如果你使用App模板生成项目，则默认继承了邮件功能

**简单的邮件发送**

如下我们示例一个简单的邮件发送

public class MyService : ITransientDependency

{

private readonly IEmailSender \_emailSender;

public MyService(IEmailSender emailSender)

{

\_emailSender = emailSender;

}

public async Task DoItAsync()

{

await \_emailSender.SendAsync(

"target@domain.com", // target email address

"Email subject", // subject

"This is email body..." // email body

);

}

}

IEmailSender有一个QueueAsync方法，可以用来将邮件添加到后台作业队列中，在后台线程中发送

**配置**

你可以再appsettings.json中配置邮件

"Settings": {

"Abp.Mailing.Smtp.Host": "127.0.0.1",

"Abp.Mailing.Smtp.Port": "25",

"Abp.Mailing.Smtp.UserName": "",

"Abp.Mailing.Smtp.Password": "",

"Abp.Mailing.Smtp.Domain": "",

"Abp.Mailing.Smtp.EnableSsl": "false",

"Abp.Mailing.Smtp.UseDefaultCredentials": "true",

"Abp.Mailing.DefaultFromAddress": "noreply@abp.io",

"Abp.Mailing.DefaultFromDisplayName": "ABP application"

}

其中Abp.Mailing.Smtp.Password 必须是一个加密值

你可以运行程序的ISettingEncryptionService.Encrypt方法查看已密码的加密值

**邮箱消息模板**

Abp提供了一下消息模板，如下一个示例

public class MyService : ITransientDependency

{

    private readonly IEmailSender \_emailSender;

    private readonly ITemplateRenderer \_templateRenderer;

    public MyService(

        IEmailSender emailSender,

        ITemplateRenderer templateRenderer)

    {

        \_emailSender = emailSender;

        \_templateRenderer = templateRenderer;

    }

    public async Task DoItAsync()

    {

        // 生成一个消息默默

        var body = await \_templateRenderer.RenderAsync(

            StandardEmailTemplates.Message,

            new

            {

                message = "This is email body..."

            }

        );

        // 发送邮件

        await \_emailSender.SendAsync(

            "target-address@domain.com",

            "Email subject",

            body

        );

    }

}

其中，生成的消息如下

<!DOCTYPE html>

<html lang="en" xmlns="http://www.w3.org/1999/xhtml">

<head>

    <meta charset="utf-8" />

</head>

<body>

    This is email body...

</body>

</html>

**NullEmailSender**

再这是时，我们并不想真正的发送邮箱，你可以使用如下方法，替换掉IEmailSender的实例

#if DEBUG

  context.Services.Replace(ServiceDescriptor.Singleton<IEmailSender, NullEmailSender>());

#endif

详细文档：https://docs.abp.io/zh-Hans/abp/latest/Emailing