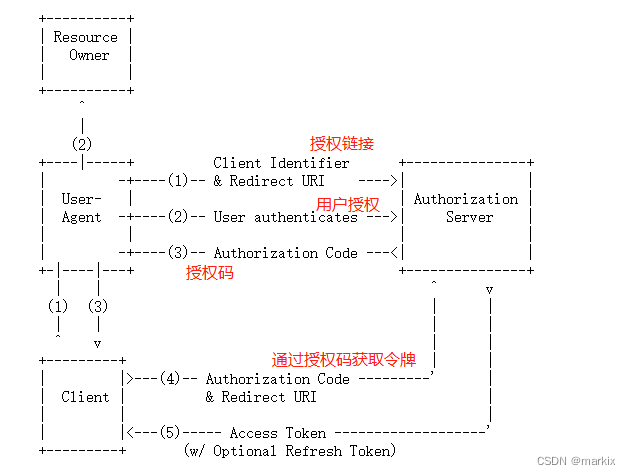
12 令牌颁发方式 之 授权码模式

Authorization Code（授权码模式）

授权码模式的流程：

\1. 客户端发起授权请求 -> 2. 用户授权 -> 3. 客户端拿到授权码code -> 4. 客户端通过code获取token



示例

环境准备

授权服务器

基于 快速搭建一个授权服务器 文章中的示例，修改 SecurityConfiguration 中 registeredClientRepository() 方法，如下：

@Bean

public RegisteredClientRepository registeredClientRepository() {

RegisteredClient registeredClient2 = RegisteredClient.withId(UUID.randomUUID().toString())

.clientId("client2")

.clientSecret("{noop}secret2")

.clientAuthenticationMethod(ClientAuthenticationMethod.CLIENT\_SECRET\_POST)

.authorizationGrantType(AuthorizationGrantType.AUTHORIZATION\_CODE)

.redirectUri("https://cn.bing.com")

.scope("test")

.clientSettings(ClientSettings.builder().requireAuthorizationConsent(true).build())

.build();

return new InMemoryRegisteredClientRepository(registeredClient2);

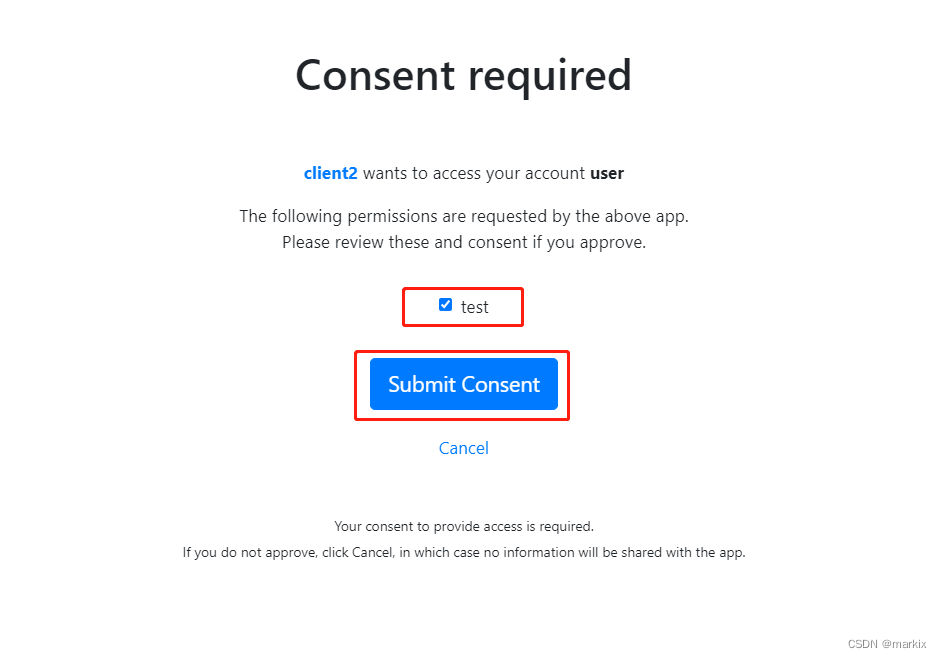
}

测试

客户端发起授权请求（浏览器访问）

http://localhost:9000/oauth2/authorize?response\_type=code&client\_id=client2&redirect\_uri=https://cn.bing.com&scope=test

用户授权：勾选 test，点击 Submit Consent



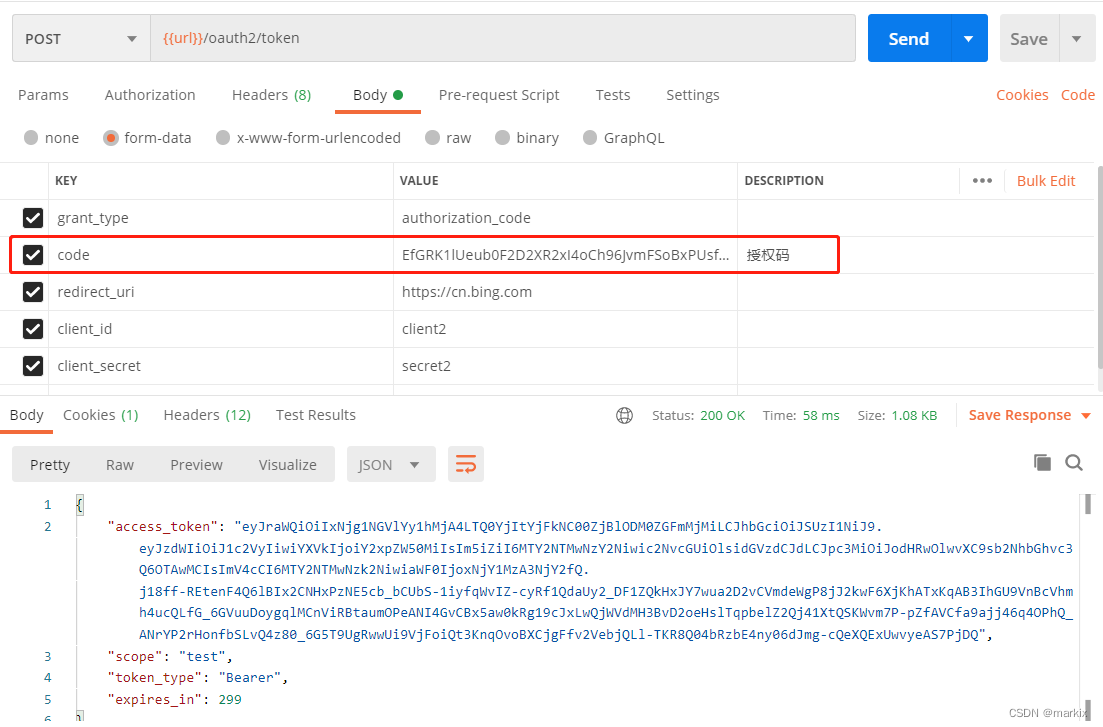
客户端拿到授权码code

用户授权后，会重定向到我们传的redirect\_uri，在url上可以得到授权码code。

https://cn.bing.com?code=EfGRK1lUeub0F2D2XR2xI4oCh96JvmFSoBxPUsfUtbdpA3AlZJw\_tYDlwfuMFrySpy9uO9EFoQH5Pf3bETb7ebA1CquQzcdpBwwqdsHSIuQD5I\_Wrww5oWKsebJPd49O

客户端通过code获取token

便于测试，直接用postman。在Body栏，填入’grant\_type=authorization\_code’、code、client\_id、client\_secret，发送请求。



如上，成功获取到令牌（access\_token）。这就是授权码模式。

源码分析

授权请求 OAuth2AuthorizationEndpointFilter

接收客户端的授权请求，若需要用户授权则响应授权页面，若不需要用户授权或用户同意授权则重定向到客户端指定的 redirect\_uri 并携带 授权码code。

另外，授权服务器会将code和请求信息存起来。

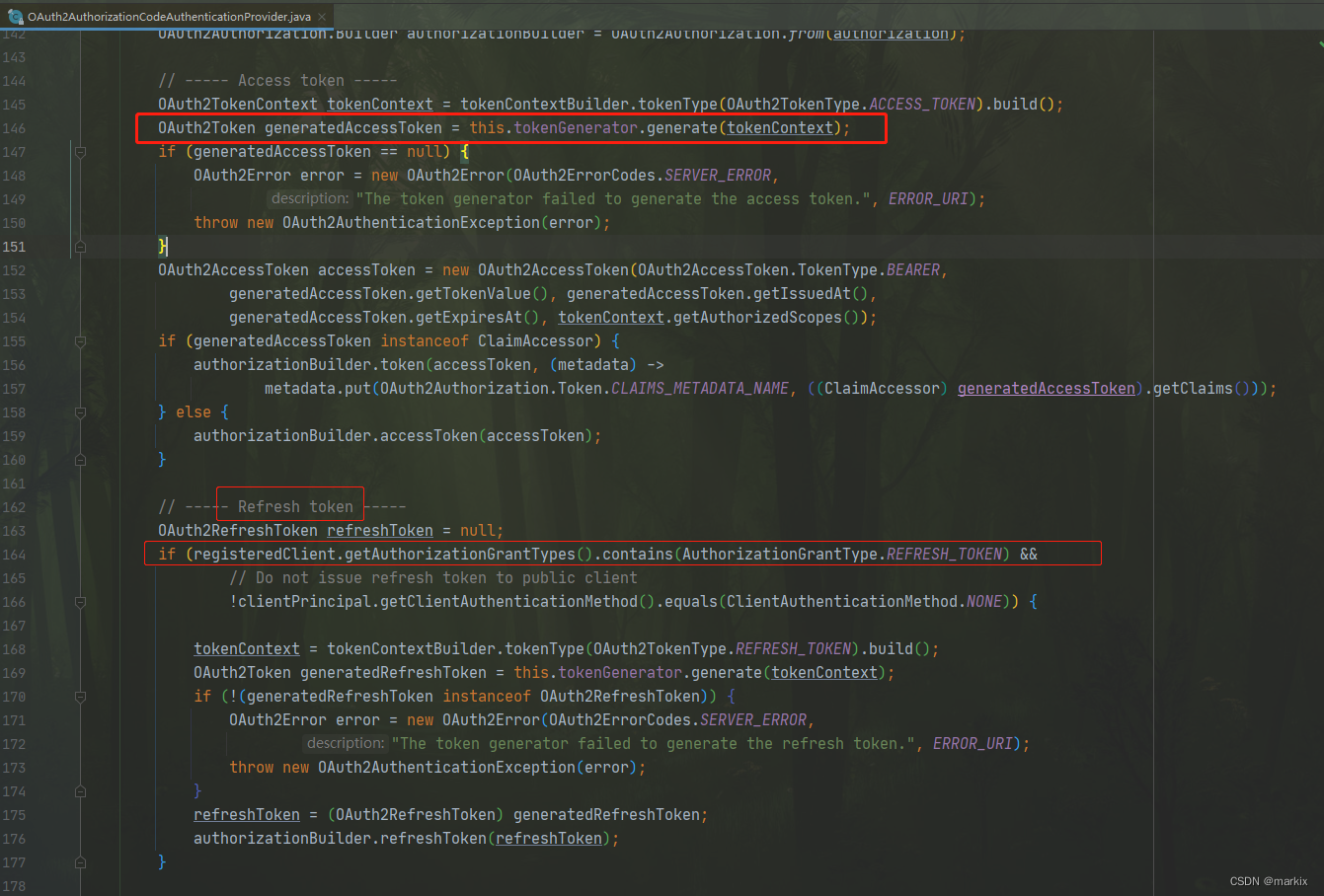
令牌颁发（授权码）

OAuth2AuthorizationCodeAuthenticationConverter

判断请求参数 grant\_type 的值是否为 authorization\_code，若是，则构建 OAuth2AuthorizationCodeAuthenticationToken。

OAuth2AuthorizationCodeAuthenticationProvider

核心还是生成访问令牌access token，还有一点需要额外提及的就是刷新令牌refresh token，在授权码模式颁发令牌时，当发现该客户端支持 REFRESH\_TOKEN 模式时，还会返回刷新令牌。这是刷新令牌模式的关键。



end