ABP框架扩展了ASP.NET Core的分布式缓存系统.

**安装**

使用Abp缓存需要安装如下包

*Install-Package Volo.Abp.Caching*

然后在模块上指定依赖

using Volo.Abp.Modularity;

using Volo.Abp.Caching;

namespace MyCompany.MyProject

{

    [DependsOn(typeof(AbpCachingModule))]

    public class MyModule : AbpModule

    {

        //...

    }

}

**IDistributedCache<TCacheItem>**

Abp定义了泛型接口IDistributedCache<TCacheItem>，如下使用示例

1. 新建缓存类

public class BookCacheItem

{

    public string Name { get; set; }

    public float Price { get; set; }

}

1. 使用缓存

public class BookService : ITransientDependency

{

    private readonly IDistributedCache<BookCacheItem> \_cache;

    public BookService(IDistributedCache<BookCacheItem> cache)

    {

        \_cache = cache;

    }

    public async Task<BookCacheItem> GetAsync(Guid bookId)

    {

        return await \_cache.GetOrAddAsync(

            bookId.ToString(), //Cache key

            async () => await GetBookFromDatabaseAsync(bookId),

            () => new DistributedCacheEntryOptions

            {

                AbsoluteExpiration = DateTimeOffset.Now.AddHours(1)

            }

        );

    }

    private Task<BookCacheItem> GetBookFromDatabaseAsync(Guid bookId)

    {

        //TODO: get from database

    }

}

IDistributedCache<TCacheItem>的观察key是字符串，如果你的key不是字符串，你可以使用IDistributedCache<TCacheItem, TCacheKey>指定你的key类型（其实内部的实现就是调用key的ToString()方法了）

**缓存源注册**

注册方式与Asp.net core相同，你可以参考其注册方式

默认的缓存源为内存分布式缓存