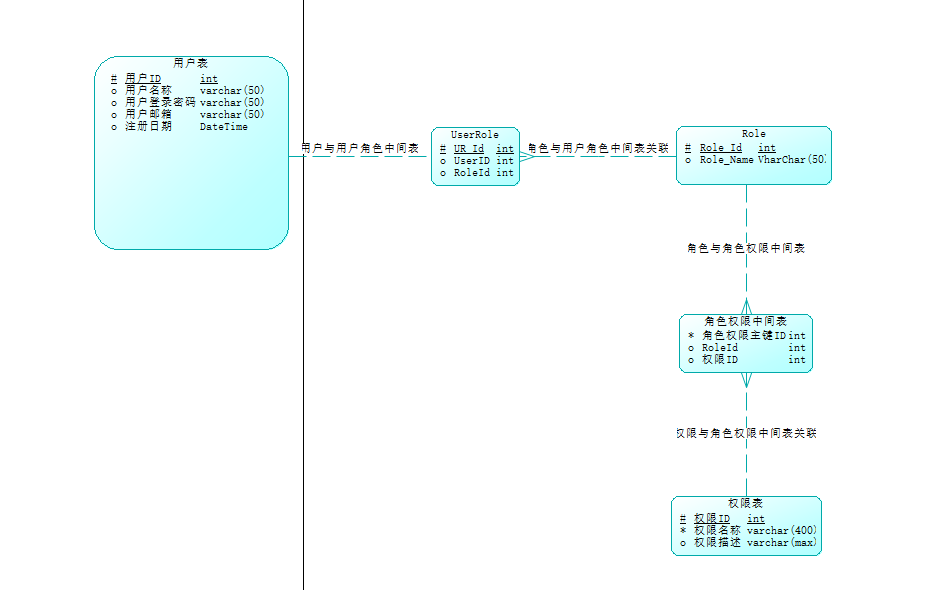
创建用户RBAC基础信息用户与角色多对多关系一个用户可以有多个角色，一个角色可以有多个用户，创建用户角色中间表存放用户ID和角色ID，角色与权限也是多对多的关系，一个角色可以有多个权限，一个权限也可以有多个角色，创建角色权限中间表存放角色ID和权限ID



定义一个泛型IRepository接口实现基础方法



定义一个泛型Repository类继承IRepository接口实现接口方法

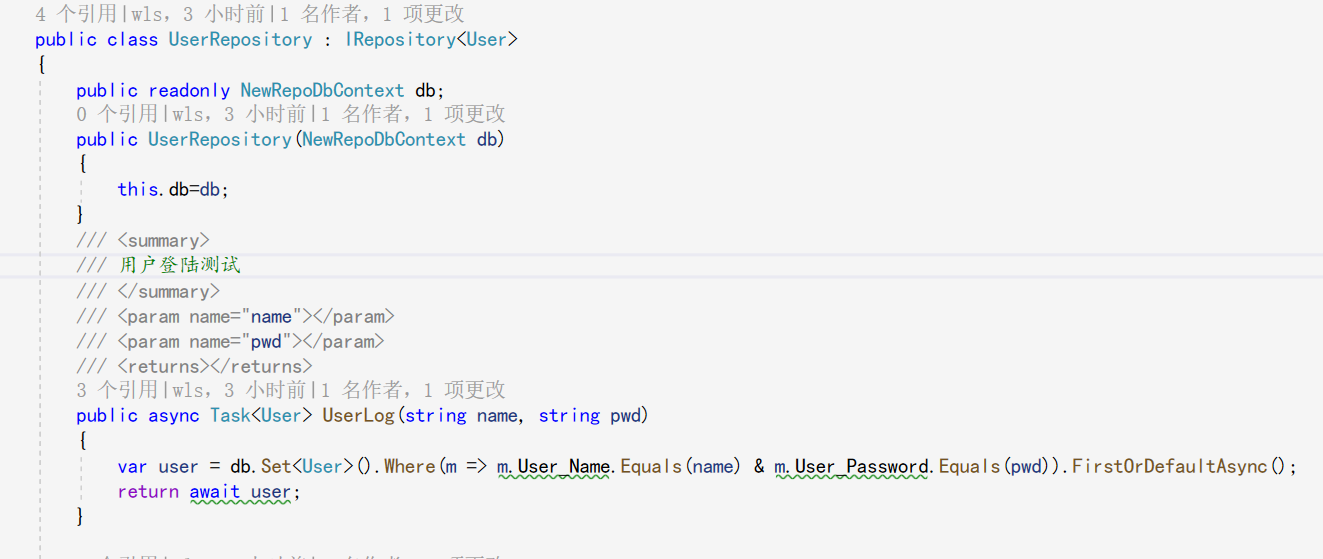


typeof(IRepository<>)：表示要注册的服务类型，即泛型接口IRepository<>。

typeof(Repository<>)：表示服务的具体实现类型，即泛型类Repository<>。由于IRepository<>是一个泛型接口，需要指定其具体实现类型才能进行注册



用户数据访问继承泛型接口实现用户登录功能实现



注入用户数据访问层，调用用户登录方法实现账户登录测试功能



**以下是百度的什么是泛型及泛型约**

泛型:即通过参数化类型来实现在同一份代码上操作多种数据类型。泛型编程是一种编程范式，它利用“参数化类型”将类型抽象化，从而实现更为灵活的复用。

****泛型约束:****

在定义泛型类时，可以对客户端代码能够在实例化类时用于类型参数的类型种类施加限制。如果客户端代码尝试使用某个约束所不允许的类型来实例化类，则会产生编译时错误。这些限制称为约束。约束是使用 where 上下文关键字指定的。

泛型有五个约束:

|  |  |
| --- | --- |
| T：struct | 类型参数必须是值类型。可以指定除 Nullable 以外的任何值类型。 |
| T：class | 类型参数必须是引用类型，包括任何类、接口、委托或数组类型。 |
| T：new() | 类型参数必须具有无参数的公共构造函数。当与其他约束一起使用时，new()  约束必须最后指定。 |
| T：<基类名> | 类型参数必须是指定的基类或派生自指定的基类。 |
| T：<接口名称> | 类型参数必须是指定的接口或实现指定的接口。可以指定多个接口约束。  约束接口也可以是泛型的。 |
| T：U | 为 T 提供的类型参数必须是为 U 提供的参数或派生自为 U 提供的参数。  这称为裸类型约束. |