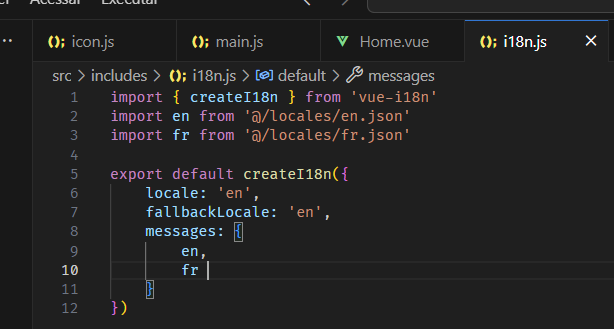
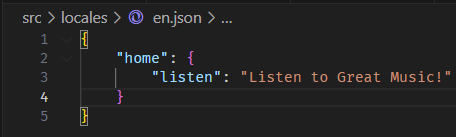
**INTRODUÇÃO A i18n**

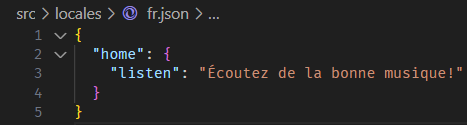
i18n é uma pequena internacionalização. É o processo de tradução para sua aplicação.

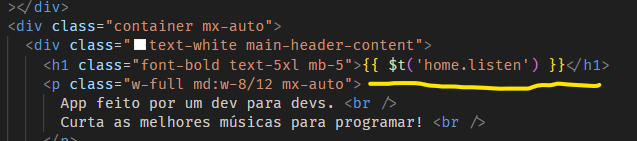
npm install vue-i18n@9

Próximo passo é criar o arquivo i18n.js dentro da pasta includes:



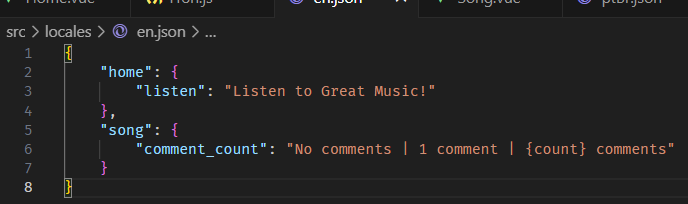


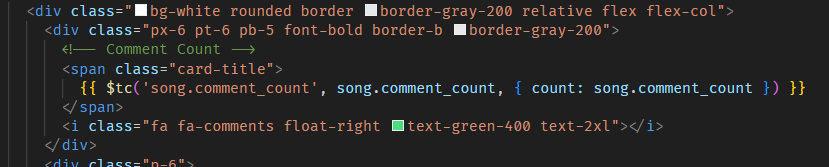




**FORMATANDO E PLURALIZANDO**

O i18n pode lidar sozinho com palavras no plural e singular, estou com o exemplo de uma seção de comentários, e para isso é necessário a seguinte adaptação:

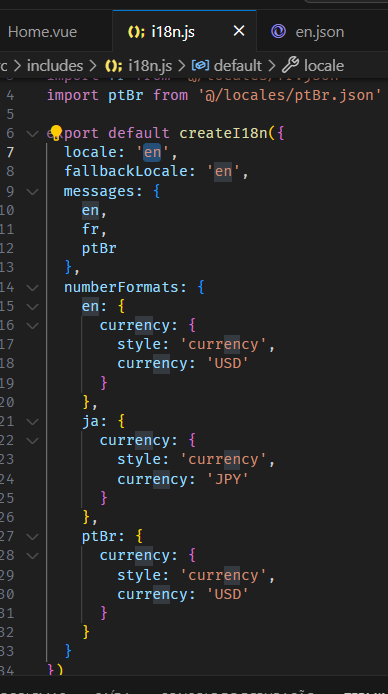


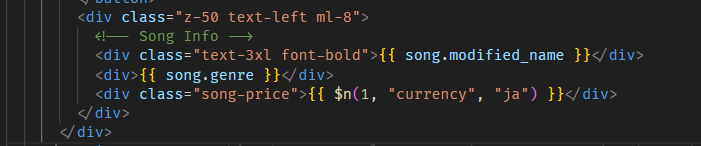


Em geral, a chamada com $t já é suficiente para traduções. Nesse caso, no entanto, precisamos de $tc porque essa nomenclatura sinaliza que podemos ter plurais. São 3 parâmetros: o primeiro é o nome da chave dentro do arquivo de tradução em json, o segundo é o que vamos traduzir, e o terceiro é a quantidade que existe ali.

**LOCALIZAÇÃO DE NÚMEROS**

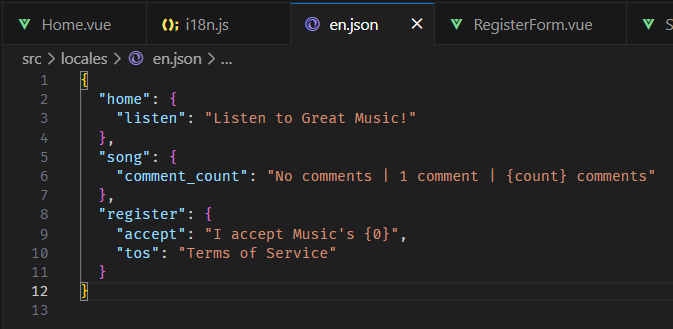
Para formatação de números do tipo moeda, basta fazer o seguinte:

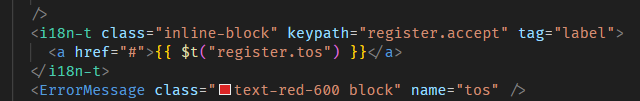




Em que, o primeiro parâmetro é o valor a ser exibido, o segundo é a chave definida no i18n e o terceiro é a chave externa, como se fosse isso.

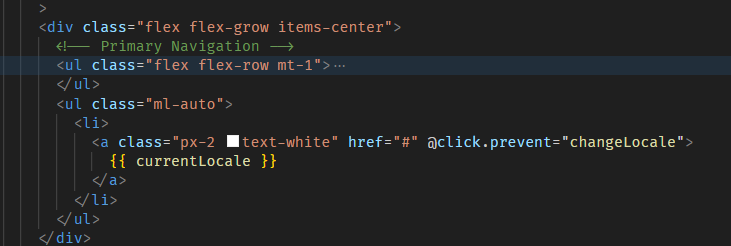
**TRADUZINDO HTML COM INTERPOLAÇÃO DE COMPONENTE**

****

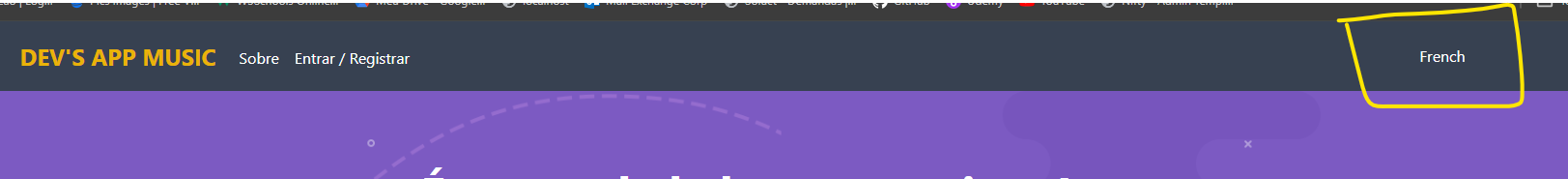
****

**ALTERANDO LOCALES**

Com esse mecanismo é possível, por exemplo, trocar rapidamente a linguagem do sistema:







De modo geral, o que está acontecendo nesses prints é o seguinte: criei uma nova li colocando ali dentro uma tag a. O valor dela é o currentLocale. Esse cara retorna o this.$i18n.locale, e faz aquele ternário ali. Essa tag contém um método que é o changeLocale(), esse vai ser o responsável por intercalar o valor do locale.