Advanced Java Application Development for the BlackBerry Smartphone Lab manual
REALIZE ANY EXPECTED SAVINGS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF BUSINESS OPPORTUNITY, OR CORRUPTION OR LOSS OF DATA, FAILURES TO TRANSMIT OR RECEIVE ANY DATA, PROBLEMS ASSOCIATED WITH ANY APPLICATIONS USED IN CONJUNCTION WITH RIM PRODUCTS OR SERVICES, DOWNTIME COSTS, LOSS OF THE USE OF RIM PRODUCTS OR SERVICES OR ANY PORTION THEREOF OR OF ANY AIRTIME SERVICES, COST OF SUBSTITUTE GOODS, COSTS OF COVER, FACILITIES OR SERVICES, COST OF CAPITAL, OR OTHER SIMILAR PECUNIARY LOSSES, WHETHER OR NOT SUCH DAMAGES WERE FORESEEN OR UNFORESEEN, AND EVEN IF RIM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, RIM SHALL HAVE NO OTHER OBLIGATION, DUTY, OR LIABILITY WHATSOEVER IN CONTRACT, TORT, OR OTHERWISE TO YOU INCLUDING ANY LIABILITY FOR NEGLIGENCE OR STRICT LIABILITY.

THE LIMITATIONS, EXCLUSIONS, AND DISCLAIMERS HEREIN SHALL APPLY: (A) IRRESPECTIVE OF THE NATURE OF THE CAUSE OF ACTION, DEMAND, OR ACTION BY YOU INCLUDING BUT NOT LIMITED TO BREACH OF CONTRACT, NEGLIGENCE, TORT, STRICT LIABILITY OR ANY OTHER LEGAL THEORY AND SHALL SURVIVE A FUNDAMENTAL BREACH OR BREACHES OR THE FAILURE OF THE ESSENTIAL PURPOSE OF THIS AGREEMENT OR OF ANY REMEDY CONTAINED HEREIN; AND (B) TO RIM AND ITS AFFILIATED COMPANIES, THEIR Successors, assigns, agents, Suppliers (including Airtime Service Providers), Authorized RIM Distributors (also including airtime Service Providers) and their respective Directors, employees and independent contractors.

IN ADDITION TO THE LIMITATIONS AND EXCLUSIONS SET OUT ABOVE, IN NO EVENT SHALL ANY DIRECTOR, EMPLOYEE, AGENT, DISTRIBUTOR, SUPPLIER, INDEPENDENT CONTRACTOR OF RIM OR ANY AFFILIATES OF RIM HAVE ANY LIABILITY ARISING FROM OR RELATED TO THE DOCUMENTATION.

Prior to subscribing for, installing or using any Third Party Products and Services it is your responsibility to ensure that your airtime service provider has agreed to support all of their features. Some airtime service providers may not offer Internet browsing functionality with a subscription to BlackBerry® Internet Service. Check with your service provider for availability, roaming arrangements, service plans and features. Installation or use of Third Party Products and Services with RIM’s products and services may require one or more patent, trademark, copyright or other licenses in order to avoid infringement or violation of third party rights. You are solely responsible for determining whether to use, Third Party Products and Services and if any third party licenses are required to do so. If required you are responsible for acquiring them. You should not install or use Third Party Products and Services until all necessary licenses have been acquired. Any Third Party Products and Services that are provided with RIM’s products and services are provided as a convenience to you and are provided “AS IS” with no express or implied conditions, endorsements, guarantees, representations or warranties of any kind by RIM and RIM assumes no liability whatsoever, in relation thereto. Your use of Third Party Products and Services shall be governed by and subject to you agreeing to the terms of separate licenses and other agreements applicable thereto with third parties, except to the extent expressly covered by a license or other agreement with RIM.

Certain features outlined in this documentation require a minimum version of BlackBerry® Enterprise Server software, BlackBerry® Desktop Software, and/or BlackBerry® Device Software and may require additional development or Third Party Products and Services for access to corporate applications.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/) and/or licensed pursuant to Apache License, Version 2.0 (http://www.apache.org/licenses/). For more information, see the NOTICE.txt file included with the software. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
The terms of use of any RIM product or service are set out in a separate license or other agreement with RIM applicable thereto. NOTHING IN THIS DOCUMENTATION IS INTENDED TO SUPERSEDE ANY EXPRESS WRITTEN AGREEMENTS OR WARRANTIES PROVIDED BY RIM FOR PORTIONS OF ANY RIM PRODUCT OR SERVICE OTHER THAN THIS DOCUMENTATION.
Chapter 7: Understanding client/server push

In the labs in this section, you write an application that uses HTTP to push data from a server application to a client application on a BlackBerry® smartphone.

The tasks in these labs require the files RIMPushClient.java and RIMPushServer.java, which you can retrieve from the Lab_05_RIMPushClient project and Lab_05_RIMPushserver project, respectively. You can also refer to the completed versions of the projects, Lab_05_RIMPushClient_complete and Lab_05_RIMPushServer_complete.
7.8 RIM push—create a new configuration

In this exercise, you will create a new BlackBerry® Smartphone Simulator configuration.

A push message is sent as a byte array to a defined port on the BlackBerry simulator. To route push messages to the BlackBerry simulator, you can run the BlackBerry® MDS simulator on the localhost. Use the device PIN of 0x2100000A and listen to data port 100.

1. Click Run > Run Configurations.
2. Right-click on the BlackBerry Simulator type and select New.
3. Enter the following name for the new configuration: Rim Push Lab Configuration.
4. Click the Simulator tab, and select the profile 9000 - JDE.
5. Click the General tab, and select the check box Launch MDS-CS with simulator.
6. Make sure that the PIN is 0x2100000A.
7. Click on the Advanced tab.
8. Click Apply, then Close.
9. If prompted to save changes, click Yes.
7.9 RIM push—create server component

In this exercise, you will configure an HttpURLConnection and then configure the connection to send data.

For more information about other request properties, see the BlackBerry Java Development Environment Guide.

1. Generate a URL with the following format:
   
   http://<host>:<port>/push?DESTINATION=<device pin>&PORT=<device_port>&REQUESTURI=<post uri>
   
   Where the <host> is MDS that resides on localhost, the Port is 8080, the device port is 100, and the requesturi is localhost. In the getPushURL() method of the RIMPushServer class, complete the TODO step titled Generate URL.
   
   ```java
   pushURL = new URL("http", MDS_HOST, MDS_PORT, "/push?DESTINATION=" + pin + "&PORT=" + DEVICE_PORT + "&REQUESTURI=localhost");
   ```

2. Call a method on the URL to return a connection. Because you want to use a POST request, you must typecast it to an HttpURLConnection. Before you can send data, configure the HttpURLConnection to output messages using the POST method and to configure the X-RIM-PUSH-ID property of the request. This operation has already been implemented. In the postData() method of the RIMPushServer class, complete the TODO step titled Open Connection.
   
   ```java
   HttpURLConnection conn = (HttpURLConnection) url.openConnection();
   ```

3. Create an OutputStream object and send the data using the write() method. After closing the OutputStream, call conn.getInputStream(). This allows MDS to recognize the push request. In the postData() method of the RIMPushServer class, complete the TODO step titled Send Data.
   
   ```java
   OutputStream out = conn.getOutputStream();
   out.write(data.getBytes());
   ```
7.10 RIM push—create client component

In this exercise, you will create the client component, which listens on a port for incoming messages. Because the operation that obtains data from a port blocks the UI, you must perform the operation in a separate thread.

1. Open the RIMPushClient.java file in the editor

2. Create an instance of ListeningThread and invoke the start() method. In the RIMPushClient constructor, complete the TODO step titled Create Thread.
   ```java
   _listeningThread = new ListeningThread();
   _listeningThread.start();
   ```

3. To create an instance of a StreamConnectionNotifier, you can use the Connector.open() method, passing it a URL that includes the port to listen on, in the following form:
   ```java
   http://:100;deviceside=false
   ```
   Connector (javax.microedition.io) is a factory class used to open many types of connections. In this case you typecast the return value to StreamConnectionNotifier. In the run() method of the ListeningThread class, complete the TODO step titled Create Notifier.
   ```java
   notifier = (StreamConnectionNotifier) Connector.open(URL + ";deviceside=false");
   ```

4. Call the appropriate StreamConnection method to return the InputStream. In the run() method of the ListeningThread class, complete the TODO step titled Open InputStream.
   ```java
   input = conn.openInputStream();
   ```

5. Invoke the InputStream read(byte[]) method until you have read all the data. The read method returns -1 when there is no data left to read. In the run() method of the ListeningThread class, complete the TODO step titled Extract Message.
   ```java
   while (-1 != (len = input.read(data)))
   ```

6. Display the message data on the UI screen. Because the updateMessage() method is called from a thread outside of the main event thread, you can call the invokeLater() method on your UiApplication. In the updateMessage() method of the RIMPushClientScreen class, complete the TODO step titled Display Message.
   ```java
   _msgField.setText(new String(data));
   ```
7.11 RIM push—run project

In this exercise, you will run the server application and run the client application on the BlackBerry Smartphone Simulator.

1. Run the server project. In the Package Explorer view, right-click the Lab_05_RimPushServer project, and then click Run As > Java Application.

2. Activate the client project. In the Package Explorer view, right-click the Lab_05_RimPushClient project, and then click Activate for BlackBerry from the menu.

3. Launch BlackBerry simulator.
   a. In the Package Explorer view, right-click the Lab_05_RimPushClient project, and then click Run As from the menu.
   b. Select BlackBerry Simulator.

4. Run the application
   a. When the BlackBerry Smartphone Simulator home screen appears, click the menu key to display all applications and folders.
   b. Use the arrow keys on your keyboard to navigate to the Downloads folder.
   c. Open the folder by pressing the Enter key on your keyboard.
   d. Use the arrow keys on your keyboard to navigate to the generic application icon with the title RIM Push Lab.
   e. Invoke the client application by pressing the Enter key on your keyboard.
   f. Type a message in the text area of the server component, and then click Send. The message should appear on the client screen in the BlackBerry Smartphone Simulator.