OAUTH

OAuth 2.0 is an authorization framework that enables third-party applications to access protected resources on behalf of a user without requiring the user’s credentials. This is achieved through the use of access tokens, which are issued by an OAuth provider and used by third-party applications to access the user’s resources

Different Entities

1. **Resource Owner:** The user who owns the resource that is being accessed by the client.
2. **Client**: The application that is requesting access to the resource on behalf of the user.
3. **Authorization Server**: The server that issues access tokens to the client after successful authentication of the user.
4. **Resource Server**: The server that holds the resource that is being accessed by the client.

# OAuth Flow

1. User requests access to a protected resource from a third-party application.
2. The third-party application redirects the user to an OAuth provider to obtain an access token.
3. The user logs in to the OAuth provider and grants permission to the third-party application to access the protected resource.
4. The OAuth provider issues an access token to the third-party application.
5. The third-party application uses the access token to access the protected resource on behalf of the user.

**Scope:** It tells what kind of data client want to access

**Github Login URL:** /oauth2/authorization/github

**Callback URL**: <http://localhost:8080/login/oauth2/code/github>

**Properties:**

spring.security.oauth2.client.registration.github.client-id

spring.security.oauth2.client.registration.github.client-secret

Basic Code Snippet:  


**Get Principal** : @AuthenticationPrincipal OAuth2User oauth2User

**Get Token**: @RegisteredOAuth2AuthorizedClient OAuth2AuthorizedClient