

Internationalization and localization

- 1 i18n, l10n
- 2 Locale
- 3 Why
- 4 How
- 5 Example

References

Few and old books on this topic



Java internationalization. A. Deitsch, D. Czarnecki. O'Reilly, 2001.



Developing international software. Dr. International group. Microsoft Press, 2003.

However there is documentation for different programming languages

- Java internationalization [guide](#) and [trail](#)
- [Internationalizing Flutter apps](#)

Slides [Designing for International Users: Practical Tips](#), WWW Consortium. Examples of internationalization problems.

Internationalization and localization

Internationalization, *i18n*

The design and development of a product, application or document content that **enables easy localization** for target audiences that vary in culture, region, or language [W3C]

Process of designing and implementing a software application so that it can be very **easily adapted** to various **languages and regions** without changes to the programming logic.

Internationalization and localization

Localization, *l10n*

The **adaptation** to meet the language, cultural and other requirements of a specific target market. [W3C]

Once a piece of software has been internationalized, is the process of adapting it for a specific language and region, that is, to a **locale**.

Internationalization and localization

What's the difference ?

- internationalization is the adaptation of products for potential use virtually everywhere.
- localization is the addition of special features for use in a specific locale (eg. translation of all texts to spanish of Mexico)
- internationalization of a product is done just once
- a localization is done once for every locale

Locale

Locale

Collection of **features of the user's environment** that are dependent on language, country and region, and cultural conventions.

In the context of programming identified by a pair
(*language, region*) like in Java `es_MX` = spanish of Mexico, `de_AT`
german of Austria.

Locale

What kind of features are susceptible of being localized ?

Obviously language, plus language related :

- regional version
- script (not only alphabet)
- writing direction : left-to-right, right-to-left
- lexicographical order
- construction of plurals
- ...

Locale

Cultural conventions

- format of date and time, see Wikipedia [date format by country](#)
- calendar
- numbers : [digits](#) and formats
- [postal addresses](#)
- [person names](#) and titles
- value of money and currency
- physical measurement units (weight, length, size . . .)
- [meaning of colors](#), [gestures](#), signs

Locale

Style	Locale	Format
SHORT	England	16/10/99 13:03
	France	16/10/99 13:03
	US	10/16/99 1:03 PM
	Germany	16.10.99 13:03
	Italy	16/10/99 13.03
	Israel	13:03 16/10/99
MEDIUM	England	16-Oct-99 13:03:01
	France	16 oct. 99 13:03:01
	US	Oct 16, 1999 1:03:01 PM
	Germany	16.10.1999 13:03:01
	Italy	16-ott-99 13.03.01
	Israel	13:03:01 16/10/1999
LONG	England	16 October 1999 13:03:01 GMT-04:00
	France	16 octobre 1999 13:03:01 GMT-04:00
	US	October 16, 1999 1:03:01 PM EDT
	Germany	16. Oktober 1999 13:03:01 GMT-04:00
	Italy	16 ottobre 1999 13.03.01 GMT-04:00
	Israel	13:03:01 GMT-04:00 16 1999 אוקטובר

Locale

European	Arabic	Indic- Bengali	Thai	Ideographic
0	٠	০	๐	零
1	١	১	๑	一
2	٢	২	๒	二
3	٣	৩	๓	三
4	٤	৪	๔	四
5	٥	৫	๕	五
6	٦	৬	๖	六
7	٧	৭	๗	七
8	٨	৮	๘	八
9	٩	৯	๙	九

Locale

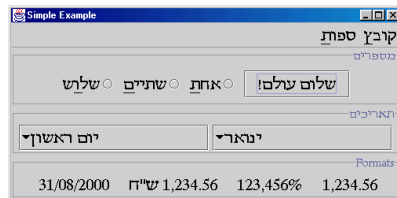
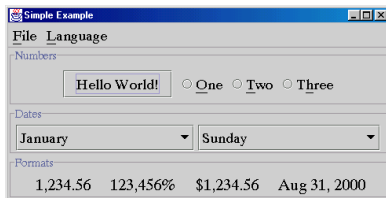
Color	Cultural Meaning
Red	China: celebration, happiness, and luck India: purity United States: stop or danger
White	Eastern cultures: mourning or death United States: purity
Blue	China: immortality Hindus: the color of Krishna Jews: holiness Middle East: protection; ward off evil spirits
Green	India: the color of Islam Ireland: religious significance Some tropical countries: danger United States: safe or go; environmental awareness
Yellow	Asia: sacred, imperial Western cultures: joy; happiness; caution
Black	Western cultures: mourning or death

Locale

Even legal issues

- country borders
- names of seas and places

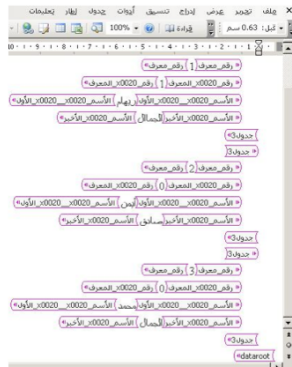
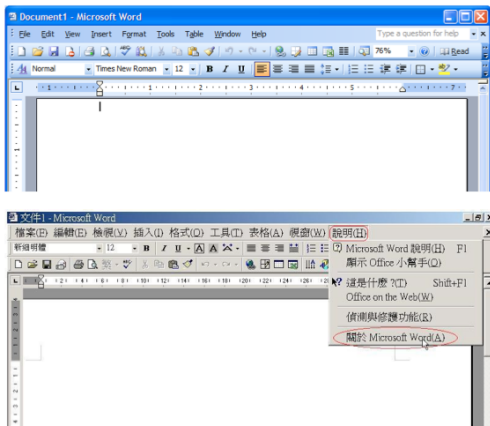
Examples



Java internationalization. A. Deitsch, D. Czarnecki. O'Reilly, 2001.

Examples

MS-Word



Examples

Facetune 2 : localization goes beyond translation



En-US locale

Examples

Facetune 2 : localization goes beyond translation



Japanese locale

Examples

Facetune 2 : localization goes beyond translation



Chinese locale

Why

People from other countries are used to the [American | Western] way and they like to be [American | Western]ized.

This may seem offensive at some places. Cultural differences **do exist** and must be taken into account.

The world speaks English.

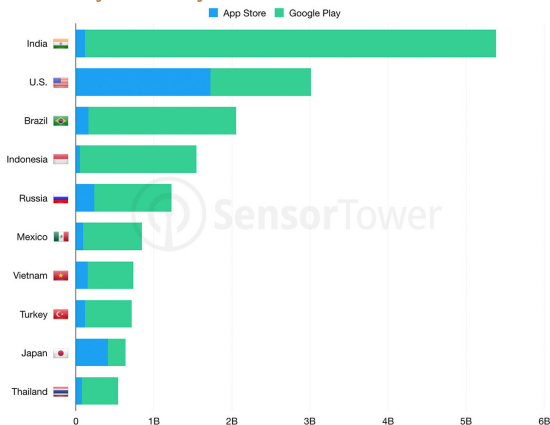
Only 8% to 10% of the world's population speaks English as their primary language.

My users can read English, so why should I localize my software ?

To avoid confusion and increase usability. Today, more and more web sites are providing content in multiple languages, and more and more software products are being designed with cultural differences in mind.

Why

Target users are my country's nationals



First-time downloads in second quarter 2020, China excluded (no Google Play there). 1B = 10^9 . Source [SensorTower](#)

Why

The [next billion internet users](#) will come from emerging economies: India, Brazil, Indonesia, Nigeria... Here's what [Google already knows](#) (2018) about them:

- They have a mobile-only mindset. Most have never used a PC and never will.
- Even the small percentage that knows English prefers content in their own language.

Users expect that the software works correctly in their language using their local conventions. “Internationalization is not a feature”: it is an essential part of the software design.

Why i18n is difficult ?

Can not rely on automatic translation. Meaning depends on context.

Can't necessarily rely on bilingual members of your design team, either. They may be reasonably fluent in the other language, but not sufficiently immersed in the culture or national standards.

Actually, there is an [industry](#) of software localization.

“You are not the user” is especially true in internationalization.

Why i18n is difficult ?

I18n is not only about translating (possibly large amounts of) text within a context. Also

- text format, size and reading direction
- images and icons
- the UI layout itself (eg. long German names)
- ...

Many potential locales: just main Western languages plus CJK (Chinese, Japanese, Korean) may account for 10 versions of the UI.

i18n and l10n in Flutter

Localization is the process of making your app available to other locales than the default (`en_US`).

Internationalization is how you architect and engineer your code to make it easy to localize without dramatic engineering efforts.

We don't want

- create a copy of your code for each language, or
- add if-then-else statements all over the place choosing which text/icon/image/layout... to display

That's where internationalization support of a specific combination of language + libraries + IDE comes to the rescue.

However, there is not a 100% clean, fast solution normally.

i18n and l10n in Flutter

Flutter doesn't have an absolutely set way of how localization is done.

It gives you a lot of freedom which can be good if you know what you're doing, but also confusing if you're just starting out.

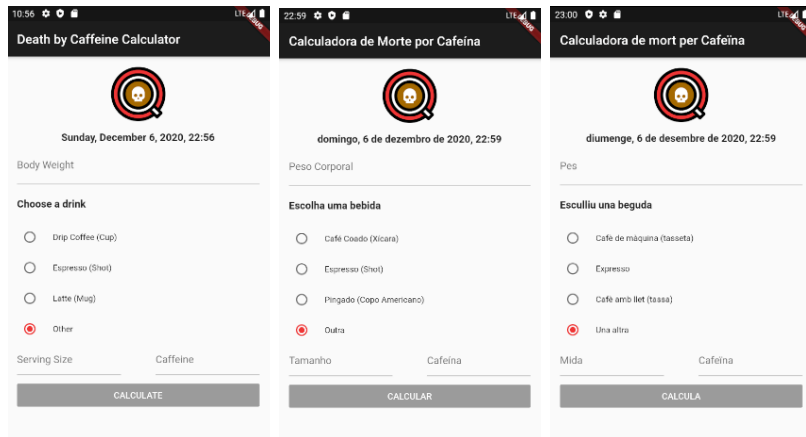
The first step in i18n is to separate the text in the interface from the source code:

- move all texts to a file and assign to each text a key
- source code refers to text by keys
- one file per locale, with the same keys
- take care of plurals

In official documentation they use `.arb` files to store your strings in multiple languages.

Example

I've followed [this tutorial](#) that sticks to .arb files but relies on Localizely's Flutter Intl plugin to generate boilerplate code.



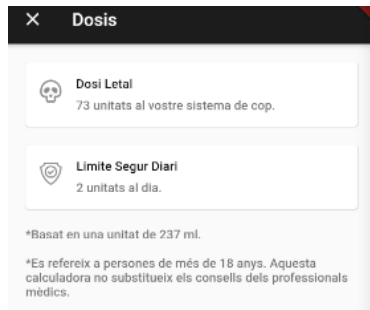
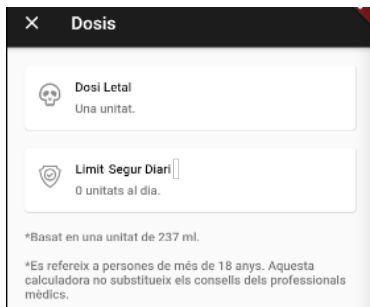
Example: texts

File intl_ca_ES.arb

```
{
  "@@locale": "ca_ES",
  "formPageAppBarTitle": "Calculadora de mort per Cafeïna",
  "formPageWeightInputLabel": "Pes",
  "formPageWeightInputSuffix": "kilos",
  "formPageRadioListLabel": "Esculliu una beguda",
  "firstSuggestedDrinkName": "Cafè de màquina (tasseta)",
  ...
  "resultsPageLethalDosageTitle": "Dosi Letal",
  "resultsPageLethalDosageMessage": "{quantity, plural,
    one{Una unitat.}
    other{{formattedNumber} unitats.}}",
  ...
}
```

Plurals are specially difficult because there [many possible cases](#). German, Spanish, English have 2 forms (1 and > 1). Russian, Polish, Czech 3. Arabic 6. This is known as *p11n* = pluralization.

Example: texts




Example: date and time

```
class _FormPageState extends State<FormPage> {  
  ...  
  Locale _userLocale;  
  DateFormat _dateFormatter = DateFormat.yMMMMEEEEd();  
  DateFormat _timeFormatter = DateFormat.Hm(); // default locale  
  
  @override  
  void didChangeDependencies() {  
    final newLocale = Localizations.localeOf(context);  
    if (newLocale != _userLocale) {  
      _userLocale = newLocale;  
      _dateFormatter = DateFormat.yMMMMEEEEd(_userLocale.toString());  
      _timeFormatter = DateFormat.Hm(_userLocale.toString());  
    }  
    super.didChangeDependencies();  
  }  
  ...  
  Text(_dateFormatter.format(DateTime.now()) + ', '  
    + _timeFormatter.format(DateTime.now())),
```

Example: physical units

Death by Caffeine Calculator



Monday, December 7, 2020, 12:10

Body Weight pounds


Choose a drink

☐ Drip Coffee (Cup)

☐ Espresso (Shot)

pounds

Calculadora de Morte por Cafeína



segunda-feira, 7 de dezembro de 2020, 12:15

Peso Corporal quilos


Escolha uma bebida

☐ Café Coado (Xicara)

☐ Espresso (Shot)

Kg

Calculadora de mort per Cafeïna



dilluns, 7 de desembre de 2020, 12:09

Pes kilos

Esculliu una beguda

☐ Café de màquina (tasseta)

☐ Expresso

Kg

$$1 \text{ kg} = 2.20462 \text{ pounds}$$

Example: physical units

File `form_page.dart`

```
final weight = _weightTextController.intValue.toPoundsIfNotAlready(  
  _userLocale,  
); // computations assume weight is in pounds
```

File `measurement_conversion.dart`

```
bool _shouldUseImperialSystem(Locale locale) {  
  final countryCode = locale.countryCode;  
  return countryCode == 'US';  
}  
  
// add method to native class int  
extension IntMeasurementConversion on int {  
  int get _roundedPoundFromKg => (this * 2.20462).round();  
  int toPoundsIfNotAlready(Locale locale) {  
    if (_shouldUseImperialSystem(locale)) {  
      return this;  
    }  
    return _roundedPoundFromKg;  
  }  
}
```

i18n and l10n in Flutter

We have internationalized

- texts (translation)
- date and time
- physical measurement units (pounds, fl. oz.)

and then localized to three locales: `en_US`, `pt_BR`, `ca_ES`.

For texts we got support of Flutter Intl plugin, for date and time from Dart's Intl library. For weight no support.

And this is just the beginning. . .