

Lab 13

[valid 2020-2021]

Internationalization and Localization

Create an application to explore the available **locales** included in the standard Java Development Kit.

The main specifications of the application are:

Compulsory (1p)

- Create a package with the name *res*. Inside this package create the files: *Messages.properties*, *Messages_ro.properties*.

```
#Messages.properties file
prompt = Input command:
locales = The available locales
are:
locale.set = The current locale
is {0}
info = Information about {0}:
invalid = Unknown command
```

```
#Messages_ro.properties file
prompt = Comanda ta:
locales = Localizarile disponibile
sunt:
locale.set = Localizarea curenta
este {0}
info = Informatii despre localizarea
{0}:
invalid = Comanda necunoascuta
```

- Create the package *com* and implement the following classes describing commands:
 - *DisplayLocales*: to display all available locales
 - *SetLocale*: to set the application current locale.
 - *Info*: to display informations about the current or a specific locale.
When the user sets a specific language tag, various information obtained using standard Java classes such as *Currency* or *DateFormatSymbols* should be displayed in a text area, as in the following example:
 - **Country**: Romania (România)
 - **Language**: Romanian (română)
 - **Currency**: RON (Romanian Leu)
 - **Week Days**: luni, marți, miercuri, joi, vineri, sâmbătă, duminică

- **Months:** ianuarie, februarie, martie, aprilie, mai, iunie, iulie, august, septembrie, octombrie, noiembrie, decembrie
 - **Today:** May 8, 2016 (8 mai 2016)
- Create the package *app* and the main class *LocaleExplore*. Inside this class, read commands from the keyboard and execute them.
- All the **locale-sensitive** information should be **translated** in at least two languages (english-default and romanian), using the resource files.

Resources

- [Internationalization](#)

Objectives

- Understand the concepts of *internationalization* and *localization*
- Identify a localization by using the *Locale* object
- Use properties files and resource bundles
- Parse and format locale-sensitive data, such as dates, number, currencies.