

Localization

Resource Bundles

What is Resource Bundle?

- properties file which contains the locale-specific objects for the program
 - basically a map with keys and values
- the keys and values in property file are separated either by = or :
- property file name is like this: Museum_en.properties
 - <name_of_the_bundle>_<target_locale>.properties
- to load a resource bundle:

```
ResourceBundle.getBundle(bundle, locale)
```

```
// Museum_en.properties  
greeting=Hello  
open=The museum is open
```

```
Hello! The museum is open  
Ciao! Il museo è aperto
```

```
// Museum_it.properties  
greeting:Ciao  
open:Il museo è aperto
```

can be omitted (default locale will be used)

```
// in the main() method  
var enRb = ResourceBundle.getBundle("Museum", new Locale("en", "US"));  
var itRb = ResourceBundle.getBundle("Museum", new Locale("it", "IT"));
```

```
System.out.println(enRb.getString("greeting") + "! " + enRb.getString("open"));  
System.out.println(itRb.getString("greeting") + "! " + itRb.getString("open"));
```

How to pick resource bundle?

1. `Museum_it_IT.properties` (requested locale)
2. `Museum_it.properties` (requested language with no country)
3. `Museum_en_US.properties` (default locale)
4. `Museum_en.properties` (default locale with no country)
5. `Museum.properties` (bundle with no locale - default bundle)
6. `MissingResourceException` (if non of above were found)

```
// formatting messages (import java.text.*)
```

```
// in property file
```

```
greetByName=Ciao, {0} e {1}
```

```
// in main method
```

```
String greet = itRb.getString("greetByName");
```

```
System.out.println(MessageFormat.format(greet, "Bruno", "Mario"));
```

```
=> Ciao, Bruno e Mario
```

```
// using Properties class to define properties outside rb file
var p = new Properties();
p.setProperty("name", "Natural Museum");

System.out.println("Welcome to " + p.getProperty("name"));
    => Welcome to Natural Museum

// works like a HashMap with String key and values
```