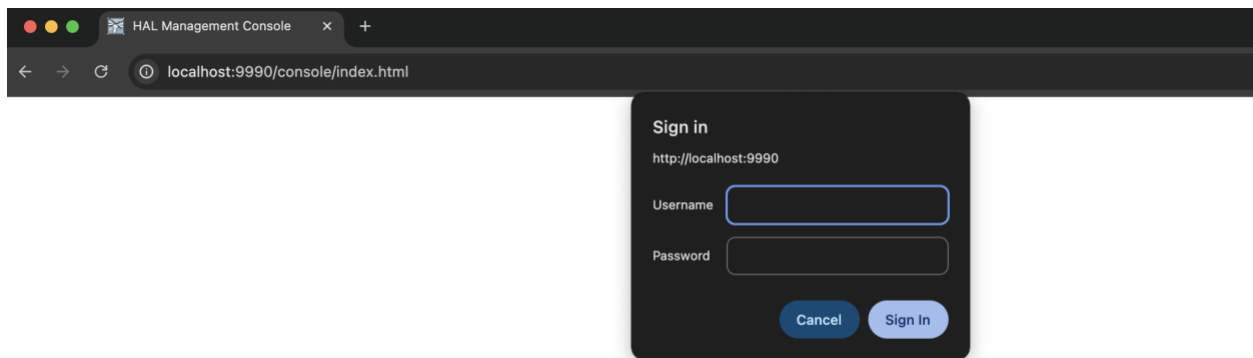


ITE5432 - J2EE Business Components

Lab Activity 2

Step 1:

Enter the following URL in web browser: <http://localhost:9990/>



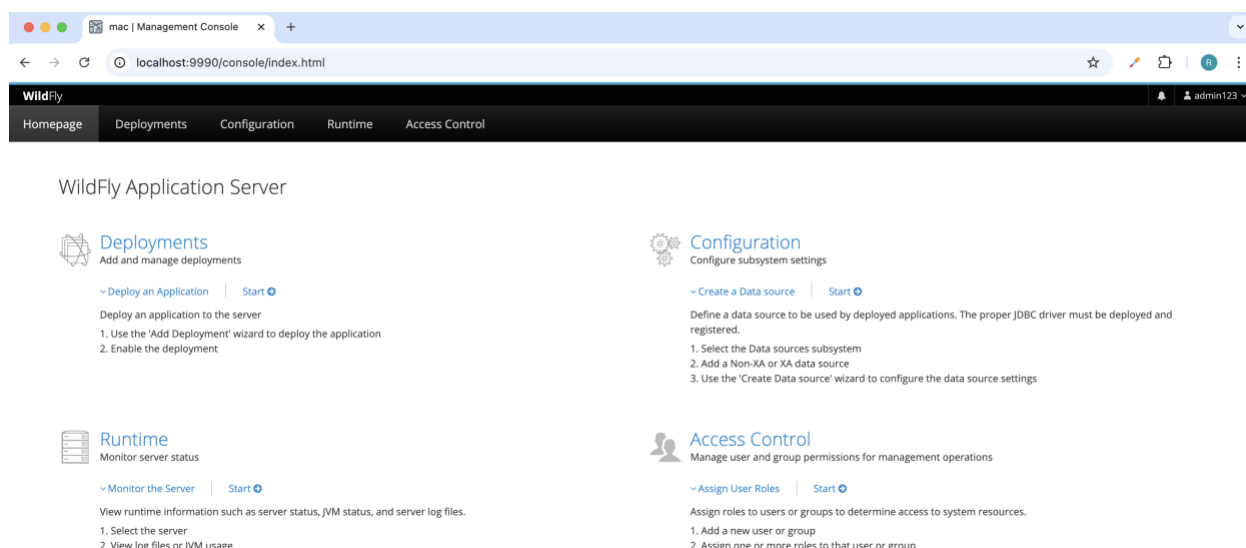
The default username is **admin**. There is no default password. You can set password for the admin user or can create a new user. To add a user, navigate to **wildfly-35.0.1.Final -> bin** folder.

- Windows user: Execute add-user.bat
- Mac user: Execute ./add-user.sh

Add a user with the following credentials:

- Username: admin123
- Password: admin123

Upon successful login, you will the following homepage.

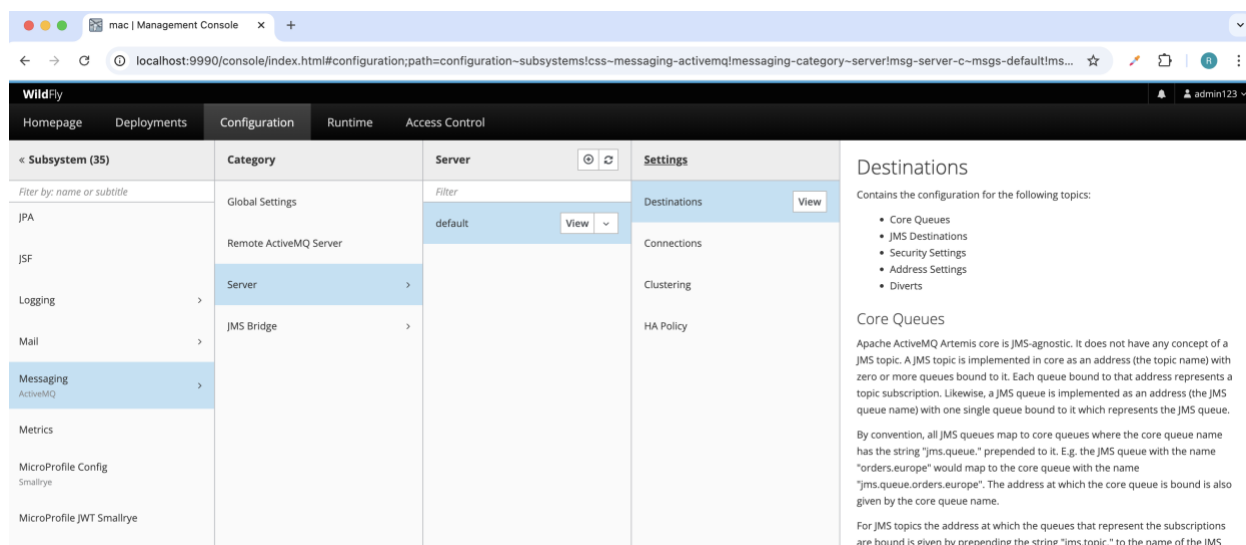


Step 2:

We need to add a JMS Queue to send and receive messages. To add a JMS Queue, navigate to **Configuration → Subsystems → Messaging → Server → default → Destinations**.

Note: If WildFly server is not running with full profile, you may not see the Messaging Tab. Refer to Step 3 to run WildFly server with full profile.

Or you can directly run WildFly server with the following command. Navigate to **wildfly-35.0.1.Final → bin** and execute the following command **./standalone.sh -c standalone-full.xml** through command line.



Click on **View**

The screenshot shows the WildFly Management Console interface. The breadcrumb navigation is: < Back / Configuration => Subsystems / Subsystem => Messaging / Category => Server / Server => default / Settings => Destinations. The left sidebar has a menu with 'Core Queue' selected. The main content area is titled 'Core Queue' and describes it as 'A Queue.'. It shows a table with columns 'Name' and 'Queue Address'. The table is empty with the message 'No data available in table'. Below the table, there is a 'Filter' section with a 'Durable' property set to 'true'. Navigation controls at the bottom of the table show '1 of 0' entries.

Click on **JMS Queue**

The screenshot shows the WildFly Management Console interface with the 'JMS Queue' page selected. The breadcrumb navigation is: < Back / Configuration => Subsystems / Subsystem => Messaging / Category => Server / Server => default / Settings => Destinations. The left sidebar has 'JMS Queue' selected. The main content area is titled 'JMS Queue' and describes it as 'Defines a Jakarta Messaging queue.'. It shows a table with columns 'Name' and 'Selector'. The table contains three entries: 'DLQ', 'ExpiryQueue', and 'TestQueue'. Below the table, there is a 'Selector' section with 'Entries' and a 'Durable' property set to 'true'. Navigation controls at the bottom of the table show '1 of 1' entries.

To add a JMS Queue, click on **Add** and input the required information as outlined below:

Name: TestQueue

Entries: java:/jms/queue/TestQueue

Press **enter** to add entries.

Add JMS Queue

×

[? Help](#)

Name

TestQueue

Durable

🔗

ON

Selector

Entries *

java:/jms/queue/TestQueue

Press ↵ to add new items and ⌘ to remove them.

Required fields are marked *

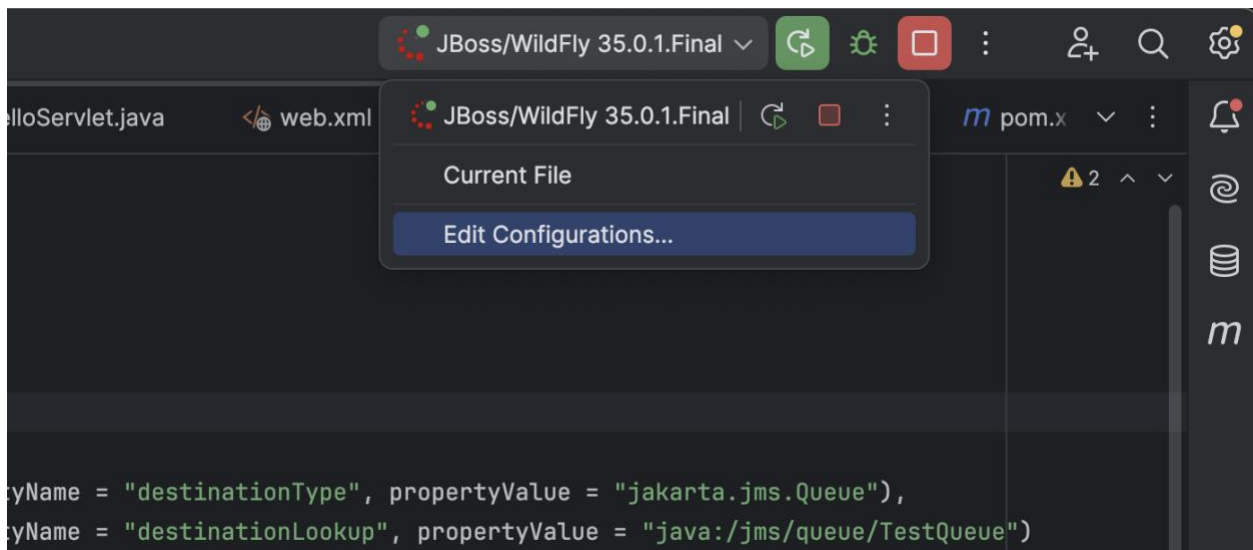
Cancel

Add

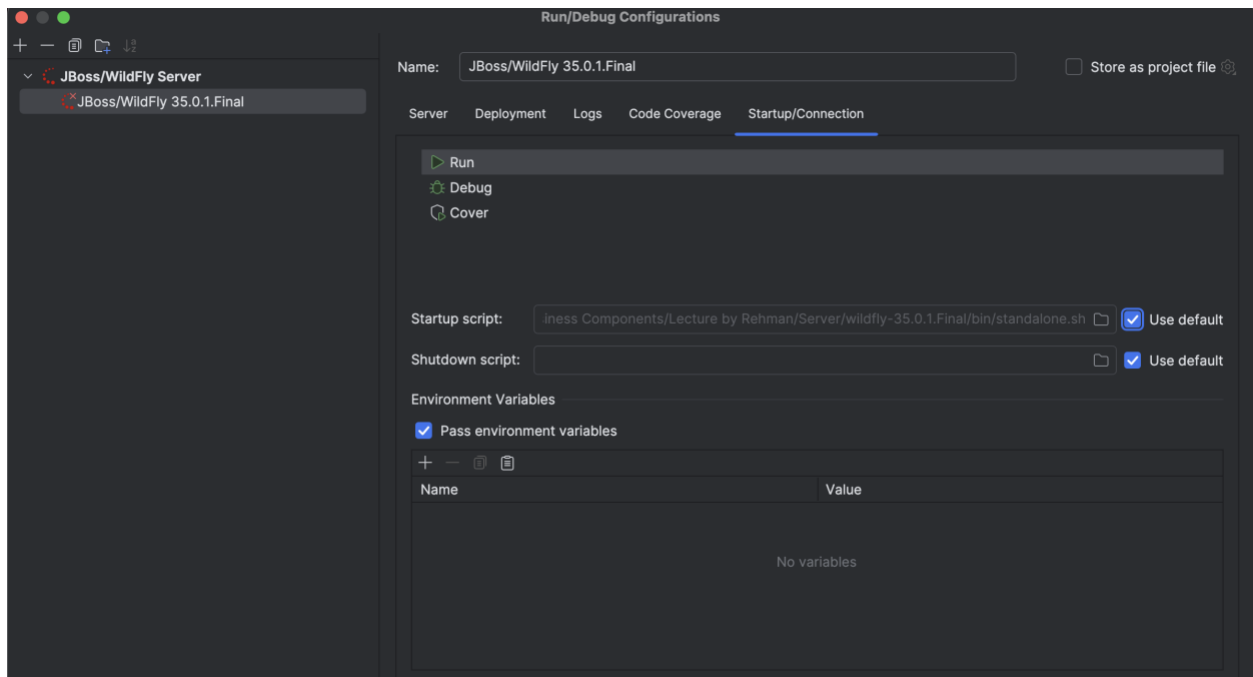
Step 3:

To run the server with full profile, follow these steps:

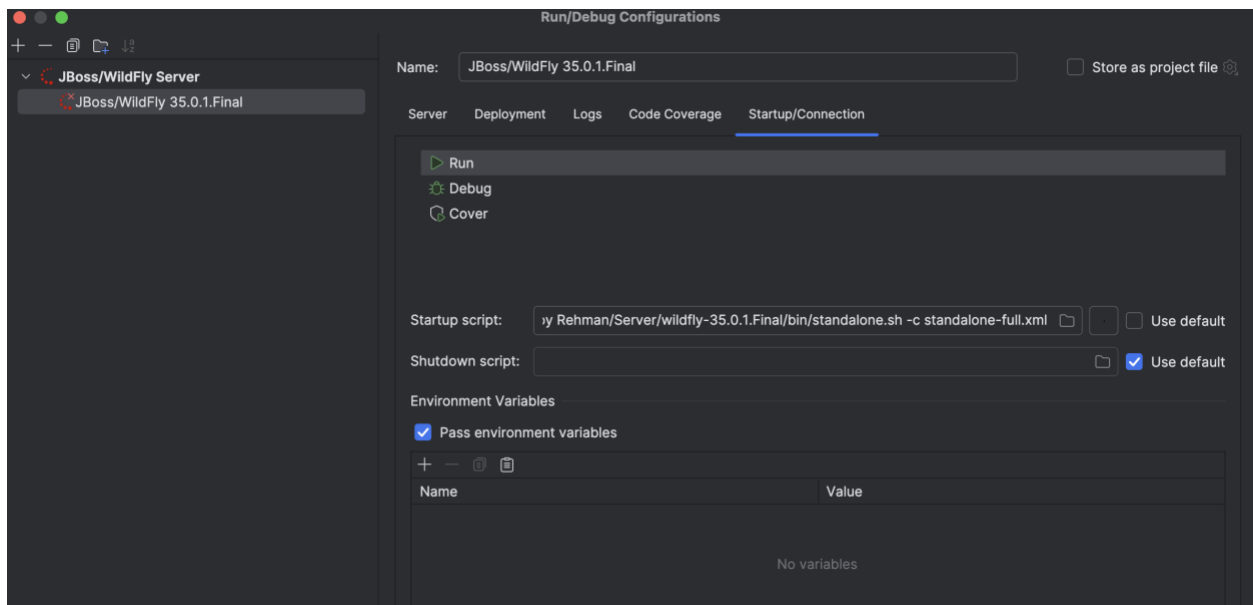
Edit Configuration



Select Startup/Connection tab

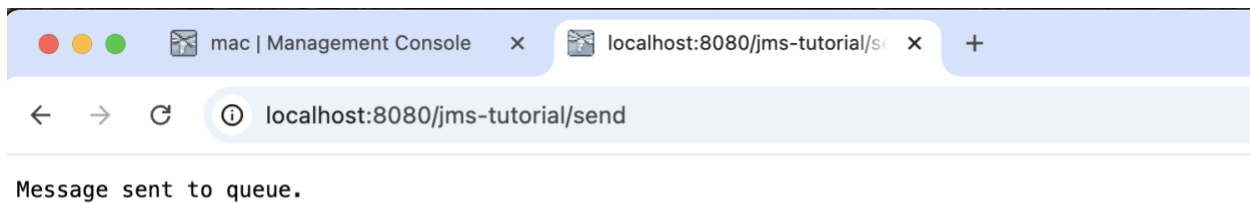


Uncheck startup script and add **-c standalone-full.xml** at the end of the path.

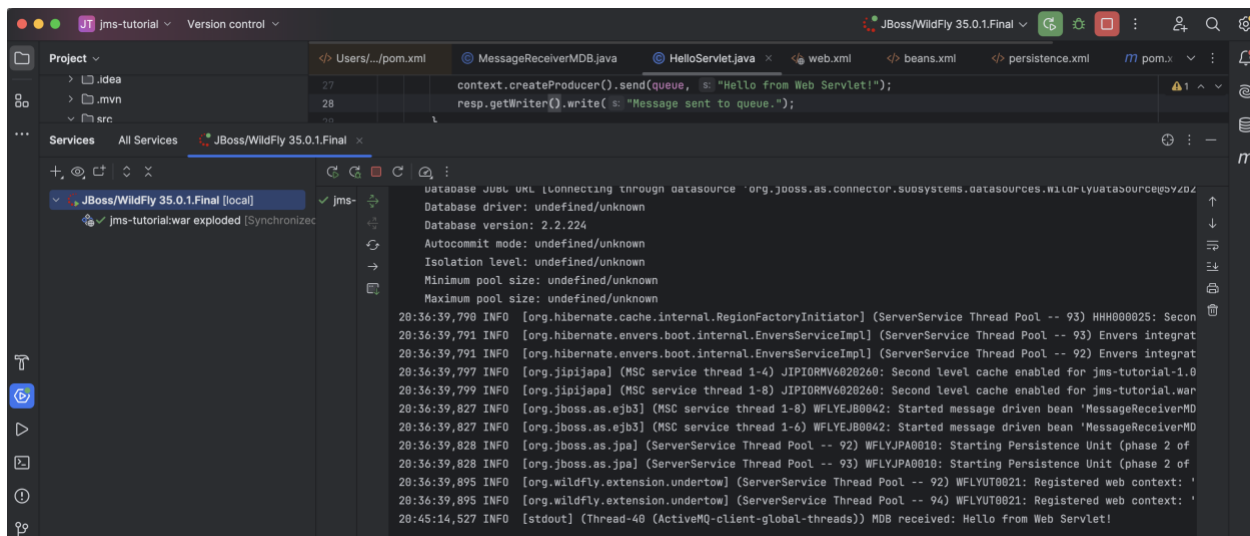


Apply your changes and run the application.

To test the application type in web browser: <http://localhost:8080/jms-tutorial/send>



Producer will send the message and Consumer will receive it. The message sent will appear in the console.



Try it yourself:

- 1) Make a form with an input field.
- 2) Bind that to enter dynamic message.
- 3) Add a label on the same page or add a new page to display the received message.
- 4) Test it with different input.
- 5) Take screenshots of the form, message sent, and message received.
- 6) Upload these screenshots at blackboard in Lab 2.
- 7) Push your code at GitHub.