Translations:

Regular Development (please follow these rules):

* Add a new entry in one of the English messages properties file:
  + Do nothing to other language versions of the properties files.
* Change the text of an existing entry in an English messages properties file:
  + Either remove the corresponding entry in the language file or add the language code in parentheses to the text, e.g. (ro)
* Remove an entry in the English messages properties file:
  + Remove the corresponding entry in the language file.

Work Flow for working with company that will translate for us:

1. Generate the list of properties that need to be translated by following these steps:
   1. Run the utility (java app) GenerateListToTranslate

Command Line:

com.inthinc.pro.i18nutil.GenerateListToTranslate <output file> <root dir> <lang code>

Example:

com.inthinc.pro.i18nutil.GenerateListToTranslate c:\toTranslate.properties c:\develop\tiwipro\root ro

This utility will

* iterate through the entire directory tree and get a list of property files that have a corresponding language file (i.e. messages.properties and messages\_ro.properties).
* add a file marker entry in the output file (e.g. \*\*\* file path \*\*\*)
* compare the entries in the two files and if the English entry is missing from the lang file or has the language marker (e.g. (ro)) or is identical to the English entry it will add it to the output file. There is a list of entry keys to ignore for each file. These are in com.inthinc.pro.i18nutil.NonTranslated.java.
  1. Send the generated output file to the translation company. Instruct them to only change the text to the right of the equal signs. (do not change file markers, etc.)
  2. When the translated file is returned, run the utility to merge the translations into our code base.

Command Line:

com.inthinc.pro.i18nutil.MergeTranslated <input file> <root dir> <lang code>

Example:

com.inthinc.pro.i18nutil.MergeTranslated c:\translated.properties c:\develop\tiwipro\root ro

This utility will

* iterate through the entire directory tree and get a list of property files that have a corresponding language file (i.e. messages.properties and messages\_ro.properties).
* match up the file markers in the input file (e.g. \*\*\* file path \*\*\*) so it can find the correct language file.
* update the entries in the language file with the new translations and save the file. (note: the file is written in no particular order, we might want to change this at some point)
  1. Run the unit test to make sure all of the entries are translated (you may need to remove the @Ignore annotation on the test).

com.inthinc.pro.i18nutil.TestTranslations

If the test reports errors on strings that do not need to be translated, add these to com.inthinc.pro.i18nutil.NonTranslated.java.