学习知识

Fluentnhibernate

Autofac

Automapper

Nhibernate

Mvc

Fluentvalidation

Webapi

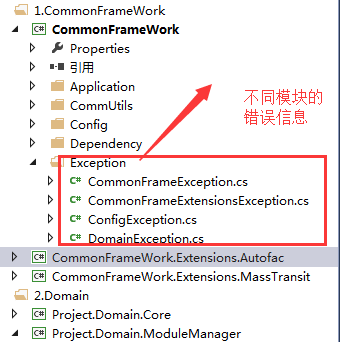
ddd

jquery

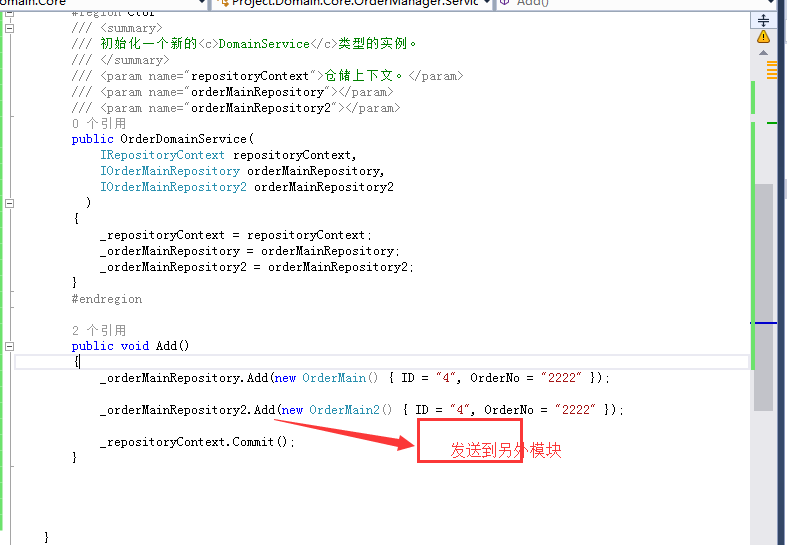
jqueryvalidate

IOC和log是比用的所以直接在初始化的时候确定

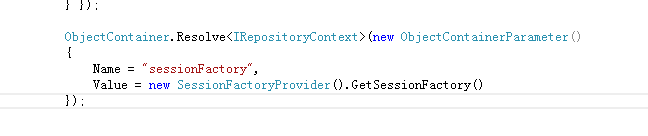
其他通过扩展实现







如果涉及不同的数据库 可以

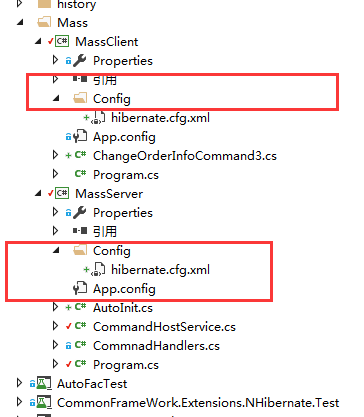


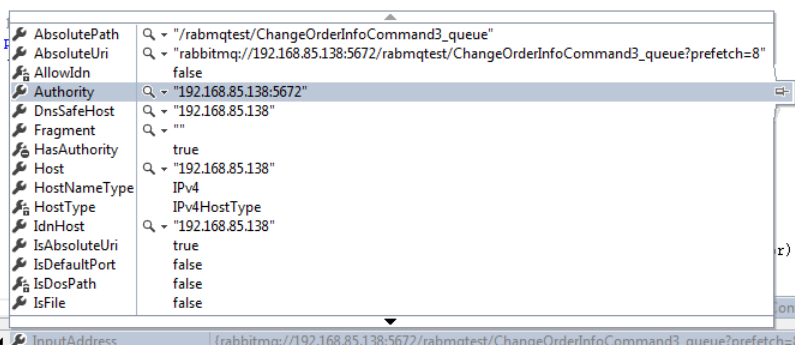
或者新建一个IRepositoryContext

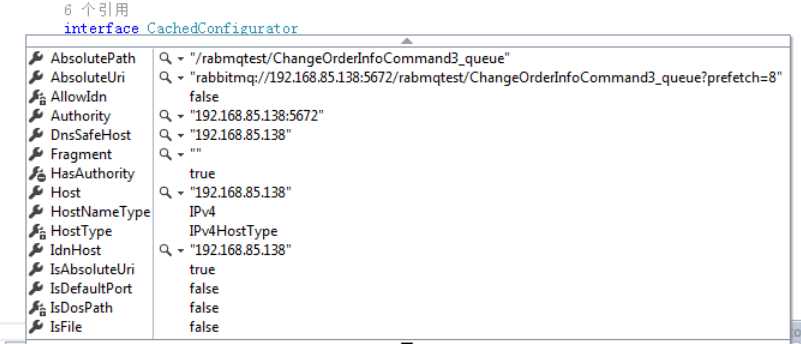
细看

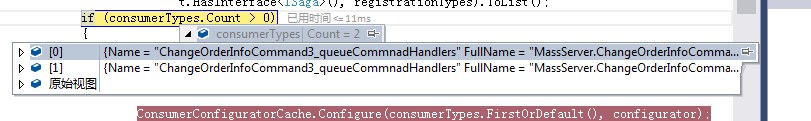


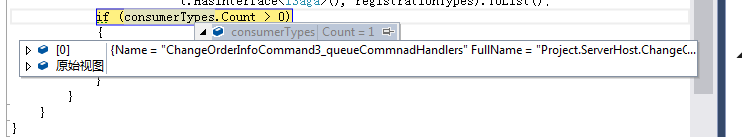
客户端和服务端都得应用nhibernate的配置 有问题！



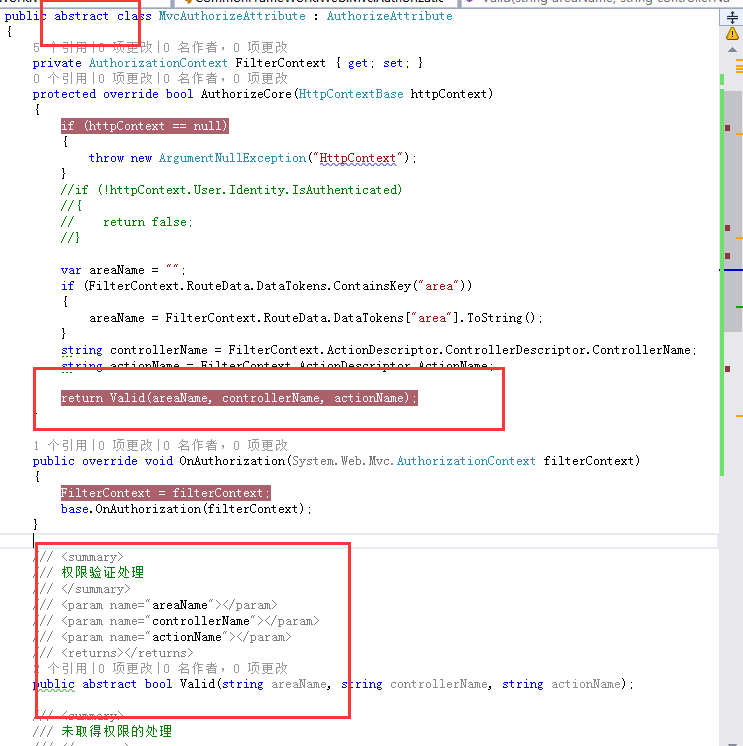








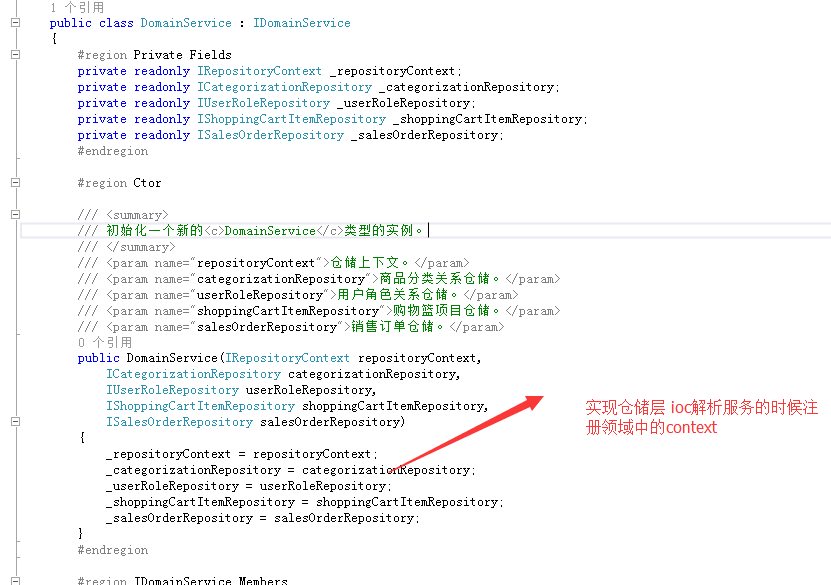
2016.12.1 mvc 权限控制



2016.12.8

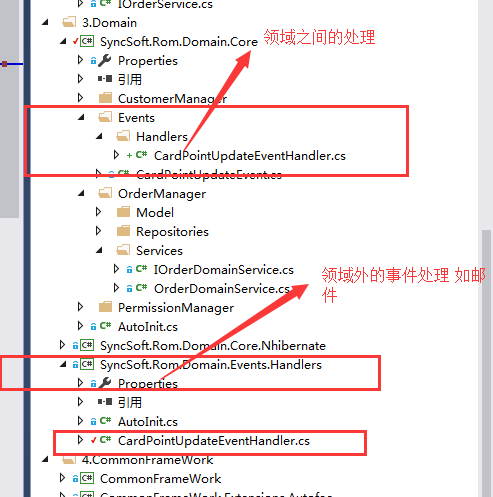
支付平台和账号平台统一用rsa做签名

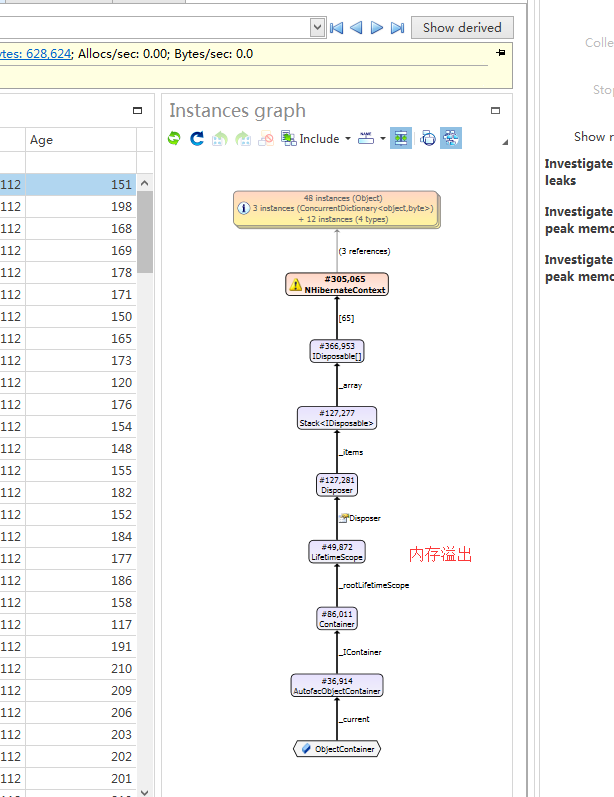
2017.3.1

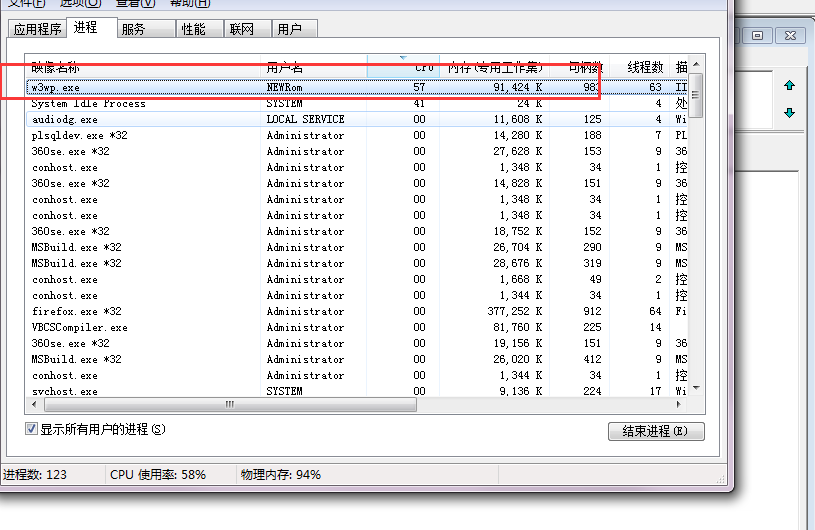


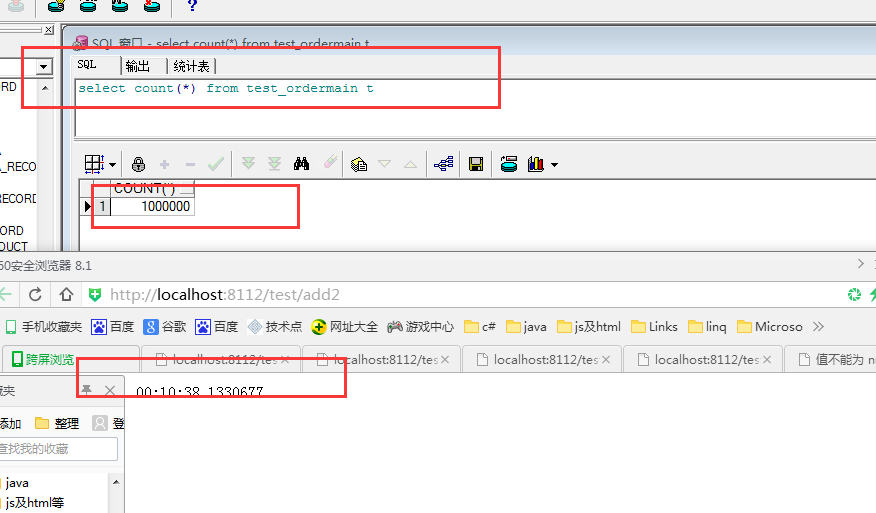


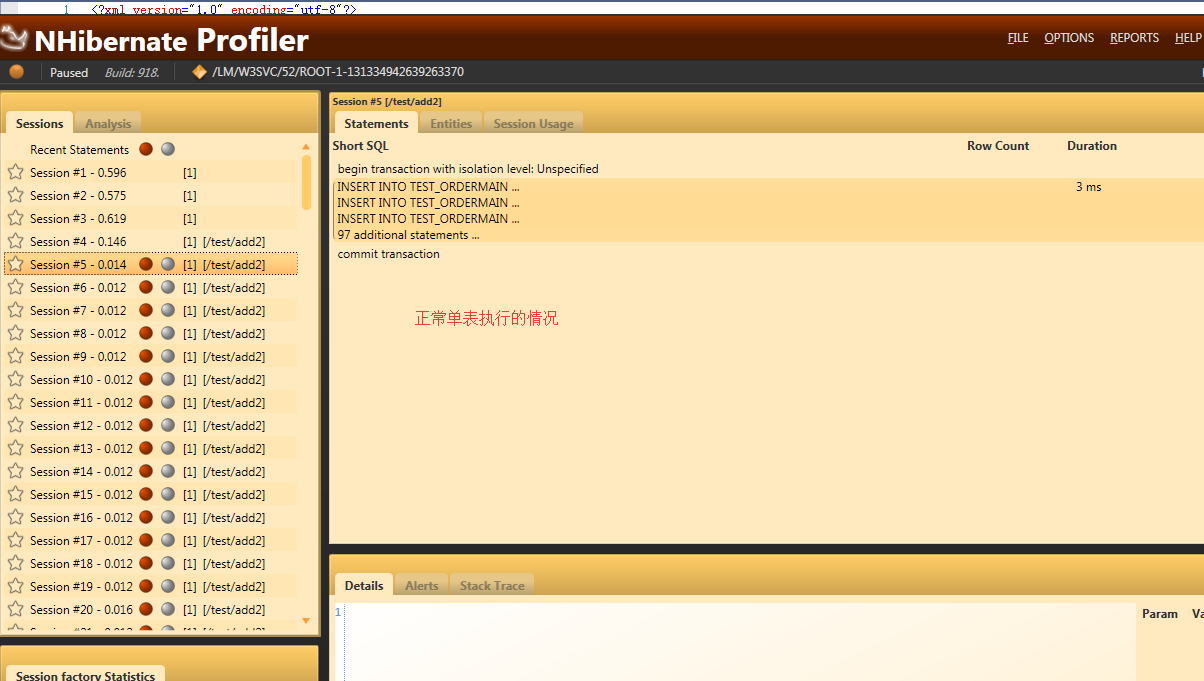
















当初上下文继承dispose导致错误

**Disposal**

Resources obtained within a unit of work - database connections, transactions, authenticated sessions, file handles etc. - should be disposed of when that work is complete. .NET provides theIDisposable interface to aid in this more deterministic notion of disposal.

Some IoC containers need to be told explicitly to dispose of a particular instance, through a method like ReleaseInstance(). This makes it very difficult to guarantee that the correct disposal semantics are used.

* Switching implementations from a non-disposable to a disposable component can mean modifying client code.
* Client code that may have ignored disposal when using shared instances will almost certainly fail to clean up when switched to non-shared instances.

[*Autofac solves these problems using lifetime scopes*](http://docs.autofac.org/en/latest/lifetime/index.html) as a way of disposing of all of the components created during a unit of work.

**using** (**var** scope = container.BeginLifetimeScope()){  
  scope.Resolve<DisposableComponent>().DoSomething();  
  
  *// Components for scope disposed here, at the*  
  *// end of the 'using' statement when the scope*  
  *// itself is disposed.*}

automap学习地址

<https://github.com/AutoMapper/AutoMapper/wiki/Configuration>

多线程并发上下文的绑定问题是否解决

统一所有的日志及错误处理

2017.5.16 统一修改常楚成