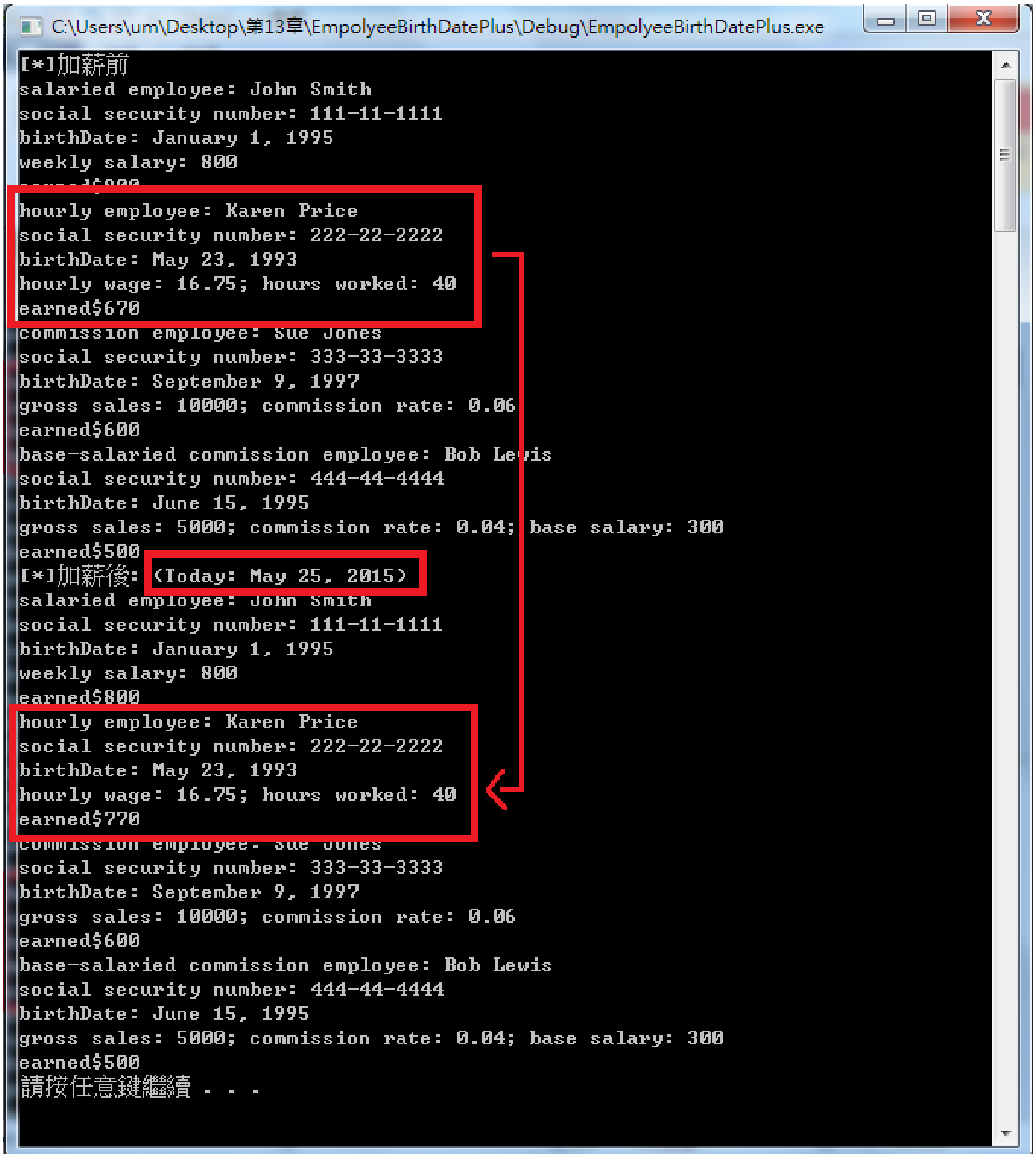
A3261576 林郁翔 說明檔

共兩題:13-12,13-13

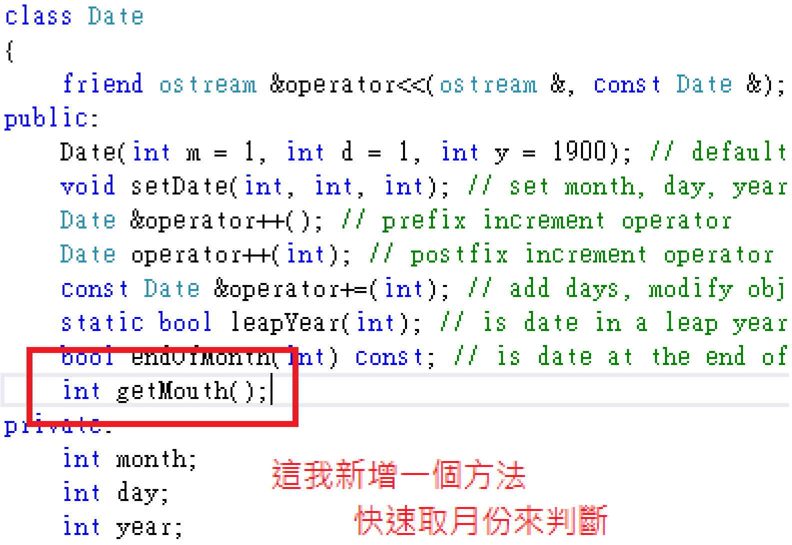
13-12:

解說:這題我把 2015 年 5 月 25 日當作判斷加薪的依據，若生日在該月則加入 $100 元進該員工的薪資如下:


C:\Users\um\Desktop\第13章\EmpolyeeBirthDatePlus\Debug\EmpolyeeBirthDatePlus.exe

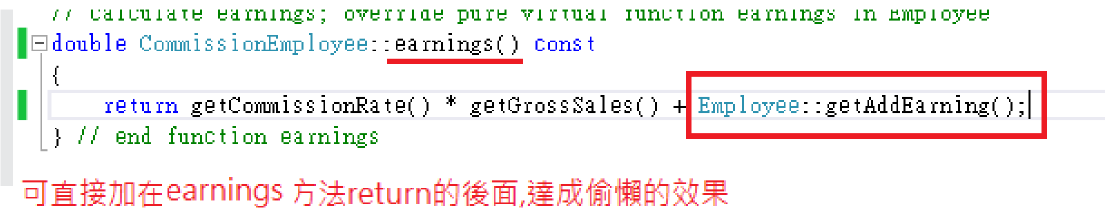
```
[*]加薪前
salaried employee: John Smith
social security number: 111-11-1111
birthDate: January 1, 1995
weekly salary: 800
earned$800
hourly employee: Karen Price
social security number: 222-22-2222
birthDate: May 23, 1993
hourly wage: 16.75; hours worked: 40
earned$670
commission employee: Sue Jones
social security number: 333-33-3333
birthDate: September 9, 1997
gross sales: 10000; commission rate: 0.06
earned$600
base-salaried commission employee: Bob Lewis
social security number: 444-44-4444
birthDate: June 15, 1995
gross sales: 5000; commission rate: 0.04; base salary: 300
earned$500
[*]加薪後: (Today: May 25, 2015)
salaried employee: John Smith
social security number: 111-11-1111
birthDate: January 1, 1995
weekly salary: 800
earned$800
hourly employee: Karen Price
social security number: 222-22-2222
birthDate: May 23, 1993
hourly wage: 16.75; hours worked: 40
earned$770
commission employee: Sue Jones
social security number: 333-33-3333
birthDate: September 9, 1997
gross sales: 10000; commission rate: 0.06
earned$600
base-salaried commission employee: Bob Lewis
social security number: 444-44-4444
birthDate: June 15, 1995
gross sales: 5000; commission rate: 0.04; base salary: 300
earned$500
請按任意鍵繼續 . . .
```

我在 Date 新增一個方法能夠取得月份



```
class Date
{
    friend ostream &operator<<(ostream &, const Date &);
public:
    Date(int m = 1, int d = 1, int y = 1900); // default
    void setDate(int, int, int); // set month, day, year
    Date &operator++(); // prefix increment operator
    Date operator++(int); // postfix increment operator
    const Date &operator+=(int); // add days, modify obj
    static bool leapYear(int); // is date in a leap year
    bool endOfMonth(int) const; // is date at the end of
    int getMouth();
private:
    int month;
    int day;
    int year;
```

這我新增一個方法
快速取月份來判斷

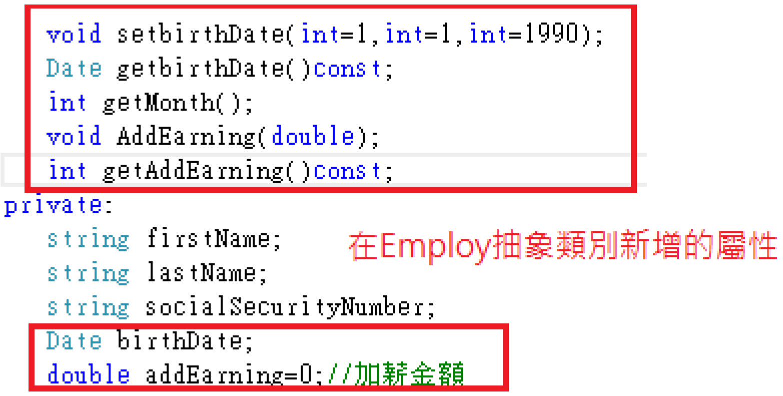


```
// calculate earnings; override pure virtual function earnings in Employee
double CommissionEmployee::earnings() const
{
    return getCommissionRate() * getGrossSales() + Employee::getAddEarning();
} // end function earnings
```

可直接加在earnings 方法return的後面,達成偷懶的效果

這裡我原本想法是把const去掉,然後在把SET 跟 GET寫在earnings 裡面然後宣告一個earning來存他,再用這earning去加100,不過太麻煩了,這樣寫感覺比較輕鬆

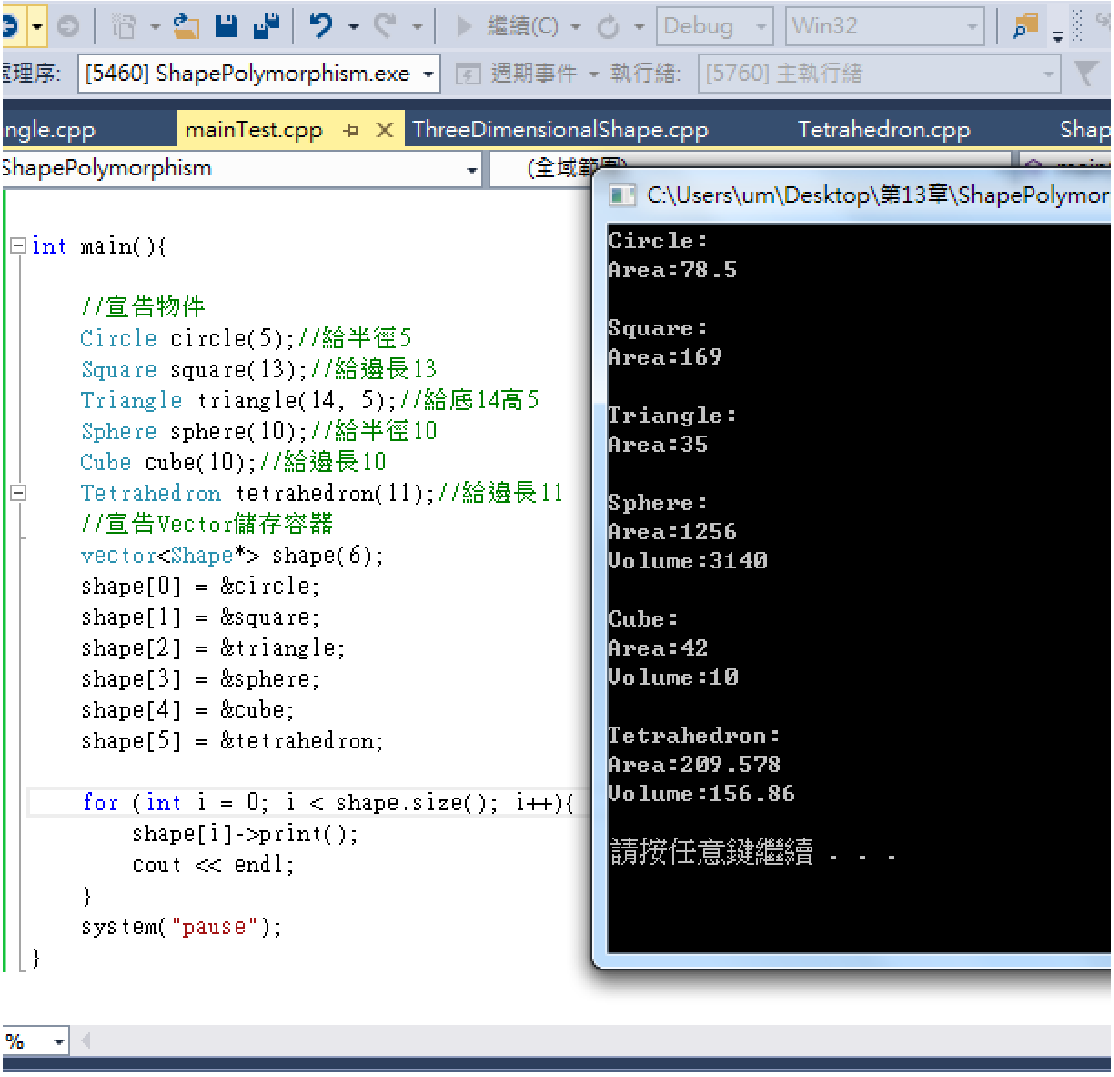
當然其衍生類別的4個earnings都要加 這裡舉一圖作範例



```
    void setbirthDate(int=1,int=1,int=1990);
    Date getbirthDate()const;
    int getMonth();
    void AddEarning(double);
    int getAddEarning()const;
private:
    string firstName;
    string lastName;
    string socialSecurityNumber;
    Date birthDate;
    double addEarning=0;//加薪金額
```

在Employ抽象類別新增的屬性

13-13



```cpp
int main(){

    //宣告物件
    Circle circle(5);//給半徑5
    Square square(13);//給邊長13
    Triangle triangle(14, 5);//給底14高5
    Sphere sphere(10);//給半徑10
    Cube cube(10);//給邊長10
    Tetrahedron tetrahedron(11);//給邊長11
    //宣告Vector儲存容器
    vector<Shape*> shape(6);
    shape[0] = &circle;
    shape[1] = &square;
    shape[2] = &triangle;
    shape[3] = &sphere;
    shape[4] = &cube;
    shape[5] = &tetrahedron;

    for (int i = 0; i < shape.size(); i++){
        shape[i]->print();
        cout << endl;
    }
    system("pause");
}
```

```
Circle:
Area:78.5

Square:
Area:169

Triangle:
Area:35

Sphere:
Area:1256
Volume:3140

Cube:
Area:42
Volume:10

Tetrahedron:
Area:209.578
Volume:156.86

請按任意鍵繼續 . . .
```

這題就列出彼此繼承關係然後，然後用繼承寫出來

然後我有三個抽象類別(TwoDimensionalShape, ThreeDimensionalShape,Shape)和六個具體類別(即 ScreenShot 上的)