

المحاضرة الثانية

الأنصاف

Classes

يعد الصنف في لغة C++ من أهم مميزات اللغة والتي تجعل منها لغة OOP وعليه فان الصنف هو عبارة عن تركيب يحوي بيانات خاصة وعامة (معطيات) كما يحتوي الصنف على تعاريف للدوال حيث يطلق على الدوال المعرفة ضمن تركيب الصنف أعضاء دالية **Members Function** ومن اجل استخدام الصنف **Class** لابد من تعريفه مسبقاً مثل تعريف المتغيرات

يحتوي الصنف علي:

1. البيانات الأعضاء :

وهي خصائص الصنف .

2. الدوال الأعضاء :

هي الوظائف المعرفة على الصنف و تقوم بمعالجة خصائص الصنف.

الصيغة العامة لإنشاء صنف :

	اسم الصنف
كلمة محجوزة	class
بداية الصنف	{
متغير محلي	data type element1 ;
متغير محلي	data type element1 ;
	public:
متغير عام	data type element3 ;
تعاريف الدوال	data type Function1(Data type Parameters);
	data type Function2(Data type Parameters);
نهاية الصنف	};

مثال

```
class rectangle
{
    int x , y ;
public:
    void set_values(int , int);
    int area ( );
};
```

هو عبارة عن إعلان عن صنف **Class** باسم **rectangle** هذا الصنف يحتوي على أربعة أعضاء **Four Members**

1. متغيرين من النوع الصحيح (x, y) وتم تعريفهم في الجزء الخاص بالمتغيرات المحلية

Two variables of type int (x and y) in the Private section

2. دالتين من النوع العام Set_values() and area()

تعريف الكائنات:-

يتم تعريف الكائنات في البرنامج الرئيسي وفقاً للصيغة التالية:-

Class name Object name ;

اسم الكائن اسم الصنف

مثلاً إذا أردنا تعريف الكائن rect من نوع الصنف rectangle نكتب الآتي:-

rectangle rect ;

أما إذا أردنا تعريف مجموعة كائنات من نوع صنف معين فيجب أن يفصل بين كل كائن وآخر بواسطة فاصلة(,)

مثلاً إذا أردنا تعريف الكائنات rect1 و rect2 و rect3 من نوع الصنف rectangle نكتب الآتي:-

rectangle rect 1,rect2,rect3 ;

يتم الوصول إلى الأعضاء خارج الصنف باستخدام الكائنات مستخدمين في ذلك أداة النقطة (.)	<input checked="" type="checkbox"/>
---	-------------------------------------

وفقاً للصيغة التالية:-

Object name. member ;

مثلاً إذا كان لدينا الكائن rect من نوع الصنف rectangle فيمكن استخدامه للوصول للدالة area()

كالآتي :- **rect. area() ;**

البرنامج الأول:-

برنامج يقوم بحساب مساحة المستطيل باستخدام الأصناف

خصائص الصنف (البيانات):-

1. الطول x

2. العرض y

العمليات المعرفة على الصنف (الدوال)

1. دالة تحديد القيم set_values()

2. دالة حساب المساحة area()

نص البرنامج :-

```
#include<iostream.h>
class crectangle
{
private:
int x,y;
public:
void set_values(int a,int b)
{
x=a;
y=b;
}
int area( )
{
return x*y;
}
};
int main( )
{
crectangle rect,rectb;
rect.set_values(3,4);
rectb.set_values(5,6);
cout<<"rect area ="<<rect.area( )<<endl;
cout<<"rectb area ="<<rectb.area( )<<endl;
return 0;

}
```

Output

```
rect area =12
rectb area =30
```

البرنامج الثاني:-

إنشاء صنف لدائرة

خصائص الصنف :

1. نصف القطر r
2. المحيط cir
3. المساحة area

العمليات المعرفة على الصنف :

1. إدخال نصف القطر (get_r())
2. إيجاد المساحة (cal_area())
3. إيجاد المحيط (cal_cir())
4. طباعة المساحة والمحيط (display())

```
#include<iostream.h>
class circle
{
private :
int r ;
float cir,area;
public :
void get_r( )
{
    cout << "Enter radius :";
    cin >> r ;
}
void cal_area( )
{
    area = r * r * 3.14 ;
}
void cal_cir( )
{
    cir = 2 * r * 3.14 ;
}
void display( )
{
    cout << "area ="<<area<<"\n cir ="<< cir ;
}
};
int main( )
{
    circle c1 ;
    c1.get_r( );
    c1.cal_area( );
    c1.cal_cir( );
    c1.display( );
    return 0;
}
```

مثال:-

برنامج يقوم بحساب مضروب العدد **n** باستخدام الأصناف

```
#include<iostream.h>
class factorial
{
    int i , f, fact ;
public:
    factorial(int n)
    {
        fact=1;
        for (i=1;i<=n;i++)
        {
            fact=fact*i;
        }
        cout<<fact<<"\n";
    }
};
int main( )
{
    factorial    fact1(5),fact2(4);
    return 0;
}
```

مؤثر محدد دقة الوصول (::) scope Resolution Operator

هو عبارة عن مؤثر يستخدم عندما تكتب تفاصيل الدالة خارج نطاق الصنف

الصيغة العامة لكتابة تفاصيل الدوال الأعضاء خارج الصنف

اسم الدالة اسم الصنف النوع

Data type Class_Name::Function_name(Function arguments)

مثال توضيحي لمؤثر دقة الوصول:-

```
#include<iostream.h>
class myclass
{
    int a;
public:
    void set_a(int);
    int get_a( );
};
void myclass::set_a(int num)
{
```

```

a=num;
}
int myclass::get_a( )
{
return a;
}
int main( )
{
int x,y;
cout<<"enter x \n";
cin>>x;
cout<<"enter y \n";
cin>>y;
myclass ob1,ob2;
ob1.set_a(x);
ob2.set_a(y);
cout<<ob1.get_a( )<<"\n";
cout<<ob2.get_a( )<<"\n";
return 0;
}

```

الرجوع في الصنف للدوال الأعضاء مشابه لرجوع الكائن إلى المتغيرات الأعضاء

على سبيل المثال:-

```

#include<iostream.h>
class myclass
{
public:
int a;
float b;
};
int main( )
{
myclass ob1,ob2;
ob1.a=10;
ob2.b=4.5;
cout<<ob1.a<<"\n";
cout<<ob2.b<<"\n";
return 0;
}

```

```

#include<iostream.h>
class student
{
    private :
        int index ;
        char name [20];
    public :
        void set_name ( char * );
        char * get_name( );
};
void student :: set_name ( char * s )
{
    strcpy (name , s );
}
char * student :: get_name( )
{
    return name ;
}
int main( )
{
    student s1,s2 ;
    s1.set_name("Ali");
    s2.set_name("Hasan");
    cout << s1.get_name( )<<" ";
    cout << s2.get_name( );
    return 0;
}

```