Day 6

StringBuilder

文本

描述已自动生成

access modifier

图形用户界面

低可信度描述已自动生成

assembly means project

private and getter and setter

图形用户界面

描述已自动生成

constructors

文本

描述已自动生成

inherit

文本

描述已自动生成

create object

图形用户界面, 文本

描述已自动生成

abstraction文本

描述已自动生成  


图形用户界面, 文本, 应用程序

描述已自动生成

virtual keyword



在C#中，使用virtual关键字可以定义虚拟方法，这与普通方法有以下区别：

1. 可以在派生类中重写：虚拟方法可以在派生类中重写，子类可以根据需要提供自己的实现，而普通方法不能被子类重写。
2. 提供默认实现：虚拟方法可以在基类中提供默认实现，如果子类没有重写该方法，则将使用基类中的默认实现。普通方法不提供默认实现。
3. 运行时多态：虚拟方法支持运行时多态，即在运行时根据对象的实际类型调用相应的方法。普通方法在编译时就已经决定了调用哪个方法，无法实现运行时多态。

如果一个派生类重写了其基类中的非虚拟方法，则称为“隐藏”（Hiding）基类中的该方法。隐藏方法与重写虚拟方法的方式是不同的。如果派生类重写了基类中的虚拟方法，那么在运行时将调用派生类中的实现，而如果派生类隐藏了基类中的非虚拟方法，则在编译时将始终调用隐藏的方法，不会调用基类中的实现。

interface文本

描述已自动生成

电脑萤幕画面

描述已自动生成

implement two interfaces with same function

文本

描述已自动生成

boxing  and unboxing. upcasting

文本

描述已自动生成

hiding

如果一个派生类重写了其基类中的非虚拟方法，则称为“隐藏”（Hiding）基类中的该方法。隐藏方法与重写虚拟方法的方式是不同的。如果派生类重写了基类中的虚拟方法，那么在运行时将调用派生类中的实现，而如果派生类隐藏了基类中的非虚拟方法，则在编译时将始终调用隐藏的方法，不会调用基类中的实现。

图形用户界面, 文本

描述已自动生成

文本

描述已自动生成

factory design pattern

文本

描述已自动生成

static keyword



sealed class cannot be inherited

图形用户界面, 文本, 应用程序

描述已自动生成