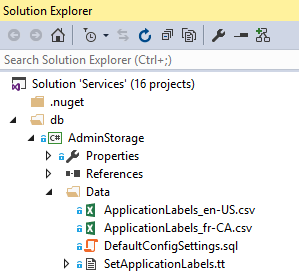
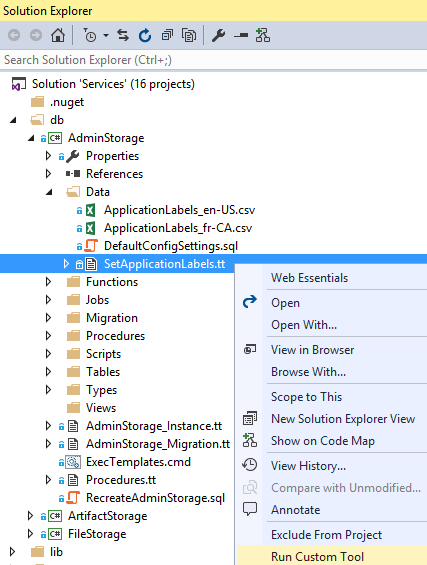
Nova Web Application – Localization Primer

# 1. How to add a new localization string

1. Open the Services solution in Visual Studio
   1. This solution is found within the blueprint GitHub repository at **blueprint/svc/Services.sln**
2. Within **<solution root>/db/AdminStorage/Data**, open the **.csv** file that corresponds to the locale that you are adding a new localization string to.

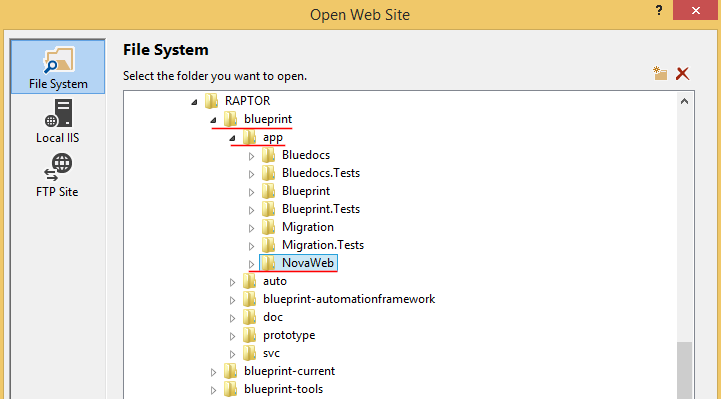


1. Update the **.csv** file with key, text combination that represents the new localization string.
2. Please ensure that the key is unique, otherwise it may not be added to the **ApplicationLabels** table within the **AdminStorage** database.
3. Execute the ‘**SetApplicationLabels.tt**’ script to re-generate the ‘**SetApplicationLabels.sql**’ script that is responsible for updating the **ApplicationLabels** table within the **AdminStorage** database.



1. Repeat step 5 for both the ‘**AdminStorage\_Instance.tt**’ and ‘**AdminStorage\_Migration.tt**’files.

# 2. How to reference localized string within Nova Web Application

1. Within Visual Studio, open the NovaWeb website
   1. This website is found within the blueprint GitHub repository at **blueprint/app/NovaWeb**.  
      

## 2.1 Reference within Html file

To reference the localization label within an Html file, simply include the following:

**{{::$root.config.labels["<localization\_key>"]}}**

Where **<localization\_key>** represents the key value that was specified *above* in step 3 of *How to add a new localization string.*

## Reference within Typescript file

To reference the localization label within a Typescript file:

1. Import the localization service, if it already has not been added.

**import {ILocalizationService} from "../../core/localization";**

1. Update the class’ **$inject** member to include the local name for the localization service.

**static $inject: [string] = [...,"localization"]**

1. Add the localization service to the class’ constructor, so that it can be accessed within the class. Be sure to use the same name specified in step 2 above.

**constructor(..., private localization: ILocalizationService) {**

**...**

**}**

1. Access the localization label within the class by through the following expression:

**localization.get(“<localization\_key>”)**

Where **<localization\_key>** represents the key value that was specified *above* in step 3 of *How to add a new localization string.*