

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

## Object Oriented Programming

البرمجة الشيئية

عدد الساعات: ٢ نظري + ٢ عملي

المتطلبات السابقة: ١١٢ حسب

المادة: م. نجلاء حسن

❖ إنشاء ملفات الترويسة header  
files وكيفية استخدامها

- ▶ ملفات تحفظ بامتداد \*.h , \*.hpp , \*.rh
- ▶ توضع بها طبقات أو دوال أو متغيرات أو أي معرفات اخري.
- ▶ تستدعي داخل ملف .cpp باستخدام # include "header file"
- ▶ يتم التعامل مع العناصر الموجودة فيها كما لو كانت موجودة في ملف .cpp

مثال ١:

انشاء ملف رأس emp1 يحتوى على الطبقة .employee

التعامل مع employee في الدالة الرئيسية:

انشاء ملف .cpp و استدعاء ملف الرأس داخله # include "empl.h"

انشاء كائن من نوع employee داخل الدالة الرئيسية ثم ملئة بالبيانات و طباعتها.

```

C:\bc5\bin\empl.h *
class employee
{
    public:
        int empno;
        float empsal;
        char empname[20];
        employee() {}
        void print ()
        {
            cout<<empno<<"    "<<empname<<"    "<<empsal<<endl;
        }
};

```



```
#include<iostream.h>
```

```
#include"emp1.h" ← Calling header file
```

```
void main()
```

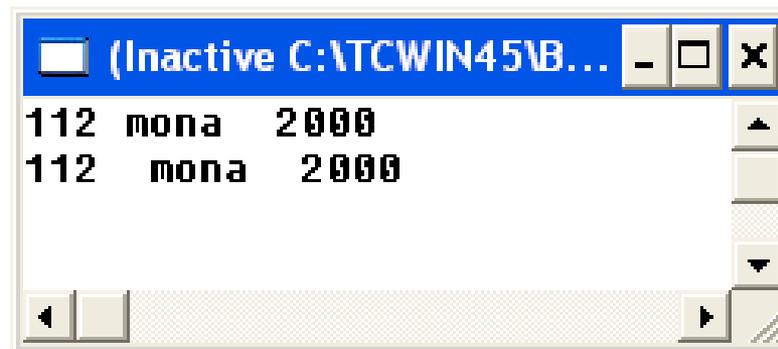
```
{
```

```
employee e1;
```

```
cin>>e1.empno>>e1.empname>>e1.empsal;
```

```
e1.print();
```

```
}
```



## اصنع مكتباتك الخاصة:-

تحتوي المكتبة على فئات وتراكيب ولا تحتوي على أوامر تنفيذية مثل الطباعة على الشاشة (دالة main)، وبالتالي لا يمكن تنفيذ المكتبة ولكن يمكن استدعاؤها في ملف source file. ثم تعريف بيانات من نوع الطبقات الموجودة في المكتبة.

الفائدة من استخدام المكتبات أو الملفات المنفصلة:

- أن التوابع التي تقوم بكتابتها لن تضيع وستقوم باستخدامها مرات ومرات كثيرة ،

- عندما يصبح البرنامج الذي تقوم بكتابته كبيراً فلن تضطر عند التعديل إلا إلى إعادة ترجمة الملف الذي تم التعديل فيه وليس كل البرنامج.

```
int add(int a, int b)
{
    return a + b;
}
```

مثال ٢

may be declared (with a function prototype) and then referred to in a second source file as follows:

```
#include "add.h"
int triple(int x)
{
    return add(x, add(x, x));
}
```