# 大三-ASP.NET-T01 C#高级特性-实践任务指南

**训练目标：**

1. **C#的委托的使用**
2. **隐式类型与匿名类型的应用**
3. **C#扩展方法与yield关键字的使用**
4. **Lambda表达式**

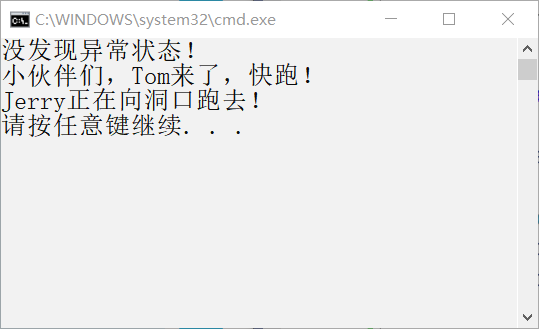
**扩展知识点：**

## 实践任务1：掌握委托的运用

### 建议完成时间：25分钟

### 需求说明：

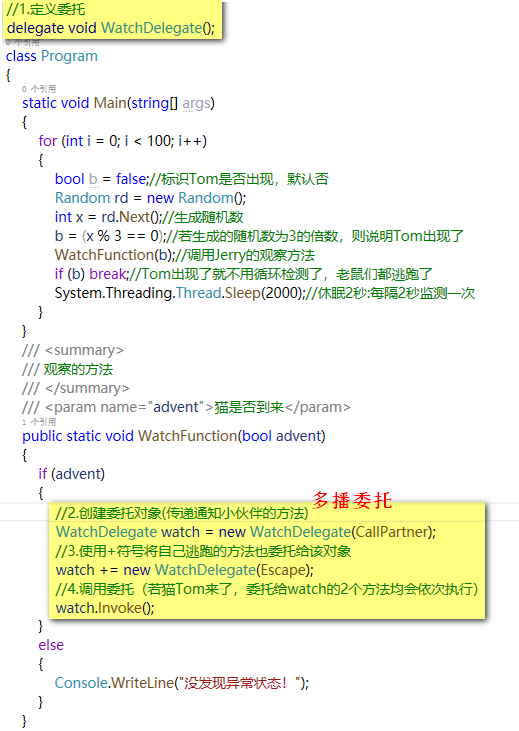
请用委托实现该场景：有一群老鼠，去某地偷吃食物，其中有一只老鼠（Jerry）放哨，当发现猫（Tom）的时候需要通知其他老鼠逃跑，同时自己也要逃跑。运行结果如图所示：

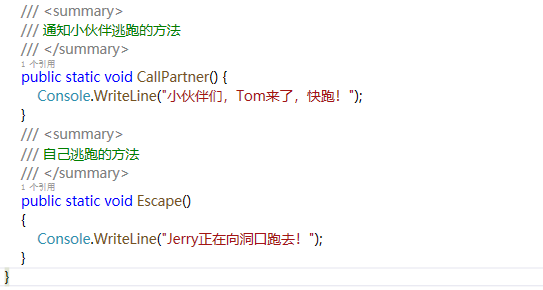


### 实践思路：

1. 编写通知小伙伴的方法和逃跑的方法。
2. 定义一个返回值为void，无参数的委托。
3. 使用多播委托编写放哨的方法，判断Tom是否出现。
4. 在主方法Main方法中随机生成bool类型的值，bool类型的值为true，则说明Tom出现，调用观察方法，同时结束循环，否则继续生成bool类型的值。

参考代码如下：



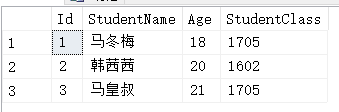


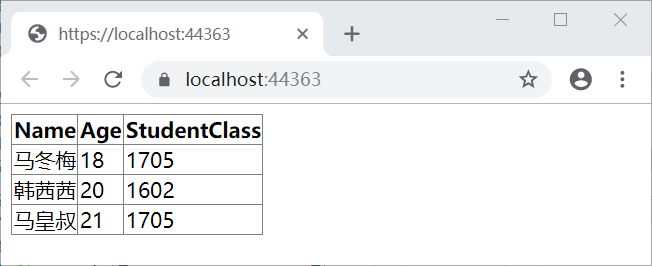
## 实践任务2：掌握委托的运用

### 建议完成时间：20分钟

### 需求说明：

现有一张学生信息表，包含学生的姓名、年龄和班级，请使用ASP.NET Web项目将表中的数据使用GridView控件予以呈现，在编写程序时，不必编写学生信息实体类，使用匿名类解决。数据表和程序运行效果如果所示：

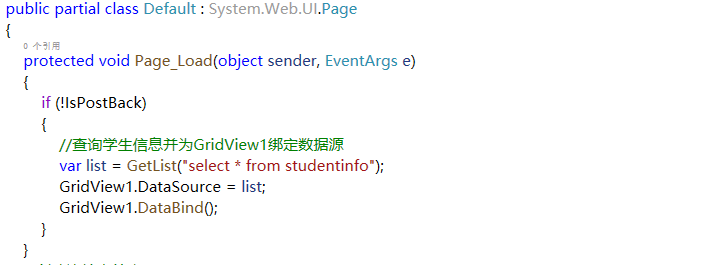


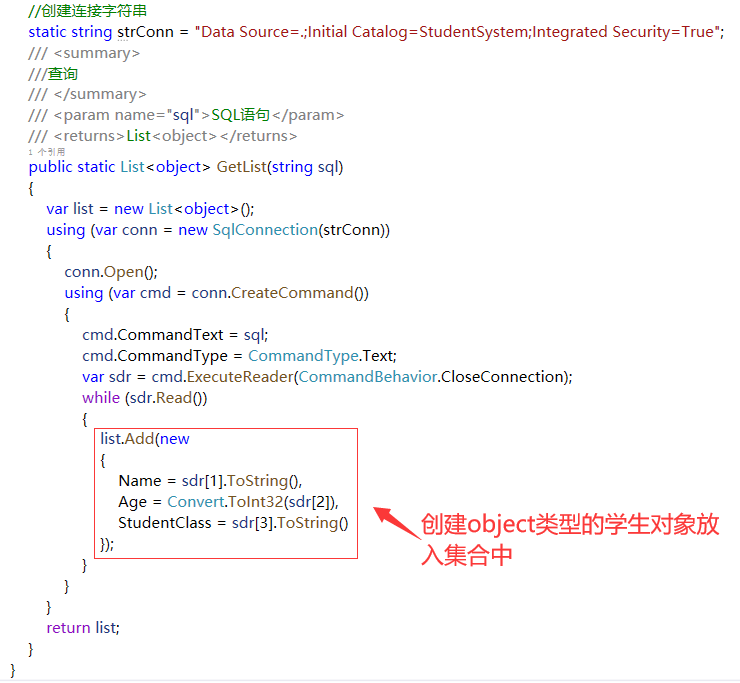


### 实践思路：

1. 找到素材文件StudentSystem.sql，在数据库管理平台执行生成对应的库和表。
2. 编写数据访问类，返回List<object>类型，其中在向List<object>添加数据的使用采用匿名类型完成。
3. 编写Web窗体Default.aspx，并在窗体中添加GridView控件，该控件接受List<object>类型的数据，并将数据通过GridView控件予以展示。

参考代码如下：



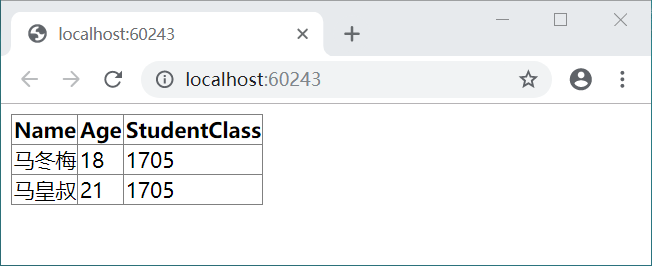


## 实践任务3：使用扩展方法修改任务2

### 建议完成时间：25分钟

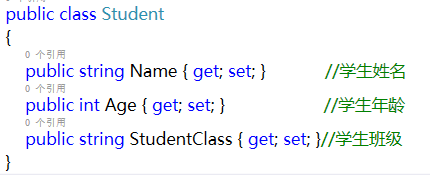
### 需求说明：

针对任务2进行修改，添加对List<Student>的扩展方法，用来根据学生姓名来筛选学生信息。运行结果如图所示：

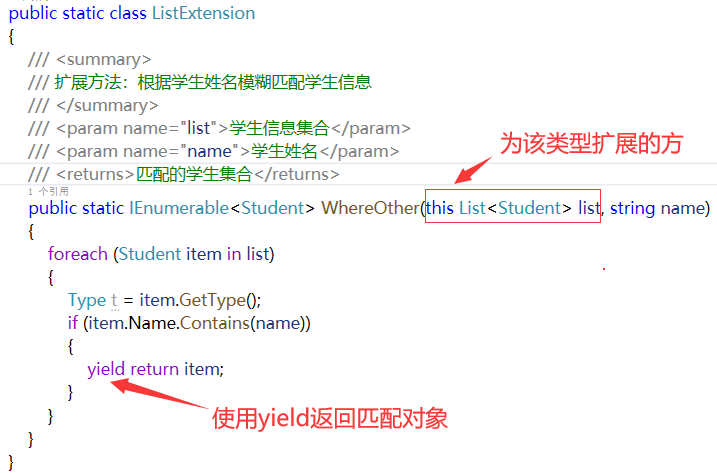


### 实践思路：

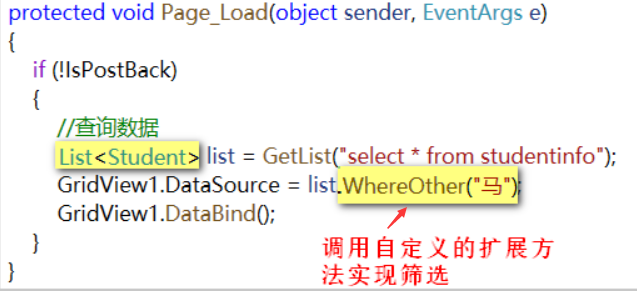
1. 在项目中创建并编写学生实体类型Student，包含学生姓名、年龄、班级信息，代码如下：

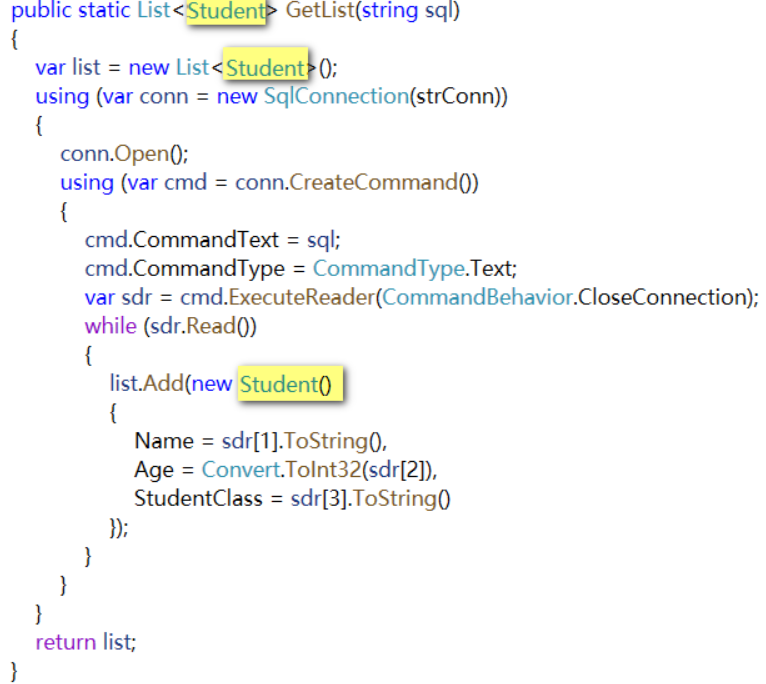


1. 编写数据访问类ListExtension，返回List<Student>类型。
2. 编写List<Student>类型的扩展方法WhereOther，该方法用于根据学生姓名筛选学生信息，使用yield返回IEnumerable<Student>类型数据，代码如下：



1. 编写Web窗体，并在窗体中添加GridView控件，该控件接受List<Student>类型的数据，并将数据通过GridView控件将过滤后的数据予以展示。关键代码如下：



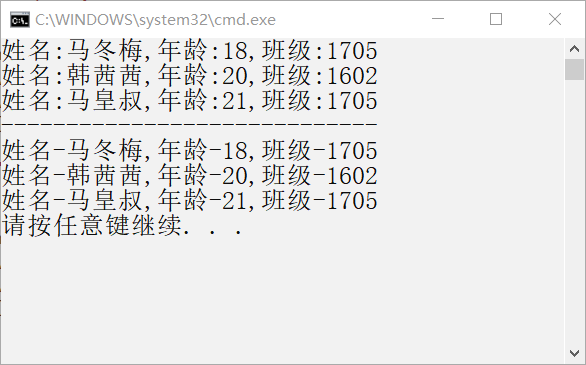


## 实践任务4：使用控制台程序将学生信息输出(委托)

### 建议完成时间：25分钟

### 需求说明：

使用控制台程序将学生信息输出。编写一个数据库访问方法，查询学生表的信息，并将学生信息输出到控制台，要求输出数据的形式由外部传入。运行结果如图所示：



### 实践思路：

1. 定义一个委托，使用委托将方法当做参数来传递。
2. 编写数据库访问方法，接受委托方法，并在该方法中调用委托。
3. 在主方法中调用输出方法，传入SQL语句和Lambda表达式方法，根据输入的表达式方法输出数据。

参考代码如下：

