Lab 6: Distributed Object with Java

BITP 3123 DISTRIBUTED APPLICATION DEVELOPMENT

EMALIANA BINTI KASMURI
FAKULTI TEKNOLOGI MAKLUMAT DAN KOMUNIKASI

Table of Contents

Lab 6: Distributed Object with Java

Learning Outcomes

When the student finished all the exercises, the student should be able to,

1. Execute a simple RMI application.
2. Develop simple RMI application

Reference Materials

1. Lecture slides from week 5.

Tools

1. Eclipse for Java EE
2. Terminal or MS-DOS Prompt

Exercise 1: Executing a Simple RMI Application

Part 1: Importing RMI files

1. This exercise requires you to download and extract a file named `'rmi java example.zip'` from Resources for lecture on week 5 in ulearn.
2. Create new Java project. Name the project as **sensorrmi**.
3. Import the extracted file into Eclipse.
4. Move to the classes to the respective package to rectify any error.

Part 2: Loading RMI Registry

1. Open a terminal or MS DOS prompt.
2. Change the directory to the **sensorrmi/bin** directory.
3. Type `rmiregistry` on the terminal.
4. Leave the terminal open.

Part 3: Executing Server-Side RMI Application

1. Ensure your location is in **sensorrmi/bin**.
2. Execute `TemperatureServerRMIApp`.
3. A message that indicate a sensor is successfully registered should be displayed on the screen.

Part 4: Execute Client-Side RMI Application

1. Ensure your location is in **sensorrmi/bin**.
2. Execute `TemperatureClientRMIApp`.
3. A message that display the current temperature should be appear in the terminal.

Exercise 2: Creating A New Remote Object

A new temperature sensor is deployed at Ayer Keroh. Register a new remote object that represent the sensor in Ayer Keroh. Display an appropriate message to indicate the sensor is successfully registered in the RMI registry.

Hint: This exercise should be implemented in TemperatureServerRMIApp.

Exercise 3: Creating A New RMI Client

Create a new client-side application to display the temperature in Ayer Keroh. The client should display the current temperature in Ayer Keroh.

Exercise 4: Defining A New Interface Method

Define a new interface method in `TemperatureSensor` to retrieve a temperature for a specified day.

Exercise 5: Load Temperature Data

Create a new method to load the following data in a HashMap. Use day as key and temperature as value.

Table 1: A week temperature in Melaka

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
32	31	33	35	36	33	33

Hint: Refer https://www.programiz.com/java-programming/hashmap

Exercise 6: Overriding Inherited Method

Override the method defined in Exercise 4. This method should return the temperature from the from the structure defined in Exercise 5.

Exercise 7: Invoking New Remote Method

Invoke the method defined in Exercise 4 in the new class created in Exercise 3. Display the value returned by the method.

Exercise 8: Computing Average Temperature

The Ayer Keroh Weather Station needs to display an average of temperature for a week. This function is also required by other stations in Melaka. Implement this requirement using RMI technique.

How to submit the exercises?

1. Upload your solution in github.
2. Specify the github link in ulearn.