1、安装包：Volo.Abp.BackgroundJobs.Quartz

2、添加Module依赖

[DependsOn(

    //...other dependencies

    typeof(AbpBackgroundJobsQuartzModule)

)]

public class YourModule : AbpModule

{

}

3、新建作业任务

    public class InventoryAlertJob : IJob, ITransientDependency

    {

        protected IServiceProvider Provider { get; }

        public InventoryAlertJob(IServiceProvider provider)

        {

            this.Provider = provider;

        }

        public async Task Execute(IJobExecutionContext context)

        {

            ...

        }

    }

4、配置作业任务

public override void ConfigureServices(

    ServiceConfigurationContext context)

{

    // 添加一个调度器

    context.Services.AddQuartz(q =>

    {

        // 调度器ID和Name

        q.SchedulerId = "InventoryAlertScheduler";

        q.SchedulerName = "InventoryAlertScheduler";

        // 关联到一个触发器和一个任务

        q.ScheduleJob<InventoryAlertJob>(

            // 关联触发器

            trigger => trigger

                    // 触发器的标识为 InventoryAlertTrigger

                    .WithIdentity("InventoryAlertTrigger")

                    // 触发器关联的任务标识为 InventoryAlertJob

                    .ForJob("InventoryAlertJob")

                    // 启动后立即执行一次任务

                    .StartNow()

                    // 每隔

                    .WithSimpleSchedule(x =>

                        // 每隔20秒触发一次

                        x.WithIntervalInSeconds(20)

                        // 指定触发器将无限重复下去

                        .RepeatForever()

                    ),

            // 关联任务

            job => job.WithIdentity("InventoryAlertJob")

        );

    });

}

5、触发器详细配置

https://www.quartz-scheduler.net/documentation/quartz-3.x/tutorial/crontriggers.html#example-cron-expressions