Παρακάτω περιγράφονται τα βήματα για την μεταγλώττιση και εκτέλεση προγραμμάτων Java RMI:

1. **Εκκίνηση κελύφους Linux**: Πρώτα πρέπει να έχουμε ανοιχτό ένα παράθυρο κελύφους Linux ώστε να εισάγουμε εντολές.
2. **Έλεγχος εντολών Java**: Έπειτα ελέγχουμε αν λειτουργούν ο μεταγλωττιστής και ο διερμηνευτής της Java μετά την εγκατάσταση της. Γι αυτό εισάγουμε τις παρακάτω εντολές στο κέλυφος Linux:

   ```
   $ which javac
   $ which java
   ```

   Θα πρέπει να εμφανίζονται στην οθόνη οι πλήρεις διαδρομές.
3. **Δημιουργία ενός καταλόγου**: Δημιουργούμε ένα κατάλογο ώστε να αποθηκεύσουμε τα παραδείγματα των προγραμμάτων μας. Έτσι, δημιουργούμε ένα κατάλογο με όνομα `javarmi_lab` και μπαίνουμε σε αυτό τον κατάλογο. Οι εντολές είναι οι εξής:

   ```
   $ mkdir javarmi_lab
   $ cd javarmi_lab
   ```

4. **Δημιουργία απομακρυσμένης διεπαφής**: Εισάγουμε την παρακάτω διεπαφή σε ένα απλό επεξεργαστή κειμένου (όπως το kedit):

   ```
   import java.rmi.*;
   public interface Factorial extends Remote
   {
         // ypografh ths apomakrysmenhs methodoy
         public int fact(int number) throws RemoteException;
   }
   ```

   Αφού εισάγουμε την παραπάνω διεπαφή, την αποθηκεύουμε σε ένα αρχείο με όνομα `Factorial.java` κάτω από το κατάλογο `javarmi_lab`. Εναλλακτικά μπορούμε να μεταφορτώσουμε την διεπαφή Factorial.java στο κατάλογο `javarmi_lab`.
5. **Υλοποίηση απομακρυσμένης διεπαφής**: Εισάγουμε την παρακάτω κλάση που περιέχει την υλοποίηση της απομακρυσμένης διεπαφής που ορίζαμε προηγουμένως σε ένα απλό επεξεργαστή κειμένου (όπως το kedit):

   ```
   import java.rmi.*;
   import java.rmi.server.*;

   // H klash ayth ylopoiei thn apomakrysmenh diepafh Factorial
   public class FactorialImpl extends UnicastRemoteObject implements
   Factorial
   {
         // Kataskeyasths
         public FactorialImpl() throws RemoteException
         {
               super();
         }

         // Kodikas ylopoihshs ths apomakrysmenhs methodoy
         public int fact(int number) throws RemoteException
         {
   ```

```
            int result = 1;

            for(int i = 1; i <= number; i++)
                  result = result * i;
            return (result);
      }
}
```

Αφού εισάγουμε την παραπάνω κλάση, την αποθηκεύουμε σε ένα αρχείο με όνομα `FactorialImpl.java` κάτω από το κατάλογο `javarmi_lab`. Εναλλακτικά μπορούμε να μεταφορτώσουμε την κλάση FactorialImpl.java στο κατάλογο `javarmi_lab`.

6. **Ανάπτυξη προγράμματος διακομιστή**: Εισάγουμε την παρακάτω κλάση που περιέχει το κώδικα του διακομιστή σε ένα απλό επεξεργαστή κειμένου (όπως το kedit):

```
import java.rmi.*;
public class FactorialServer
{
      private static final String HOST = "localhost";
      public static void main(String[] args) throws Exception
      {
            // Dhmioyrgia antikeimenoy
            FactorialImpl robj = new FactorialImpl();
            // Eggrafh toy antikeimenoy sthn yphresia onomasias RMI
            // kato apo to onoma Factorial
            String rmiObjectName = "rmi://" + HOST + "/Factorial";
            Naming.rebind(rmiObjectName,robj);
      }
}
```

Αφού εισάγουμε την παραπάνω κλάση, την αποθηκεύουμε σε ένα αρχείο με όνομα `FactorialServer.java` κάτω από το κατάλογο `javarmi_lab`. Εναλλακτικά μπορούμε να μεταφορτώσουμε την κλάση FactorialServer.java στο κατάλογο `javarmi_lab`.

7. **Ανάπτυξη προγράμματος πελάτη**: Εισάγουμε την παρακάτω κλάση που περιέχει το κώδικα του πελάτη σε ένα απλό επεξεργαστή κειμένου (όπως το kedit):

```
import java.rmi.*;
public class FactorialClient
{
      private static final String HOST = "localhost";
      public static void main(String[] args)
      {
            try
            {
                  // Anazhtish toy apomakrysmenoy antikeimenoy kai metatropi thn
                  // anafora toy se klash apomakrysmenhs diepafhs Factorial
                  Factorial ref = (Factorial) Naming.lookup("rmi://" + HOST +
                  "/Factorial");
                  // Klhsh ths apomakrysmenhs methodoy
                  int result = ref.fact(3);
                  System.out.println("The factorial of 3 is " + result);
            }
            catch (RemoteException re)
            {
                  System.out.println("Remote Exception");
                  re.printStackTrace();
```

```
            }
            catch (Exception e)
            {
                  System.out.println("Other Exception");
                  e.printStackTrace();
            }
      }
}
```

Αφού εισάγουμε την παραπάνω κλάση, την αποθηκεύουμε σε ένα αρχείο με όνομα `FactorialClient.java` κάτω από το κατάλογο `javarmi_lab`. Εναλλακτικά μπορούμε να μεταφορτώσουμε την κλάση FactorialClient.java στο κατάλογο `javarmi_lab`.

8. **Μεταγλώττιση Java**: Για να μεταγλωττίσουμε τις παραπάνω κλάσεις Java σε εκτελέσιμα αρχεία πληκρολογούμε τις παρακάτω εντολές στο κέλυφος ως εξής:

```
$ javac Factorial.java
$ javac FactorialImpl.java
$ javac FactorialServer.java
$ javac FactorialClient.java
```

Έπειτα εισάγουμε την εντολή `ls` για να δούμε τα αρχεία `class` που δημιούργησε ο μεταγλωττιστής στον τρέχοντα μας κατάλογο.

9. **Μεταγλώττιση κλάσης διεπαφής**: Για να μεταγλωττίσουμε το αρχείο της κλάσης που υλοποιεί την διεπαφή δηλαδή το `FactorialImpl` πληκτρολογούμε την παρακάτω εντολή στο κέλυφος ως εξής:

```
$ rmic FactorialImpl (δεν χρειάζεται να γράψουμε την επέκταση .class)
```

Έπειτα εισάγουμε την εντολή `ls` για να δούμε το αρχείο στέλεχος `FactorialImpl_stub.class` που δημιούργησε ο μεταγλωττιστής στον τρέχοντα μας κατάλογο. Το αρχείο αυτό είναι απαραίτητο για την εκτέλεση της εφαρμογής.

10. **Εκκίνηση υπηρεσίας ονομασίας RMI**: Για να εκκινήσουμε την υπηρεσία ονομασία του συστήματος RMI στον διακομιστή εισάγουμε την παρακάτω εντολή στο κέλυφος ως εξής:

```
$ rmiregistry &
```

Η παραπάνω εντολή έχει σαν αποτέλεσμα να εκτελείται η υπηρεσία ονομασίας στο παρασκήνιο.

11. **Εκτέλεση**: Για να εκτελέσουμε τα δύο προγράμματα διακομιστή και πελάτη εισάγουμε διαδοχικά τις παρακάτω εντολές στο κέλυφος:

```
$ java FactorialServer & (δεν χρειάζεται να γράψουμε την επέκταση
      .class)
$ java FactorialClient (δεν χρειάζεται να γράψουμε την επέκταση
      .class)
```

Τέλος, πρέπει να έχουμε την παρακάτω έξοδος στην οθόνη μας:

```
The factorial of 3 is 6
```