# Connecting to Your Seller Organization

Linking your front-end application to an OrderCloud seller or buyer organization can be accomplished using the `scope` and `clientid` constants described below.

## Application Constants

`src/app/app.constants.json` is a small, yet powerful file host a JSON object of key value pairs to be used throughout the application. From this object, the build process will generate

a file containing AngularJS constants:

[here](https://github.com/ordercloud-api/angular-seller/blob/development/gulp.config.js#L86), which can then be injected into your various AngularJS controllers, factories, providers, etc. Think of it as a base configuration or settings file for your application.

Constant | Type | Description

--- | --- | ---

`appname` | string | A short name for your application. This will be used in the `<title>` as well as displayed in the top left navigation of the application.

`scope` | string | A space delimited string of [OrderCloud roles](https://documentation.ordercloud.io/guides/authentication/security-profiles#Roles) that will be requested during authentication.

`clientid` | string | An [OrderCloud ClientID](https://documentation.ordercloud.io/guides/getting-started/using-the-dashboard#Applications) for the seller, buyer network, or buyer application that will be used for authentication.

`environment` | string | A string declaring the OrderCloud environment the application will point to. Currently, only `production` is available; however, when a large release consisting of breaking changes is scheduled, a `staging` environment will be provided.

`defaultstate` | string | The default ui-router state within the application that users will be directed to should they attempt to access a state that does not exist.

`html5mode` | bool | True/false whether you want HTML5 Mode enable within the application.

`bootswatchtheme` | string | The Bootswatch theme that is automatically applied to the application during the build process. A list of available themes can be found [here](https://bootswatch.com/). A value of `null` will apply the default angular-seller theme.

## Process Environment Variable Overrides

The constants provided above can be overwritten within your hosting providers application settings. For example, within Heroku, you can override these constants using their [Config Variables](https://devcenter.heroku.com/articles/config-vars#setting-up-config-vars-for-a-deployed-application). This is accomplished in the [`gulp.config.js`](https://github.com/ordercloud-api/angular-seller/blob/development/gulp.config.js#L116-L121) file, which can be customized to include additional application constants.