1. Overview of Internationalization (I18N)

- **Definition:**
  Designing Java applications to support local user conventions and languages.
- **Purpose:**
  To provide services (number, date, and message formatting) as per the user's locale using UNICODE.

---

2. Grouping Users by Locality

To represent local users, Java uses the **java.util.Locale** class. Users are divided based on three parameters:

Parameter	Representation	Example
Language	Two lowercase letters	en, hi, fr
Country	Two uppercase letters	US, IN, FR
System Variant	Three lowercase letters	win, lin, mac

Locale Constructors:

1. **Language Only:**

   ```
   java
   CopyEdit
   Locale l = new Locale("en");
   ```

2. **Language and Country:**

   ```
   java
   CopyEdit
   Locale l = new Locale("en", "US");
   ```

3. **Language, Country, and System Variant:**

   ```
   java
   CopyEdit
   Locale l = new Locale("en", "US", "win");
   ```

---

3. Internationalization Services in Java

A. Number Formatting

- **Class:** `java.text.NumberFormat`
- **Factory Method:**

  ```
  java
  ```

```
CopyEdit
public static NumberFormat getInstance(Locale l)
```

- **Usage Example:**

```
java
CopyEdit
import java.text.NumberFormat;
import java.util.Locale;

public class Main {
    public static void main(String[] args) {
        Locale locale1 = new Locale("en", "US");
        NumberFormat nf1 = NumberFormat.getInstance(locale1);
        System.out.println("en US :" + nf1.format(123456789.34567));

        Locale locale2 = new Locale("it", "IT");
        NumberFormat nf2 = NumberFormat.getInstance(locale2);
        System.out.println("it IT :" + nf2.format(123456789.34567));

        Locale locale3 = new Locale("fr", "FR");
        NumberFormat nf3 = NumberFormat.getInstance(locale3);
        System.out.println("fr FR :" + nf3.format(123456789.34567));
    }
}
```

- **Output Sample:**
  - en US : 123,456,789.346
  - it IT : 123.456.789,346
  - fr FR : 123 456 789,346

B. Date Formatting

- **Class:** `java.text.DateFormat`
- **Factory Method:**

```
java
CopyEdit
public static DateFormat getDateInstance(int dateStyle, Locale l)
```

(where `dateStyle` may be 0, 1, 2, or 3)

- **Usage Example:**

```
java
CopyEdit
import java.text.DateFormat;
import java.util.Date;
import java.util.Locale;

public class Main {
    public static void main(String[] args) {
        Locale locale1 = new Locale("en", "US");
```

```
        DateFormat dateFormat1 = DateFormat.getDateInstance(0,
locale1);
        System.out.println(dateFormat1.format(new Date()));

        Locale locale2 = new Locale("it", "IT");
        DateFormat dateFormat2 = DateFormat.getDateInstance(0,
locale2);
        System.out.println(dateFormat2.format(new Date()));

        Locale locale3 = new Locale("fr", "FR");
        DateFormat dateFormat3 = DateFormat.getDateInstance(0,
locale3);
        System.out.println(dateFormat3.format(new Date()));

        Locale locale4 = new Locale("te", "IN");
        DateFormat dateFormat4 = DateFormat.getDateInstance(0,
locale4);
        System.out.println(dateFormat4.format(new Date()));
    }
}
```

- **Output Sample (if run on November 17, 2024):**
  - en US: Sunday, November 17, 2024
  - it IT: domenica 17 novembre 2024
  - fr FR: dimanche 17 novembre 2024
  - te IN: 17, నవంబర్ 2024, ఆదివారం

C. Message Formatting

- **Step 1:** Prepare properties files with key-value pairs (keys in English, values in local language).
  File Naming Format:
  `baseName_lang_country.properties`

  Example Files:

  abc_en_US.properties

  ```
  ini
  CopyEdit
  welcome=Welcome To en US Users.
  ```

  abc_it_IT.properties

  ```
  ini
  CopyEdit
  welcome=Welcomeo teo it IT Userseo.
  ```

  abc_hi_IN.properties

  ```
  ini
  CopyEdit
  welcome=Aapka Swaghat Hi.
  ```

- **Step 2:** Create `Locale` and `ResourceBundle` objects.
  - **Factory Method:**

    ```
    java
    CopyEdit
    public static ResourceBundle getBundle(String baseName, Locale
    l)
    ```

- **Step 3:** Retrieve message using `getString` method.

```
java
CopyEdit
import java.util.Locale;
import java.util.ResourceBundle;

public class Main {
    public static void main(String[] args) {
        Locale l1 = new Locale("en", "US");
        ResourceBundle rb1 = ResourceBundle.getBundle("abc", l1);
        System.out.println("en US : " + rb1.getString("welcome"));

        Locale l2 = new Locale("it", "IT");
        ResourceBundle rb2 = ResourceBundle.getBundle("abc", l2);
        System.out.println("it IT : " + rb2.getString("welcome"));

        Locale l3 = new Locale("hi", "IN");
        ResourceBundle rb3 = ResourceBundle.getBundle("abc", l3);
        System.out.println("hi IN : " + rb3.getString("welcome"));
    }
}
```

- **Output Sample:**
  - en US : Welcome To en US Users.
  - it IT : Welcomeo Teo it IT Userseo.
  - hi IN : Aapka Swaghat Hi.