# Development Process - React

Step-By-Step

1. Create a free developer account at okta.com
2. Add OpenID Connect client app in Okta
3. Set up app configuration for OpenID Connect
4. Install Okta SDK dependencies
5. Integrate Okta Sign-In Widget
6. Add Configurations to our application
7. Create new Security Routes & Navigation Login/Logout

# Create free developer account at okta.com

- Visit `http://developer.okta.com`
- Sign up for free account
- Check your email to verify your account

**Free Developer Account**
- No credit card required
- Supports 1,000 users

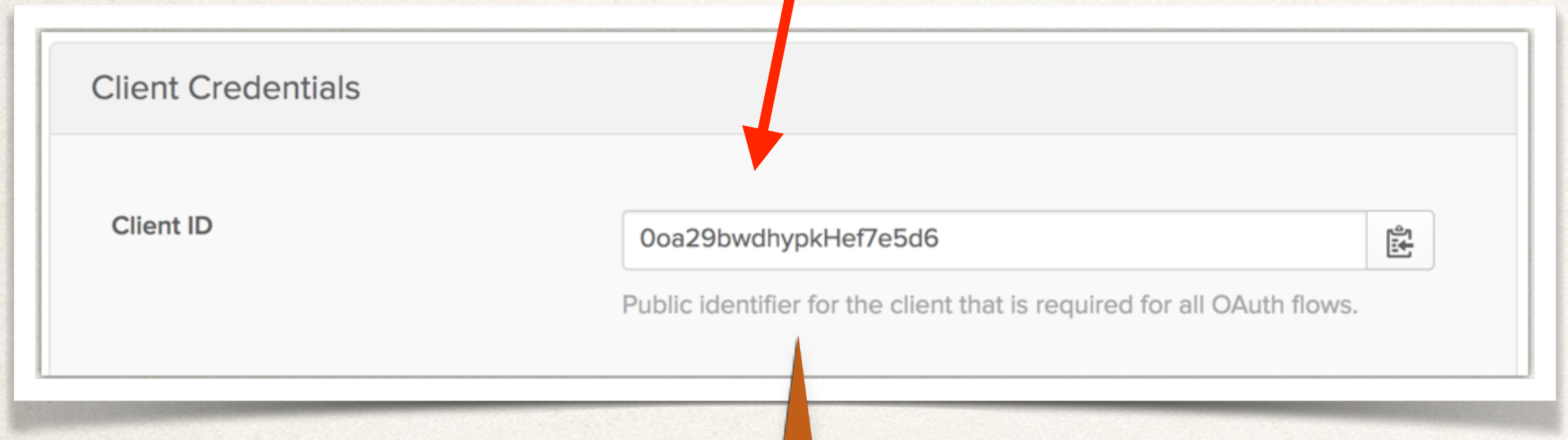
# Add OpenID Connect client app in Okta

- In your Okta Developer Account
- Create a new application
- Select option for:
  - **OIDC: Open ID Connect > Single-Page Application**
  - *Other details covered in later video*

# Set up app configuration for OpenID Connect

- To connect with Okta, need to provide configs

- Need the **`clientId`** and **`issuer`** ... available on Okta application details screen



Client Credentials

Client ID

0oa29bwdhypkHef7e5d6

Public identifier for the client that is required for all OAuth flows.

**public identifier of the client app**

dev-1234567.okta.com

**Issuer of tokens**

**`https://${yourOktaDomain}/oauth2/default`**

# Step 3: Set up app configuration for OpenID Connect

File: my-app-config.ts

```
export default {

    oidc: {
        clientId: '0oafUnNvI7c0ding',
        issuer: 'https://dev-1234567.okta.com/oauth2/default',
        redirectUri: 'http://localhost:4200/login/callback',
        scopes: ['openid', 'profile', 'email'],
      }

}
```

public identifier of the client app

URL when authorizing
with Okta Authorization Server
`https://${yourOktaDomain}/oauth2/`

Once user logs in ...
send them here

# Step 3: Set up app configuration for OpenID Connect

**File: my-app-config.ts**

```
export default {

    oidc: {
        clientId: '0oafUnNvI7c0ding',
        issuer: 'https://dev-1234567.okta.com/oauth2/default',
        redirectUri: 'http://localhost:4200/login/callback',
        scopes: ['openid', 'profile', 'email'],
      }

}
```

`openid`: required for authentication requests
`profile`: user's first name, last name, phone, etc
`email`: user's email address

OpenID Connect (OIDC) *scopes* provide access to information about a user such as name, phone number, email etc

The *scope* has a set of user attributes known as "claims".

# Install Okta SDK dependencies

- We will use Okta SDK dependencies

Okta Sign-In Widget

Okta React SDK

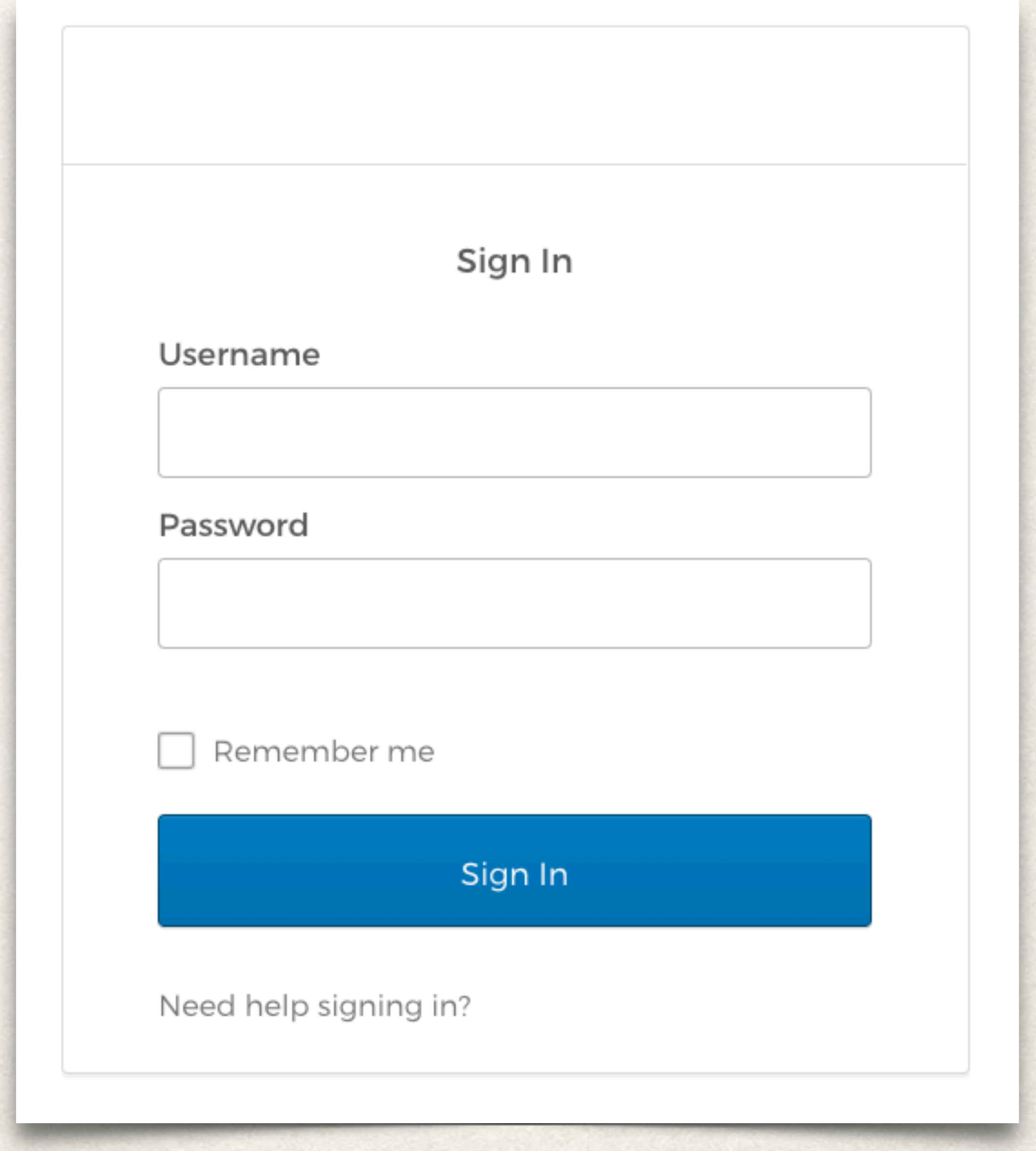
Okta JavaScript SDK

# Install Okta SDK dependencies

- **Okta Sign-In Widget** is a JavaScript library for application login

- You don't have to create the HTML ... just integrate the widget into your application

- Customizable ... use your own logo, field names and custom fields



Sign In

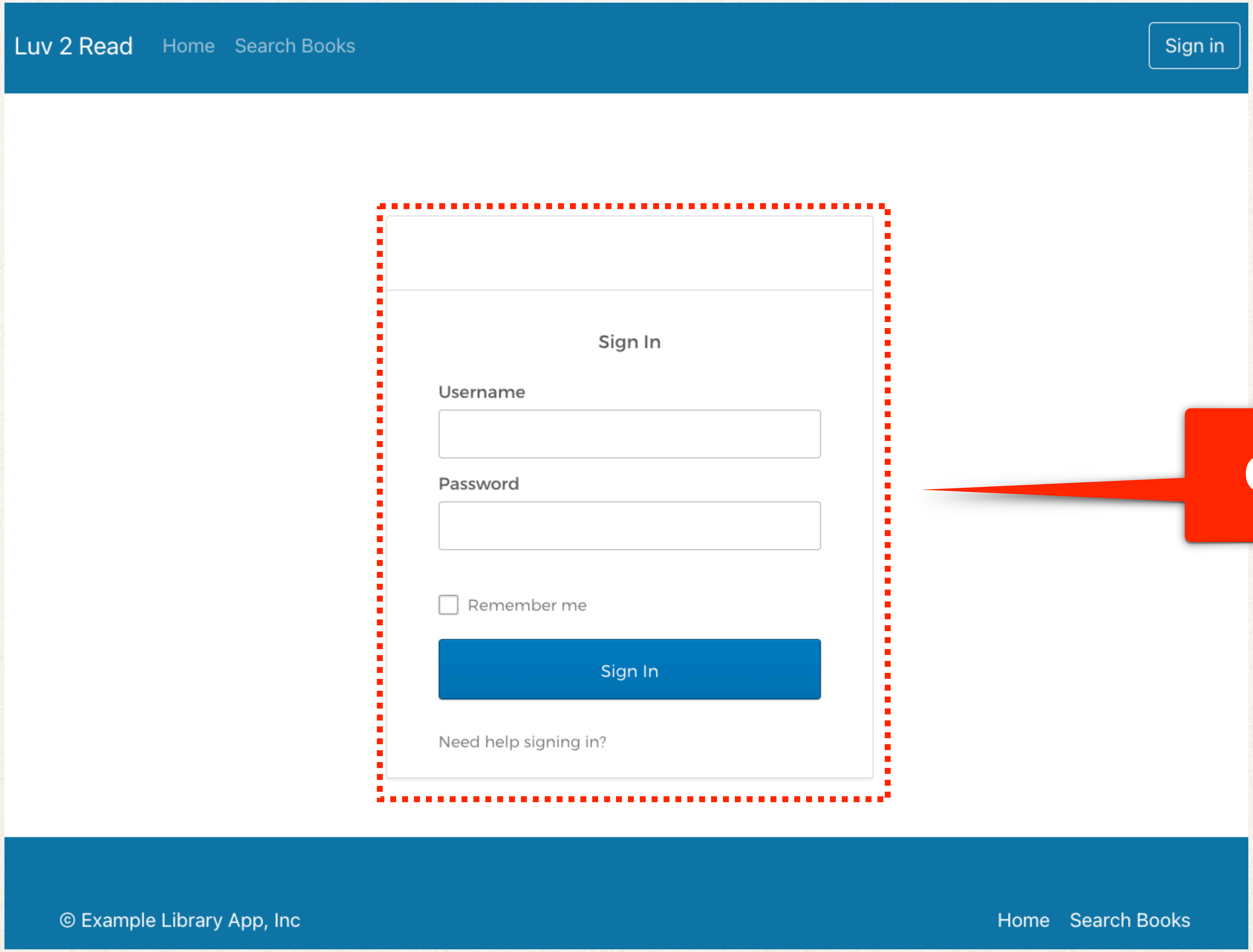
Username

Password

□ Remember me

Need help signing in?

# Install Okta SDK dependencies



Luv 2 Read Home Search Books Sign in

Sign In

Username

Password

□ Remember me

Sign In

Need help signing in?

© Example Library App, Inc Home Search Books

Okta Sign-In Widget

# Install Okta SDK dependencies

- **Okta React SDK** provides integration with React Router for authentication and authorization

- Core Features
  - Login/logout from Okta using OAuth 2.0 API
  - Retrieve user information and determine authentication status

- Additional Features
  - Add protected routes that require authentication
  - Subscribe to changes in authentication state

# Install Okta SDK dependencies

Okta Sign-In Widget

```
$ npm install @okta/okta-signin-widget


$ npm install @okta/okta-react
```

Okta React SDK