Process of designing web application in such a way that it provides support for various countries, languages, currencies automatically without performing any change in application is called Internationalization (i18N).

**Ex**: If request is from India then response should be Indian people understandable form.

If request is from US then the response should be in US people understandable form.

We can implement internationalization by using following 3 classes:

1. Locale
2. NumberFormat
3. DateFormat

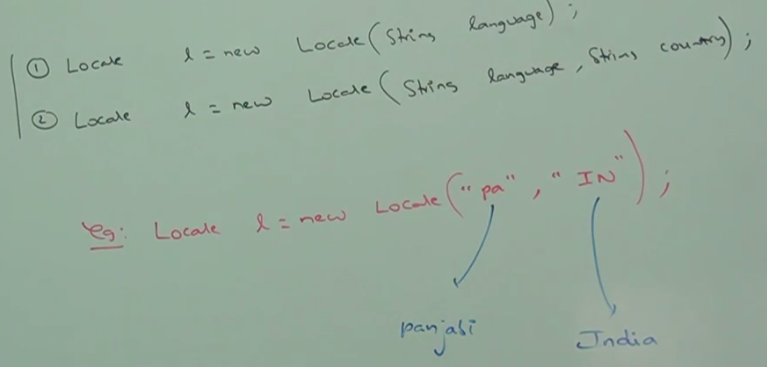
**Locale:**

It represents a geographic location or country or language.

We can create a Locale object to represent India or English language.

Present in **java.util** package. It is a **final** class and direct child class of Object. Implements Serializable and Cloneable interfaces.

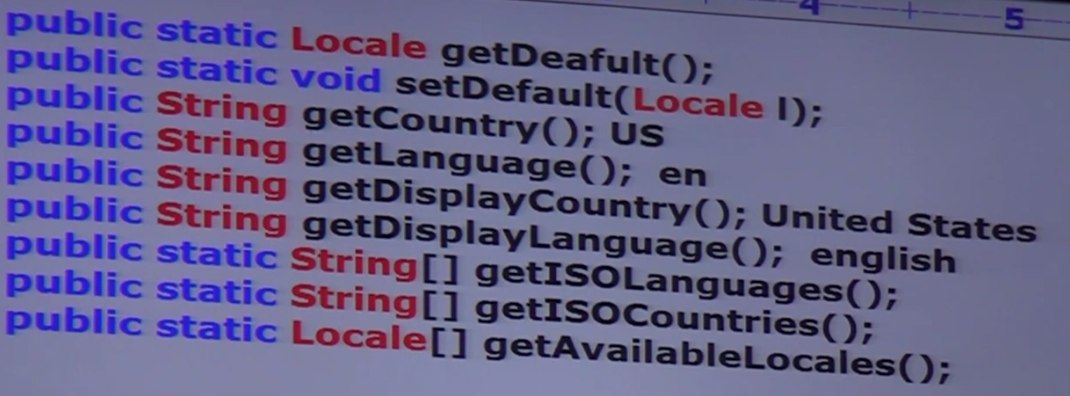
**Constructors:**

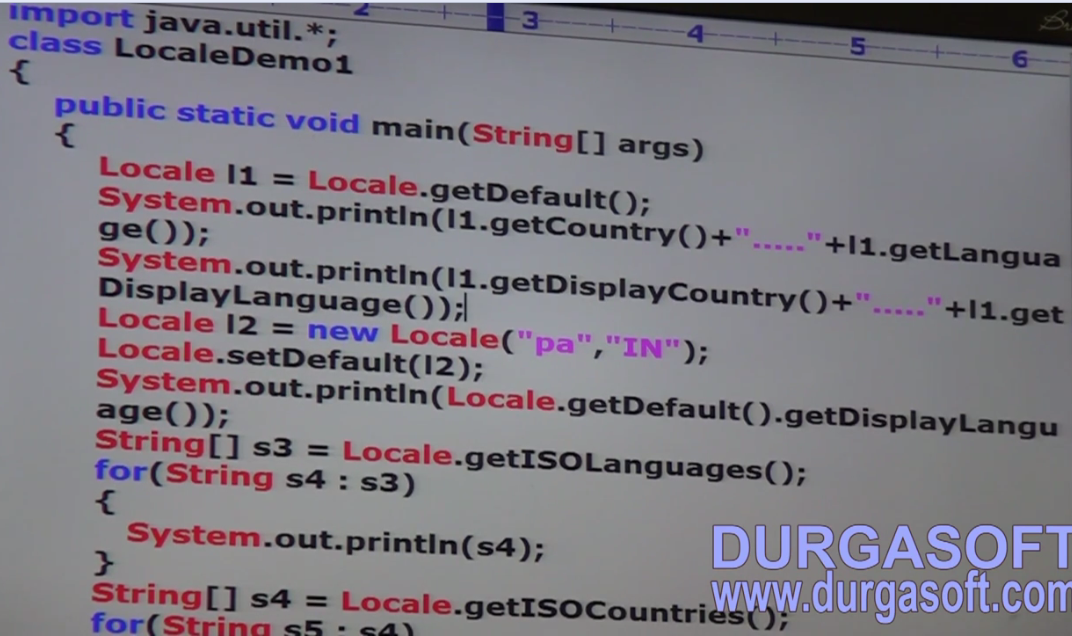


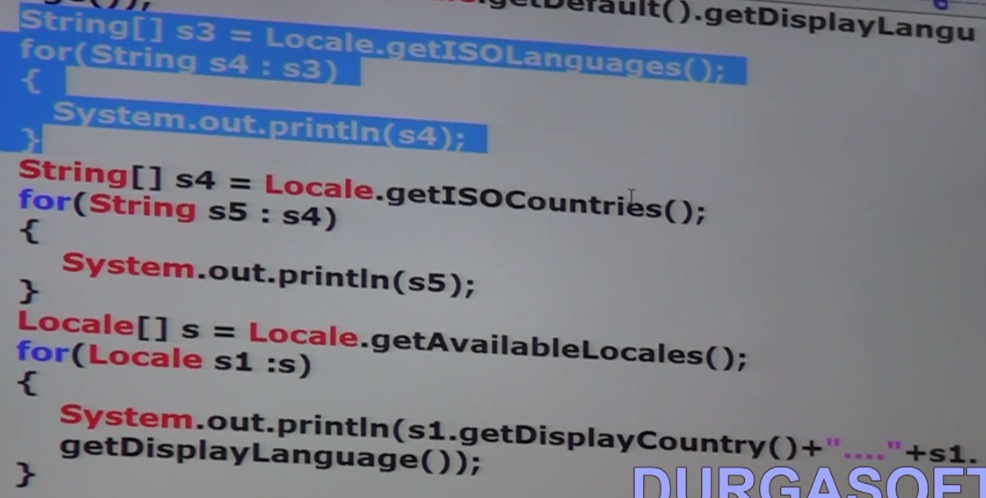
Locale class already defined some constant.

Ex: Locale.US, Locale.UK, Locale.English

**Methods:**







**NumberFormat (java.text package and it is an abstract class)**

Various locations follow various style to represent a java number:

Double d = 123456.789

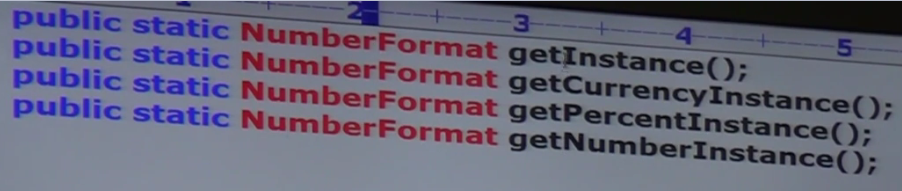
India: 1,23,456.789

US: 123,456.789

We can use NumberFormat class to represent the number.

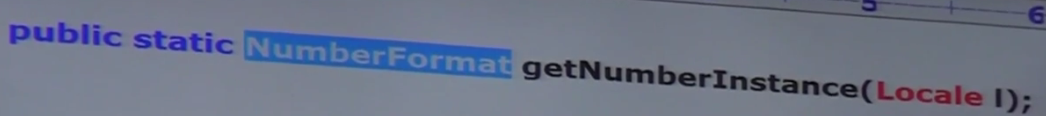
**Number format object for default locale:**

NumberFormat class defines the following methods for this purpose:

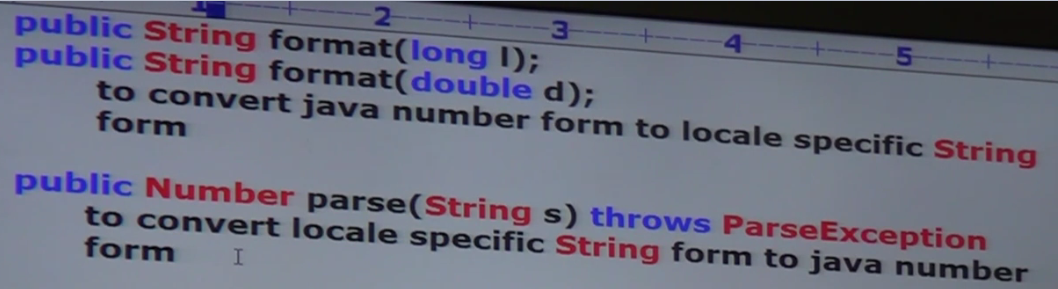


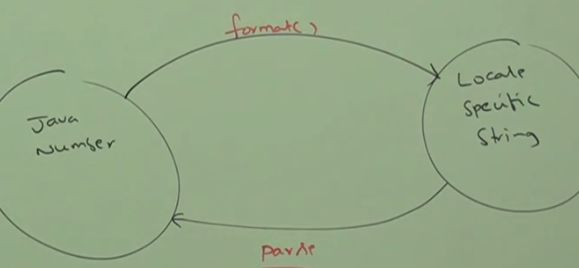
**Number format object for specific locale:**

**Pass locale to the above methods**

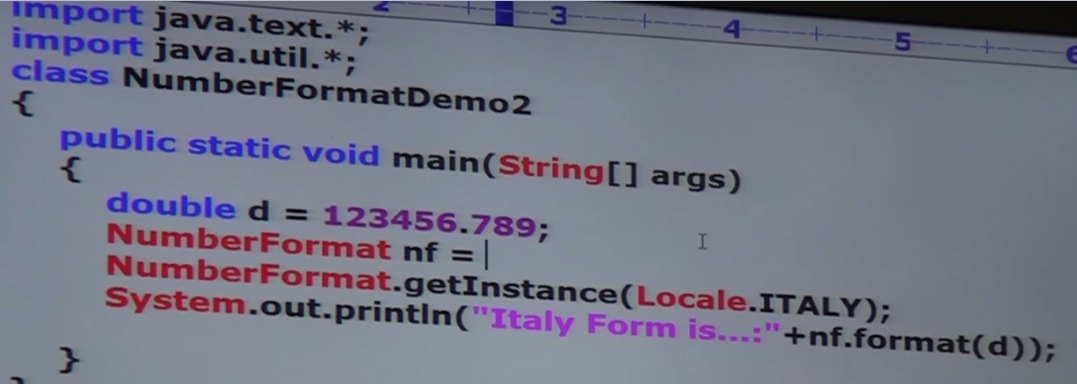


Once we got NumberFormat object we can call format and parse methods on that object.

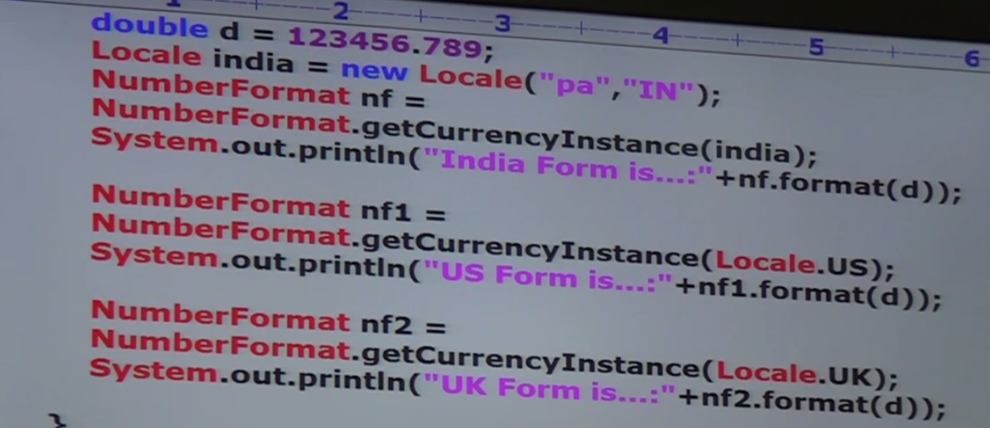




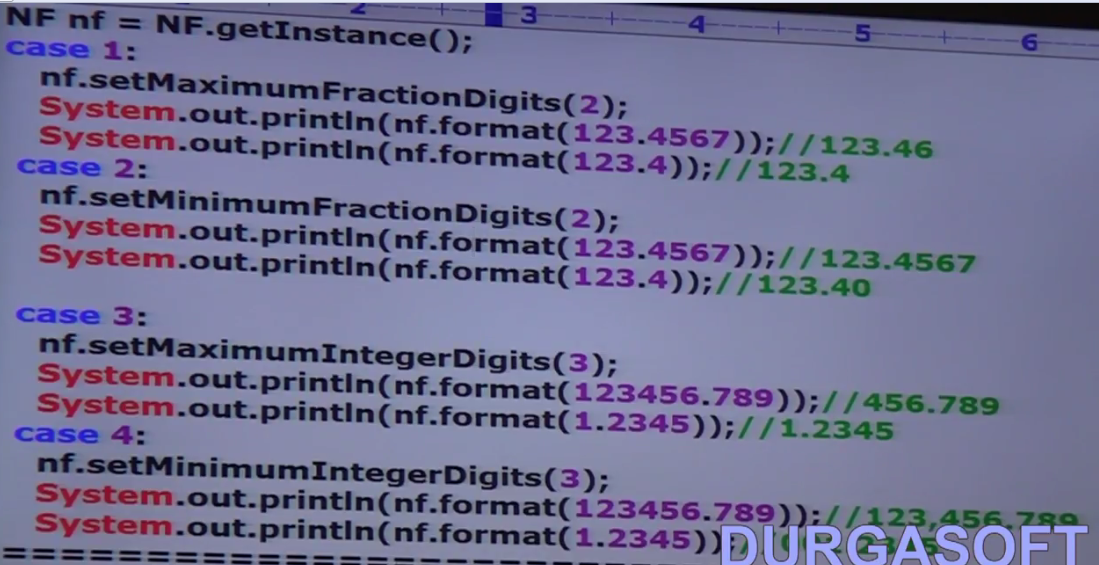
Program to display a number in Italy specific form:

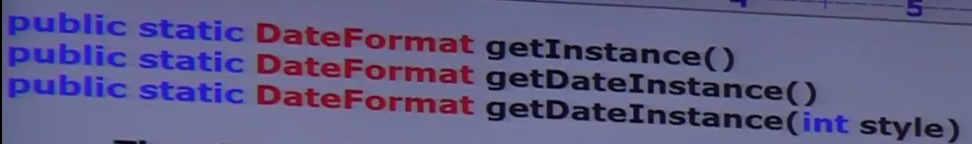


**Program to display number in different currency format:**



Maximum and Minimum fraction digit:





Allowed styles are:

