

الواجهات (Interfaces)

مقرر برمجة متقدمة - ١ (عملي)
المحاضرة الثانية
آ. كنانة الحلواني

مفهوم الواجهات (Interfaces)

➤ من المعروف أنه في لغة c# لا يمكن تطبيق مفهوم الوراثة المتعددة (أي كل صف ابن يجب يرث من صف أب واحد فقط)، و لحل هذه المشكلة تم إيجاد الواجهات (Interfaces).

➤ الواجهة تشبه الصف و لكنها تحوي فقط التصريح عن الخصائص و الطرائق و ليس تعريف كامل لها.

➤ يتم تعريف الواجهة في c# كما يلي:

```
namespace ConsoleApplication4
{
    interface myInterface
    {

    }

    class myClass : myInterface
    {

    }
}
```

مفهوم الواجهات (Interfaces)

- الواجهة لا تحوي أي تعليمة برمجية، و إنما فقط التصريح عن التوابع () أو الخاصيات.
- لا يمكن التصريح عن الحقول (المتحولات) ضمن الواجهة.
- لا يمكن تحديد معرفات وصول (access modifiers) ضمن الواجهة و إنما يتم تحديد معرف الوصول عند تعريف التابع أو الخاصية في الصف الابن، و يجب أن يكون public .
- لا يمكن انشاء أغراض من الواجهات.
- يمكن للصف ان يرث من أكثر من واجهة.
- يمكن للواجهة أن ترث من واجهة أخرى، و في هذه الحالة الصف الابن يجب أن يطبق كل الطرائق و الخاصيات الموجودة في كلا الواجهتين.
- لا يمكن للواجهة أن ترث من صف..
- عندما يرث الصف من صف اخر و عدة واجهات، يجب دائماً أن نكتب الصف قبل الواجهات في عملية الوراثة.

تطبيق عملي (1)

- ليكن لدينا واجهة لتمثيل بيانات وجبة بيتزا باسم `Ipizza` تتضمن طريقة لطباعة معلومات الطلب.
- و صف لتمثيل بيانات وجبة سريعة باسم `FastFood` يتضمن اسم الوجبة `Food_name` و عدد السعرات الحرارية `calories` فيها.
- و صف لتمثيل بيانات نوع من أنواع البيتزا باسم `pepperoniPizza` يتضمن نفس بيانات الصف و الواجهة السابقين.
- قم بإنشاء البنية البرمجية السابقة محدداً العلاقة بينها ان وجدت، ثم قم بإنشاء غرض من الصف الأخير في التابع الرئيسي `main` و قم باسناد قيم للمتحولات و استدعاء التوابع .

حل تطبيق عملي (1)

```
interface Ipizza
{
    int Price { get; }
    void order();
}
class fastfood
{
    string foodname;
    int calories;
    public string Foodname {
        set { foodname = value; }
        get { return foodname; } }
    public int Calories {
        set { calories = value; }
        get { return calories; } }
}
class peppronipizza :fastfood, Ipizza
{
    int price;
    public int Price { set { price = value; } get { return price; } }
    public void order()
    {
        Console.WriteLine("pepproni pizza order");
    }
}
```

حل تطبيق عملي (1)

```
class Program
{
    static void Main(string[] args)
    {
        peppronipizza pizza = new peppronipizza();
        pizza.Calories = 120;
        pizza.Foodname = "pepproni";
        pizza.order();
    }
}
```

تطبيقات عملي (2)

➤ لدينا الواجهات التالية :

- واجهة لتمثيل بيانات مركبات باسم Ivehicle تتضمن: خاصية للموديل، و خاصية لتمثيل السرعة القصوى، و تابع يعيد السرعة القصوى على شكل رسالة للمستخدم.
- و لدينا واجهة أخرى IBicycle تمثل بيانات الدراجات الهوائية، تتضمن : خاصية لاسم الشركة المصنعة.

➤ و لدينا الصفوف التالية:

- صف motion يمثل بيانات احدى شركات الدراجات النارية يتضمن حقل لاسم الدولة (قابل للقراءة فقط).
- صف car يمثل بيانات سيارات يتضمن نفس محتوى الواجهة Ivehicle.
- صف Motorcycle يمثل بيانات الدراجات النارية يتضمن نفس محتوى الواجهات Ivehicle ، IBicycle و الصف motion
- قم بإنشاء البيئة البرمجية السابقة ثم قم بإنشاء غرض في الصفين الأخيرين في التابع الرئيسي main و قم بإسناد قيم للحقول و استدعاء التوابع.

حل تطبيق عملي (2)

1

```
interface Ivehicle // واجهة المركبات
{
    string Model { set; get; }
    int MaxSpeed { set; get; }
    string displaySpeed();
}
interface IBicycle // واجهة الدراجات الهوائية
{
    string Manufacture;
}
class motion // صف الدراجات النارية
{
    private string country="UK";
    public string Country
    {
        get { return country; }
    }
}
```

2

```
class car : Ivehicle // صف السيارات
{
    private string model;
    private int maxspeed;
    public string Model
    {
        set { model = value; }
        get { return model; }
    }
    public int MaxSpeed
    {
        set { maxspeed = value; }
        get { return maxspeed; }
    }
    public string displaySpeed()
    {
        return "The maximum speed is:" + maxspeed;
    }
}
```


3

// صف الدراجات النارية يرث من صف و واجهتين

```

class Motocycle : motion, Ivehicle, IBicycle
{
    private string model;
    private int maxspeed;
    private string manufacture;
    public string Model
    {
        set { model = value; }
        get { return model; }
    }
    public int MaxSpeed
    {
        set { maxspeed = value; }
        get { return maxspeed; }
    }
    public string Manufacture
    {
        set { manufacture = value; }
        get { return manufacture; }
    }
    public string displaySpeed()
    {
        return "The maximum speed is:" + maxspeed;
    }
}

```

4

class Program

```

{
    static void Main(string[] args)
    { // اشتقاق اغراض من الصفين
        car c = new car();
        c.MaxSpeed = 200;
        c.Model = "2016";
        Console.WriteLine( c.displaySpeed());

        Motocycle m = new Motocycle();
        m.Manufacture = "Ford";
        m.MaxSpeed = 250;
        m.Model = "2017";
        Console.WriteLine( m.displaySpeed()+m.Country);
    }
}

```

😊 انتهت المحاضرة 😊

😊 مساحة لاستفساراتكم 😊