#ifndef _PERSON_
#define _PERSON_

class Person
{
private:
    string firstname;
    string lastname;

public:
    Person();
    Person(string, string);
    ~Person();
    void showInfo();
    void set_firstname(string fn);
    string get_firstname();
    void set_lastname(string ln);
    string get_lastname();
};
#endif
```

```
#include "person.h"

Person::~Person()
{
    cout << "A Person is removed." << endl;
}

Person::Person()
{
    cout << "A Person is created." << endl;
}

Person::Person(string fn, string ln)
{
    firstname=fn;
    lastname=ln;
}

void Person::showInfo()
{
    cout << "Name: " << firstname << " " << lastname << endl;
}

void Person::set_firstname(string fn)
{
    firstname=fn;
}

void Person::set_lastname(string ln)
{
    lastname=ln;
}

string Person::get_firstname()
{
    return firstname;
}

string Person::get_lastname()
{
    return lastname;
}
```

```
#ifndef _STUDENT_
#define _STUDENT_
#include "person.h"

class Student : public Person
{
private:
    string ID;
public:
    void set_ID(string id);
    string get_ID();
};
#endif
```

```
#include "student.h"

void Student::set_ID(string id)
{
    ID=id;
}

string Student::get_ID()
{
    return ID;
}
```

```
using namespace std;

#include <iostream>
#include "student.h"

int main()
{
    Student *amy = new Student();
    amy->set_firstname("Amy");
    amy->set_lastname("Chang");
    amy->set_ID("s111418099");
    Student *tony = new Student(*amy);
    tony->set_firstname("Tony");
    amy->showInfo();
    tony->showInfo();

    delete amy;
    delete tony;
    return 0;
}
```

```
}
```

```
all: person.o student.o
    g++ main.cpp person.o student.o -o main

person.o: person.cpp person.h
    g++ -c person.cpp
student.o: student.cpp student.h
    g++ -c student.cpp

clean:
    rm -f *.o main *.~*~ *
```

From:

<https://junwu.nptu.edu.tw/dokuwiki/> - Jun Wu的教學網頁

國立屏東大學資訊工程學系

CSIE, NPTU

Total: 173721

Permanent link:

<https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=c++:classexamples:person4>



Last update: **2022/05/05 16:09**