RMI IMPLEMENTATION IN CLOUD ENVIRONMENT

Name: Nagasurya

Roll No:2023115106

Aim To design and implement a **Remote Method Invocation (RMI)**
based
distributed application using **Java**, where the **RMI server is hosted in**
a cloud environment (Microsoft Azure Virtual Machine) and the
client executes on a local machine, enabling invocation of remote
methods over a network.

Software & Tools Used

Component	Description
Programming Language	Java
Distributed Model	Java RMI
Cloud Platform	Microsoft Azure
Server OS	Ubuntu Linux (Azure VM)
Client OS	Windows 10
JDK Version	OpenJDK 17
Network Port	1009

System Architecture

- RMI Server and Registry run on **Azure VM**
- Client accesses remote object using **public IP**
- Communication occurs over TCP/IP

Remote Methods Implemented

Method Name	Description
multiplyMatrix	Perform Matrix Multiplication
getPrimes	Print prime numbers in a range

Execution Steps

Step 1: Install Java on Azure VM

sudo apt update

sudo apt install openjdk-17-jdk -y

Step 2: Compile RMI Programs

javac *.java

Step 3: Run RMI Server

java RMIServer

Output:

RMI Server running on Azure VM at port 1099

```
○ ♠ ~ > ssh -i ~/azureuser.pem azureuser@98.70.25.35          ○ 12:37:38 AM
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1017-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Thu Jan 29 19:07:38 UTC 2026

  System load:  0.03       Processes:             160
  Usage of /:   9.6% of 28.02GB  Users logged in:       1
  Memory usage: 5%         IPv4 address for eth0: 172.17.0.4
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

34 updates can be applied immediately.
27 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

4 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Thu Jan 29 18:58:15 2026 from 49.206.15.107
azureuser@VM01:~$ cd RMI/
azureuser@VM01:~/RMI$ java RMIServer
RMI Server running on port 1099
```

Step 4: Configure Azure NSG

- Allow **Inbound TCP Port 8080**

  . Source: Any

  · Destination: Any

Step 5: Run Client on Local Machine

java RMIClient
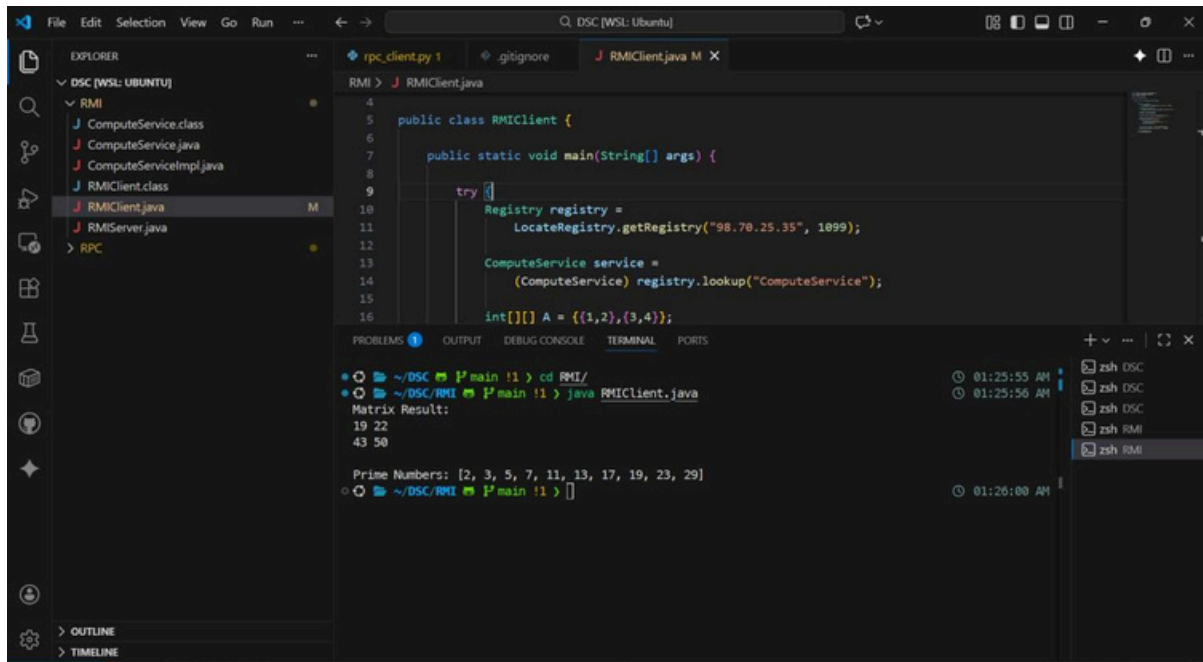
Error Handling

- RemoteException handled in all methods

- Network timeout handled via try–catch

- Registry lookup failure handled

Result Thus, a **Java RMI-based distributed application** was successfully
implemented and executed in a **cloud environment using Microsoft Azure**, allowing a local client to invoke methods on a remote server and receive correct outputs.

Conclusion

Java RMI provides a robust mechanism for building distributed applications by allowing remote method calls with object-oriented abstraction. Hosting the RMI server on Azure ensures scalability and remote accessibility.