


Contacts Test All Accounts | Quickstart

✕

 Got a second? We would love to hear your feedback on quickstarts. →

```
.....
        .WithRedirectUri("https://login.microsoftonline.com/common/oauth2/nativeclient")
        .WithAuthority(AzureCloudInstance.AzurePublic, Tenant)
        .Build();
```

Where:

`ClientId` Is the **Application (client) ID** for the application registered in the Azure portal. You can find this value in the app's **Overview** page in the Azure portal.

Requesting tokens

MSAL has two methods for acquiring tokens: `AcquireTokenInteractive` and `AcquireTokenSilent` .

Get a user token interactively

Some situations require forcing users interact with the Microsoft identity platform endpoint through a popup window to either validate their credentials or to give consent. Some examples include:

- The first time users sign in to the application
- When users may need to reenter their credentials because the password has expired
- When your application is requesting access to a resource that the user needs to consent to
- When two factor authentication is required

```
authResult = await App.PublicClientApp.AcquireTokenInteractive(_scopes)
                                     .ExecuteAsync();
```

Where:

`_scopes` Contains the scopes being requested, such as `{ "user.read" }` for Microsoft Graph or `{ "api:///access_as_user" }` for custom Web APIs.

Get a user token silently

You don't want to require the user to validate their credentials every time they need to access a resource. Most of the time you want token acquisitions and renewal without any user interaction. You can use the `AcquireTokenSilent` method to obtain tokens to access protected resources after the initial `AcquireTokenInteractive` method:

```
var accounts = await App.PublicClientApp.GetAccountsAsync();
var firstAccount = accounts.FirstOrDefault();
authResult = await App.PublicClientApp.AcquireTokenSilent(scopes, firstAccount)
                                     .ExecuteAsync();
```