



CREATING WEB SERVICES WITH INTEGRATED WEB SERVICES (IWS)

TURN YOUR RPG PROGRAMS INTO WEB SERVICES WITH EASE

Presented by Mike Larsen

[Email: Mikelarsen.rbl@gmail.com](mailto:Mikelarsen.rbl@gmail.com)

LinkedIn: www.linkedin.com/in/mike-larsen-a4414a20

AGENDA

- Objectives
- Brief introduction to IWS
- Creating a Web Server
- Lab 1: Creating a very simple Web Service
- Lab 2: Creating a Web Service with Select and Insert capabilities
- Lab 3: Creating a Web Service with Update and Delete capabilities

OBJECTIVES

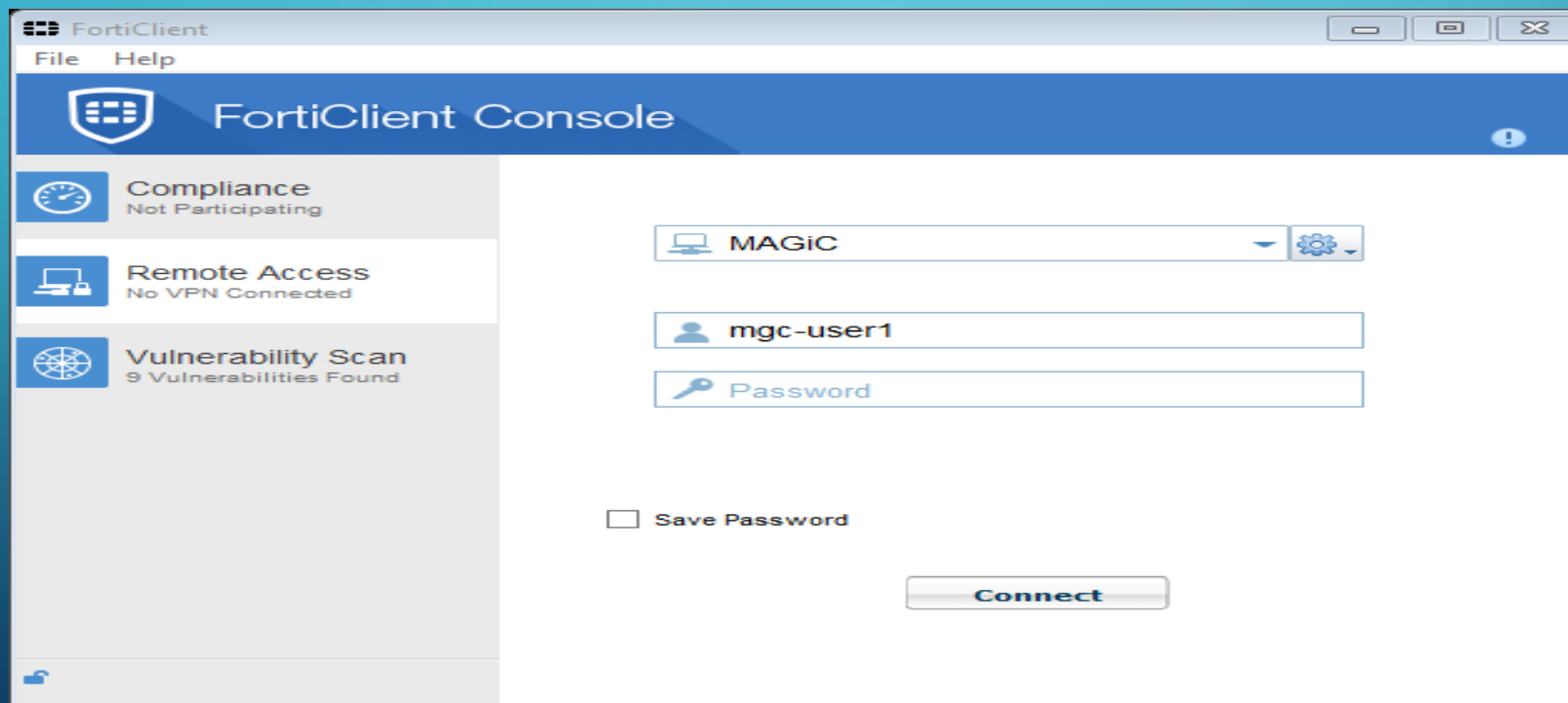
- Learn how to create a Web Server on IBM i
- Learn how to create and deploy RPG programs as Web Services
- Learn how to test Web Services using SoapUi

NOTES

- This session is an introduction to creating Web Services
- All source code used in the session is provided
- RPG code is in fully free format. RDi is recommended, but not required
- SoapUi will be used for testing the Web Services and can be downloaded for free. <https://www.soapui.org/news/soapui-5-3-is-here!.html>

DOWNLOAD FORTICLIENT VPN

- <https://forticlient.com/>



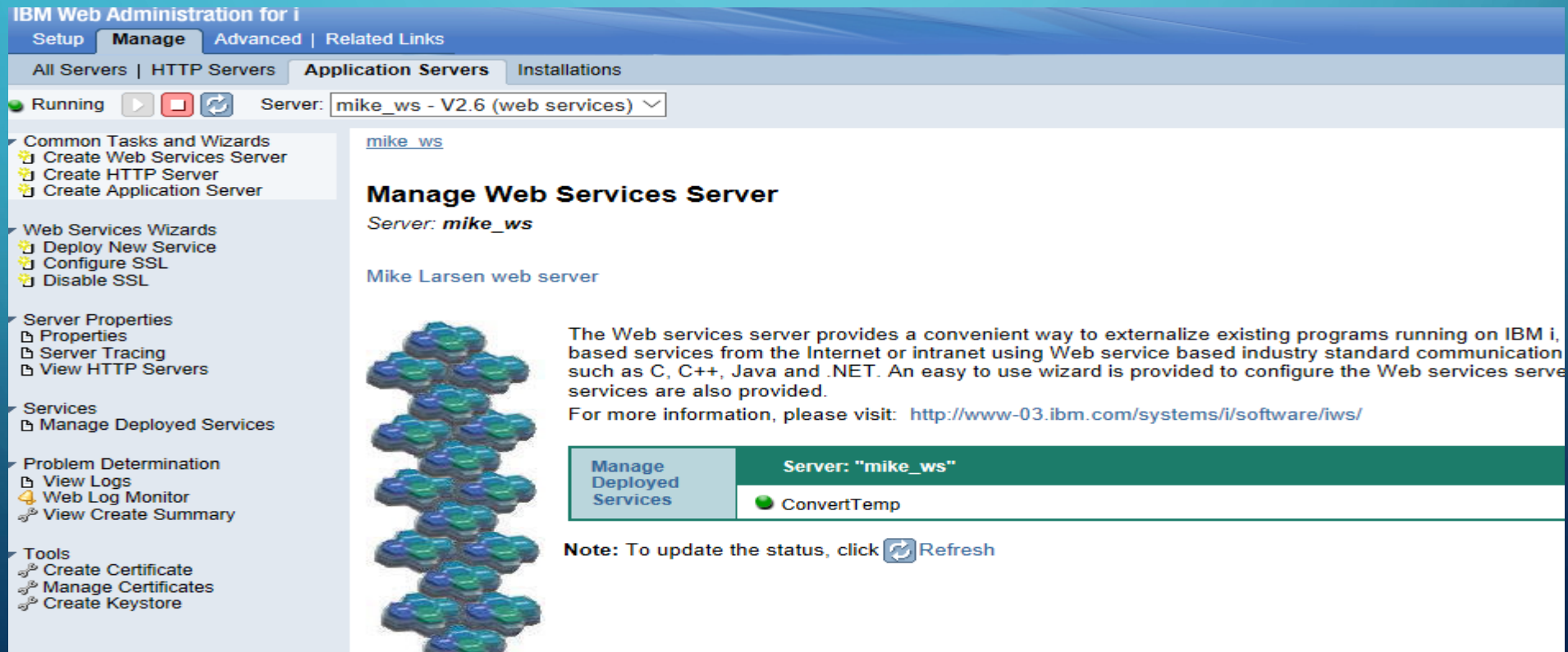
BRIEF INTRODUCTION TO IWS

- Allows ILE programs to be deployed as Web Services
- Wizard based deployment
- Supported on IBM i 6.1 and above
- SOAP or REST services. We'll focus on REST.
- FAQ's/prerequisites –

<https://www-03.ibm.com/systems/power/software/i/iws/faq/>

LOG IN TO IWS

- In a browser, navigate to <http://10.2.5.250:2001/HTTPAdmin>
- Enter your IBM i credentials



The screenshot displays the IBM Web Administration for i interface. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and the 'Server' dropdown menu is set to 'mike_ws - V2.6 (web services)'. The left sidebar contains a tree view with categories: 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', 'Problem Determination', and 'Tools'. The main content area is titled 'Manage Web Services Server' for 'Server: mike_ws'. It features a 'Mike Larsen web server' section with a stack of server icons and a description: 'The Web services server provides a convenient way to externalize existing programs running on IBM i, based services from the Internet or intranet using Web service based industry standard communication such as C, C++, Java and .NET. An easy to use wizard is provided to configure the Web services server services are also provided. For more information, please visit: <http://www-03.ibm.com/systems/i/software/iws/>'. Below the description is a table with a header 'Server: "mike_ws"' and one row containing a green status indicator and the text 'ConvertTemp'. A 'Manage Deployed Services' button is visible on the left. At the bottom, a note states: 'Note: To update the status, click Refresh' with a refresh icon.

CREATING A WEB SERVER – STEP 1

- Click the link ‘Create Web Services Server’

The screenshot displays the IBM Web Administration for i console. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and a dropdown menu shows the selected server: 'mike_ws - V2.6 (web services)'. On the left side, a navigation tree is visible with categories like 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', 'Problem Determination', and 'Tools'. The main content area is titled 'Create Web Services Server' and indicates it is 'Step 1 of 3'. It contains a welcome message and a link for more information. Below this, there is a section titled 'Specify a unique name for this server' with a 'Server name' input field containing 'mike_ws2' and a 'Server description' input field containing 'Web services server created by the Create Web Servic'. At the bottom of the wizard, there are 'Back', 'Next', and 'Cancel' buttons.

IBM Web Administration for i
Setup Manage Advanced | Related Links
All Servers | HTTP Servers Application Servers Installations
Running Server: mike_ws - V2.6 (web services)

Create Web Services Server
Specify Web services server name - Step 1 of 3

Welcome to the Create Web Services Server wizard. A Web services server provides a convenient way to externalize existing programs run based services from the Internet or intranet via Web service based industry standard communication protocols such as SOAP. The clients everything needed to run Web services.
For more information, please visit: <http://www-03.ibm.com/systems/i/software/i/ws/>

Specify a unique name for this server

Server name:

Server description:

Back Next Cancel

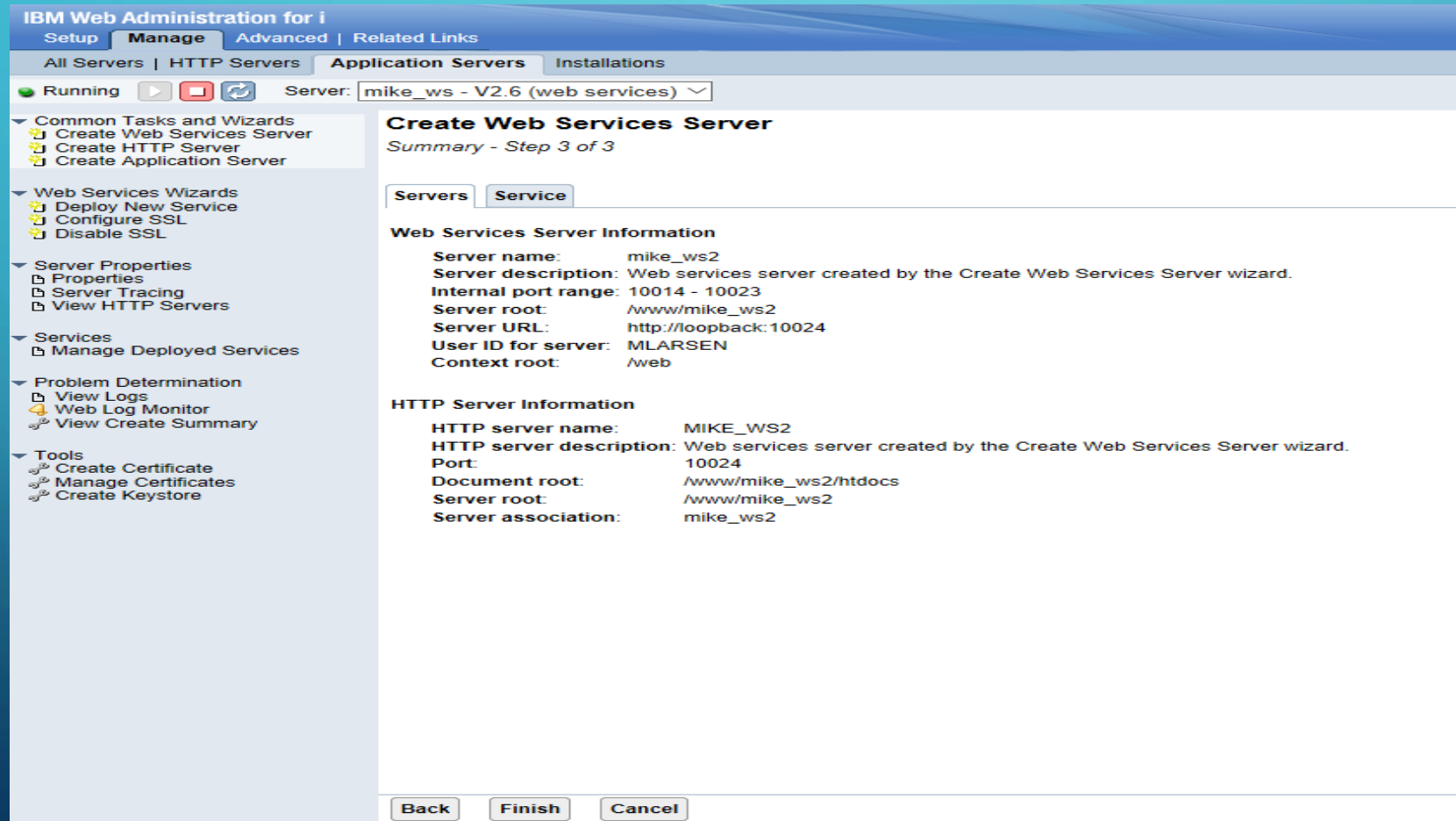
CREATING A WEB SERVER – STEP 2

- Use default user ID

The screenshot displays the IBM Web Administration for i console. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The current server is identified as 'mike_ws - V2.6 (web services)'. The left sidebar contains a tree view with categories: 'Common Tasks and Wizards' (including 'Create Web Services Server'), 'Web Services Wizards' (including 'Deploy New Service'), 'Server Properties', 'Services', 'Problem Determination', and 'Tools'. The main content area is titled 'Create Web Services Server' and is at 'Step 2 of 3', 'Specify User ID for Server'. It contains the text: 'The server requires an IBM i user ID to run the server's jobs. It is recommended that a special user ID is specified to run'. Below this, there is a section 'Specify user ID for this server:' with two radio button options: 'Use default user ID' (which is selected) and 'Specify an existing user ID'. A note states: 'Note: The default server user ID is MLARSEN.' At the bottom of the console, there are three buttons: 'Back', 'Next', and 'Cancel'.

CREATING A WEB SERVER – STEP 3

- Click 'Finish'



The screenshot displays the IBM Web Administration for i console. The main window title is "IBM Web Administration for i". The navigation tabs include "Setup", "Manage", "Advanced", and "Related Links". The current view is "Application Servers" for the server "mike_ws - V2.6 (web services)".

The left sidebar contains a tree view with the following categories:

- Common Tasks and Wizards
 - Create Web Services Server
 - Create HTTP Server
 - Create Application Server
- Web Services Wizards
 - Deploy New Service
 - Configure SSL
 - Disable SSL
- Server Properties
 - Properties
 - Server Tracing
 - View HTTP Servers
- Services
 - Manage Deployed Services
- Problem Determination
 - View Logs
 - Web Log Monitor
 - View Create Summary
- Tools
 - Create Certificate
 - Manage Certificates
 - Create Keystore

The main content area is titled "Create Web Services Server" and shows "Summary - Step 3 of 3". It has two tabs: "Servers" and "Service".

Web Services Server Information

- Server name: mike_ws2
- Server description: Web services server created by the Create Web Services Server wizard.
- Internal port range: 10014 - 10023
- Server root: /www/mike_ws2
- Server URL: http://loopback:10024
- User ID for server: MLARSEN
- Context root: /web

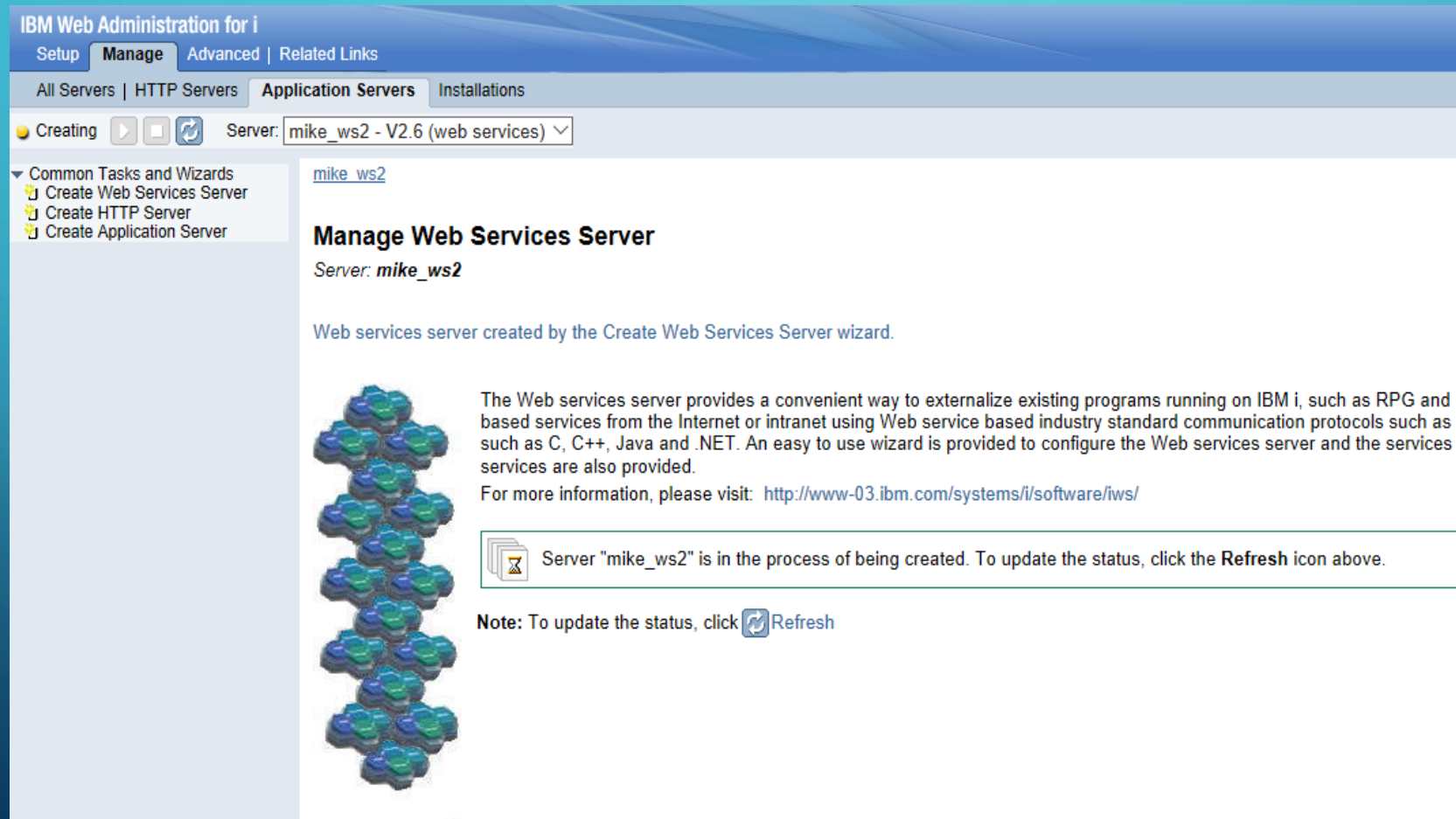
HTTP Server Information

- HTTP server name: MIKE_WS2
- HTTP server description: Web services server created by the Create Web Services Server wizard.
- Port: 10024
- Document root: /www/mike_ws2/htdocs
- Server root: /www/mike_ws2
- Server association: mike_ws2

At the bottom of the window, there are three buttons: "Back", "Finish", and "Cancel".

CREATING A WEB SERVER

- Click 'Refresh'



The screenshot displays the IBM Web Administration for i interface. At the top, there are tabs for 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below these are navigation tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and a dropdown menu shows 'Server: mike_ws2 - V2.6 (web services)'. On the left, a sidebar lists 'Common Tasks and Wizards' including 'Create Web Services Server', 'Create HTTP Server', and 'Create Application Server'. The main content area is titled 'Manage Web Services Server' for server 'mike_ws2'. It includes a description of the Web services server and a note that the server is in the process of being created, with a 'Refresh' icon to update the status.

IBM Web Administration for i

Setup **Manage** Advanced | Related Links

All Servers | HTTP Servers **Application Servers** Installations

Creating [Refresh] [Stop] [Play] Server: mike_ws2 - V2.6 (web services) v

Common Tasks and Wizards

- Create Web Services Server
- Create HTTP Server
- Create Application Server

[mike_ws2](#)

Manage Web Services Server


Server: *mike_ws2*

Web services server created by the Create Web Services Server wizard.

The Web services server provides a convenient way to externalize existing programs running on IBM i, such as RPG and C based services from the Internet or intranet using Web service based industry standard communication protocols such as SOAP, REST, etc. An easy to use wizard is provided to configure the Web services server and the services for which services are also provided.

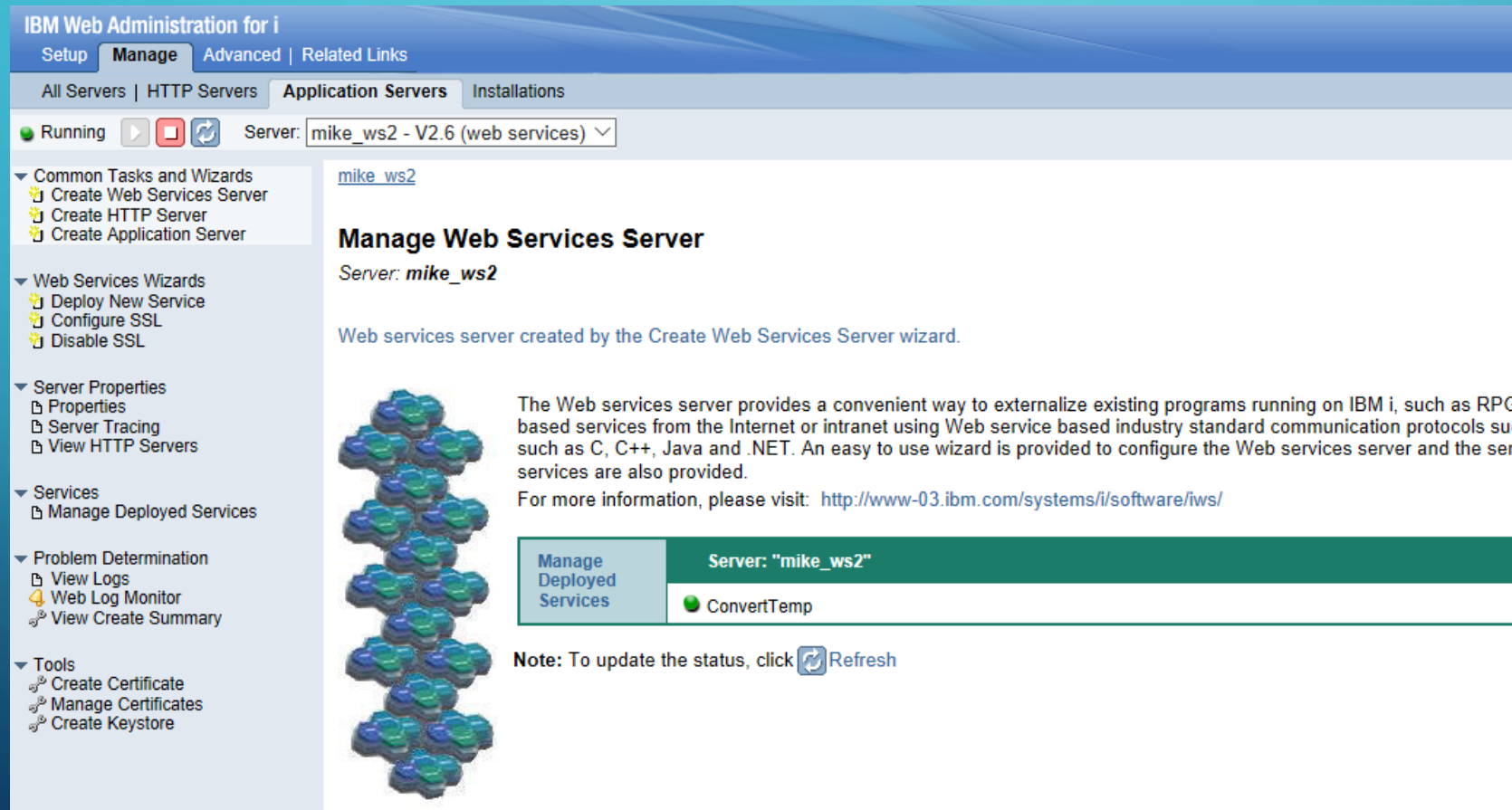
For more information, please visit: <http://www-03.ibm.com/systems/i/software/iws/>

Server "mike_ws2" is in the process of being created. To update the status, click the **Refresh** icon above.

Note: To update the status, click  Refresh

CREATING A WEB SERVER

- Web server is created with a default service 'ConvertTemp'



The screenshot displays the IBM Web Administration for i console. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and the server 'mike_ws2 - V2.6 (web services)' is selected. The left sidebar contains a tree view with categories like 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', 'Problem Determination', and 'Tools'. The main content area is titled 'Manage Web Services Server' for 'Server: mike_ws2'. It includes a description of the web services server, a link to external information, and a table showing the status of the 'ConvertTemp' service.

IBM Web Administration for i
Setup Manage Advanced Related Links
All Servers | HTTP Servers Application Servers Installations
Running Server: mike_ws2 - V2.6 (web services) ▼

Common Tasks and Wizards
Create Web Services Server
Create HTTP Server
Create Application Server

Web Services Wizards
Deploy New Service
Configure SSL
Disable SSL

Server Properties
Properties
Server Tracing
View HTTP Servers

Services
Manage Deployed Services

Problem Determination
View Logs
Web Log Monitor
View Create Summary

Tools
Create Certificate
Manage Certificates
Create Keystore

mike_ws2

Manage Web Services Server

Server: *mike_ws2*

Web services server created by the Create Web Services Server wizard.

The Web services server provides a convenient way to externalize existing programs running on IBM i, such as RPG based services from the Internet or intranet using Web service based industry standard communication protocols such as C, C++, Java and .NET. An easy to use wizard is provided to configure the Web services server and the services are also provided.

For more information, please visit: <http://www-03.ibm.com/systems/i/software/iws/>

Manage Deployed Services	Server: "mike_ws2"
	● ConvertTemp

Note: To update the status, click Refresh

CREATING A WEB SERVICE

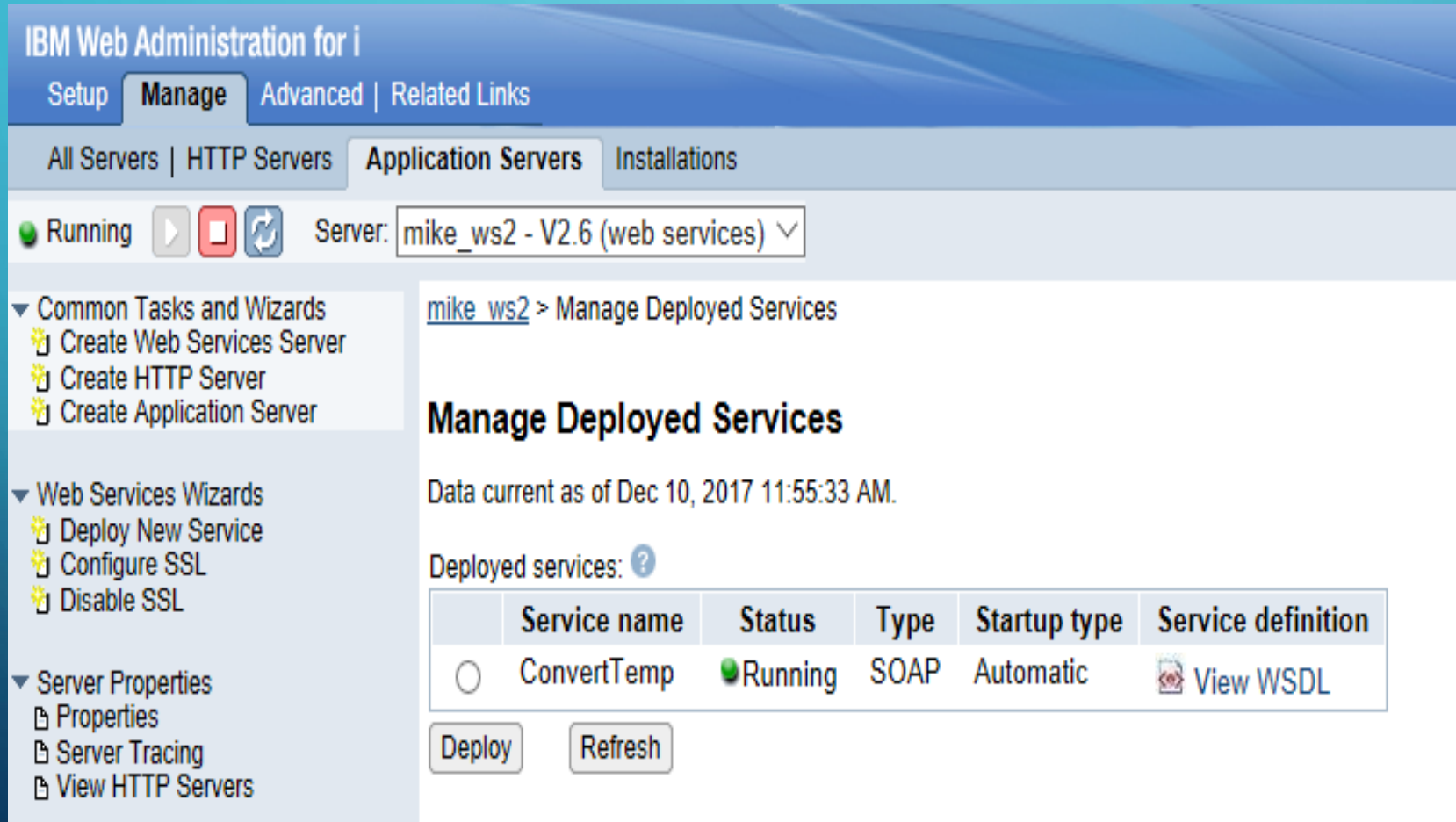
- Copy RPG & SQL source members to your personal library, Create source physical files, if necessary.
- RPG source is in QRPGLSRC/MLARSEN
 - REST_1_ML
 - REST_2_ML
 - REST_3_ML
 - WRITE_LOG
- SQL source is in QDDL SRC/MLARSEN
 - RESTLOG
 - SALES_HIST
- RPG copy source is in QRPGCOPYI/MLARSEN
 - REST_DEMO

CREATING A WEB SERVICE

- Compile RPG & SQL source members to your personal library
- Use the cheat sheet to compile the SQL tables
- REST_1_ML – CRTBNDRPG & WRITE_LOG. Create as a program.
- REST_2_ML & REST_3_ML – CRTSQLRPGI. Create as modules, then service programs.
- REST_DEMO – Don't compile.

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Click 'Deploy New Service' or 'Deploy' button.






The screenshot displays the IBM Web Administration for i interface. The top navigation bar includes 'Setup', 'Manage' (selected), 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers' (selected), and 'Installations'. A status bar shows 'Running' with control icons and a dropdown menu for the server 'mike_ws2 - V2.6 (web services)'. The left sidebar contains a tree view with categories: 'Common Tasks and Wizards' (Create Web Services Server, Create HTTP Server, Create Application Server), 'Web Services Wizards' (Deploy New Service, Configure SSL, Disable SSL), and 'Server Properties' (Properties, Server Tracing, View HTTP Servers). The main content area is titled 'mike_ws2 > Manage Deployed Services' and shows 'Data current as of Dec 10, 2017 11:55:33 AM.' Below this, a table lists 'Deployed services:'. The table has columns for Service name, Status, Type, Startup type, and Service definition. One service, 'ConvertTemp', is listed with a status of 'Running', Type of 'SOAP', and Startup type of 'Automatic'. A 'View WSDL' link is provided for this service. At the bottom of the table area, there are 'Deploy' and 'Refresh' buttons.

IBM Web Administration for i

Setup **Manage** Advanced | Related Links

All Servers | HTTP Servers **Application Servers** Installations

Running    Server: mike_ws2 - V2.6 (web services) ▼

▼ Common Tasks and Wizards

- ✚ Create Web Services Server
- ✚ Create HTTP Server
- ✚ Create Application Server

▼ Web Services Wizards

- ✚ Deploy New Service
- ✚ Configure SSL
- ✚ Disable SSL


▼ Server Properties



- ▢ Properties
- ▢ Server Tracing
- ▢ View HTTP Servers

mike_ws2 > Manage Deployed Services

Manage Deployed Services

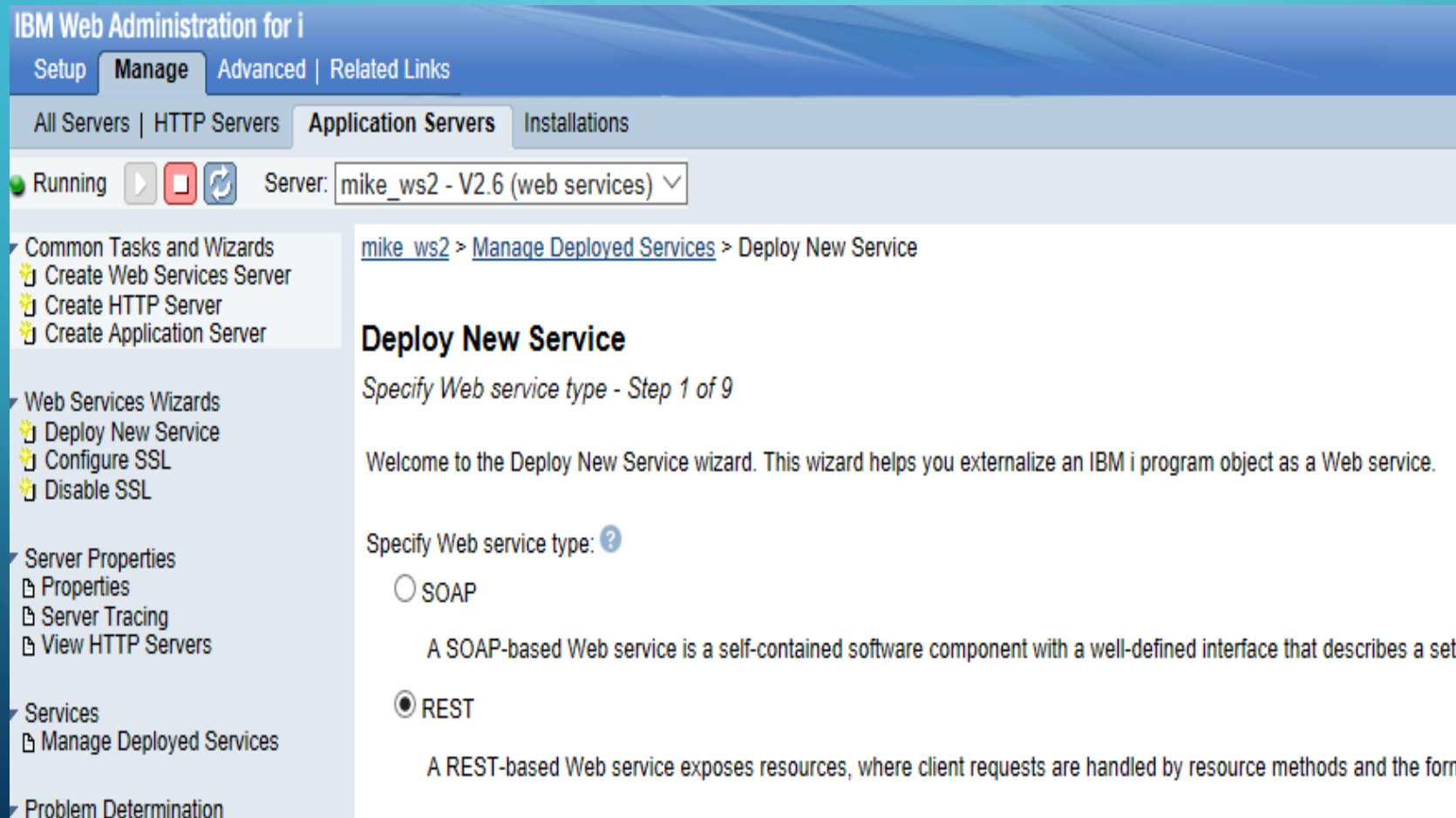
Data current as of Dec 10, 2017 11:55:33 AM.

Deployed services: 

	Service name	Status	Type	Startup type	Service definition
<input type="radio"/>	ConvertTemp	 Running	SOAP	Automatic	 View WSDL

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Select 'REST', then click 'Next'.



The screenshot displays the IBM Web Administration console interface. At the top, there are navigation tabs for 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below these, there are sub-tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and a dropdown menu shows the selected server: 'mike_ws2 - V2.6 (web services)'. The left sidebar contains a tree view with categories like 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', and 'Problem Determination'. The 'Web Services Wizards' category is expanded, showing 'Deploy New Service' as the selected option. The main content area shows the 'Deploy New Service' wizard, titled 'Specify Web service type - Step 1 of 9'. The wizard text reads: 'Welcome to the Deploy New Service wizard. This wizard helps you externalize an IBM i program object as a Web service.' Below this, there is a section 'Specify Web service type: ?' with two radio button options: 'SOAP' and 'REST'. The 'REST' option is selected. A partial description for 'SOAP' is visible: 'A SOAP-based Web service is a self-contained software component with a well-defined interface that describes a set of...'. A partial description for 'REST' is also visible: 'A REST-based Web service exposes resources, where client requests are handled by resource methods and the form...

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Populate library name and ILE Object name. Select *PGM, then click 'Next'.

The screenshot shows the IBM Web Administration for i console. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The current server is 'mike_ws2 - V2.6 (web services)'. The left sidebar contains a tree view with categories like 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', 'Problem Determination', and 'Tools'. The main content area displays the 'Deploy New Service' wizard, which is currently on 'Step 2 of 9: Specify Location of IBM i Program Object'. The wizard text states: 'The IBM i object to be externalized as a Web service must be an existing ILE program (*PGM) or service program (*SRVPGM) located on the system.' Under the heading 'Specify the program object for the Web service.', there are two radio button options: 'Specify IBM i library and ILE program object name (Recommended)' (which is selected) and 'Browse the integrated file system for the IBM i program object'. The 'Recommended' option includes input fields for 'Library name' (containing 'mlarsen') and 'ILE Object name' (containing 'Rest_1_ml'), and a radio button for 'ILE Object type' set to '*PGM'.

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Populate resource name and service description, then click ‘Next’.

The screenshot shows the IBM Web Administration console interface. At the top, there are tabs for 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below these are navigation tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and a dropdown menu shows 'Server: mike_ws2 - V2.6 (web services)'. On the left, a sidebar contains a tree view with categories: 'Common Tasks and Wizards' (Create Web Services Server, Create HTTP Server, Create Application Server), 'Web Services Wizards' (Deploy New Service, Configure SSL, Disable SSL), 'Server Properties' (Properties, Server Tracing, View HTTP Servers), and 'Services' (Manage Deployed Services). The main content area shows the breadcrumb 'mike_ws2 > Manage Deployed Services > Deploy New Service' and the title 'Deploy New Service'. Below the title is the subtitle 'Specify Name for Service - Step 3 of 9'. A paragraph explains: 'The Web service to be externalized is a resource. The URI path template identifies matching patterns for incoming HTTP further restrict what is allowed.' Below this are three input fields: 'Resource name:' with the value 'REST_1_ML', 'Service description:' with the value 'Rest lab 1', and 'URI path template:' with the value '/'. To the right of the URI path template field is an example: 'e.g. /temperature, /temperature/{temp:id+'.

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Change 'ParmInMessage' to input, then click 'Next'.

The screenshot shows the IBM Web Administration console interface. The main content area displays the 'Deploy New Service' wizard, which is part of the 'Manage Deployed Services' task. The wizard is currently on 'Step 4 of 9', titled 'Select Export Procedures to Externalize as a Web Service'. The interface includes a left-hand navigation pane with various tasks and a main content area with a table of exportable procedures.

Deploy New Service
Select Export Procedures to Externalize as a Web Service - Step 4 of 9

Exported procedures are entry points to a program object and are mapped to Web service operations. A procedure is a more procedures. A program contains only one procedure.

The table below lists all the exported procedures found in the program object that can be externalized through this Web by clients and what is returned by the Web service.

Detect length fields

Use parameter name as element name for data structures

Export procedures: [?](#)

Select	Procedure name/Parameter name	Usage	Data type
<input checked="" type="checkbox"/>	Rest_1_ml		
	ParmlnMessage	input	char
	ParmOutMessage	output	struct

Select All Deselect All Expand All Collapse All

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Use the cheat sheet to populate the URI path.

The screenshot shows the IBM Web Administration for i console. The main content area is titled "Deploy New Service" and is part of a wizard with 9 steps, currently on step 5. The breadcrumb path is "mike_ws2 > Manage Deployed Services > Deploy New Service".

Deploy New Service
Specify Resource Method Information - Step 5 of 9

Procedures are mapped to resource methods. Each resource method needs to be defined to handle client requests by mapping an HTTP request method

Specify resource method information. ?

Procedure name: Rest_1_ml
URI path template for resource: /
HTTP request method: GET
URI path template for method: /{{ParmInMessage:\w+}} or...
HTTP response code output parameter: *NONE
HTTP header array output parameter: *NONE
Allowed input media types: *ALL or...
Returned output media types: *JSON or...
Whether to wrap input parameters:
 Wrap input parameters
 Do not wrap input parameters

Input parameter mappings:

Parameter name	Data type	Input source	Identifier	Default Value
ParmInMessage	char	*PATH_PARAM	ParmInMessage	*NONE or...

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Click 'Next'.

The screenshot shows the IBM Web Administration for i console. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and a dropdown menu shows 'Server: mike_ws2 - V2.6 (web services)'. The main content area displays the 'Deploy New Service' wizard, which is currently at 'Step 6 of 9'. The wizard title is 'Deploy New Service' and the subtitle is 'Specify User ID for this Service - Step 6 of 9'. The main text reads: 'The service requires an IBM i user ID to run the program object that contains the Web service business logic.' Below this, there are two radio button options: 'Use server's user ID' (which is selected) and 'Specify an existing user ID'. A help icon (?) is visible next to the 'Specify User ID for this Service:' label. The left sidebar contains a navigation tree with categories like 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', and 'Services'.

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Click 'Next'.

The screenshot displays the IBM Web Administration for i interface. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and the server 'mike_ws2 - V2.6 (web services)' is selected. The left sidebar contains a tree view with categories: 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', and 'Problem Determination'. The 'Web Services Wizards' category is expanded, showing 'Deploy New Service' as the selected option. The main content area shows the 'Deploy New Service' wizard, specifically 'Step 7 of 9: Specify Library List'. The text reads: 'The functionality of the IBM i program you want to externalize as a Web service may depend upon other programs used.' Below this, there are two radio button options for specifying the library list position: 'Insert libraries in front of user library portion of the library list' (unselected) and 'Insert libraries at the end of user library portion of the library list' (selected). A 'Library list entries' section contains a table with one entry: 'MLARSEN'. There are 'Add' and 'Remove All' buttons below the table.

IBM Web Administration for i

Setup **Manage** Advanced | Related Links

All Servers | HTTP Servers **Application Servers** Installations

Running Server: mike_ws2 - V2.6 (web services) ▼

▼ Common Tasks and Wizards

- ✚ Create Web Services Server
- ✚ Create HTTP Server
- ✚ Create Application Server

▼ Web Services Wizards

- ✚ **Deploy New Service**
- ✚ Configure SSL
- ✚ Disable SSL

▼ Server Properties

- ▢ Properties
- ▢ Server Tracing
- ▢ View HTTP Servers

▼ Services

- ▢ Manage Deployed Services

▼ Problem Determination

- ▢ View Logs
- 🔍 Web Log Monitor
- 📄 View Create Summary

mike_ws2 > Manage Deployed Services > Deploy New Service

Deploy New Service

Specify Library List - Step 7 of 9

The functionality of the IBM i program you want to externalize as a Web service may depend upon other programs used.

Specify library list position for this Web service:

- Insert libraries in front of user library portion of the library list
- Insert libraries at the end of user library portion of the library list

Library list entries: ?

Library name
<input type="radio"/> MLARSEN

Add Remove All

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Select all transport metadata, and an HTTP header, then click 'Next'.

IBM Web Administration for i
Setup **Manage** Advanced | Related Links

All Servers | HTTP Servers **Application Servers** Installations

Running [Play] [Stop] [Refresh] Server: mike_ws2 - V2.6 (web services) ▾

Common Tasks and Wizards
Create Web Services Server
Create HTTP Server
Create Application Server

Web Services Wizards
Deploy New Service
Configure SSL
Disable SSL

Server Properties
Properties
Server Tracing
View HTTP Servers

Services
Manage Deployed Services

Problem Determination
View Logs
Web Log Monitor
View Create Summary

Tools
Create Certificate
Manage Certificates
Create Keystore

mike_ws2 > Manage Deployed Services > Deploy New Service

Deploy New Service

Specify Transport Information to Be Passed - Step 8 of 9

Specify transport information to be passed to the web service implementation code.

Information to be passed to web service implementation code ?

Specify Transport Metadata:

Transport Metadata	
<input checked="" type="checkbox"/>	QUERY_STRING
<input checked="" type="checkbox"/>	REMOTE_ADDR
<input checked="" type="checkbox"/>	REMOTE_USER
<input checked="" type="checkbox"/>	REQUEST_METHOD
<input checked="" type="checkbox"/>	REQUEST_URI
<input checked="" type="checkbox"/>	REQUEST_URL
<input checked="" type="checkbox"/>	SERVER_NAME
<input checked="" type="checkbox"/>	SERVER_PORT

Specify HTTP Headers:

HTTP Headers	
<input checked="" type="radio"/>	user-id

Add Remove Remove All Continue

LAB 1 – CREATE A SIMPLE WEB SERVICE

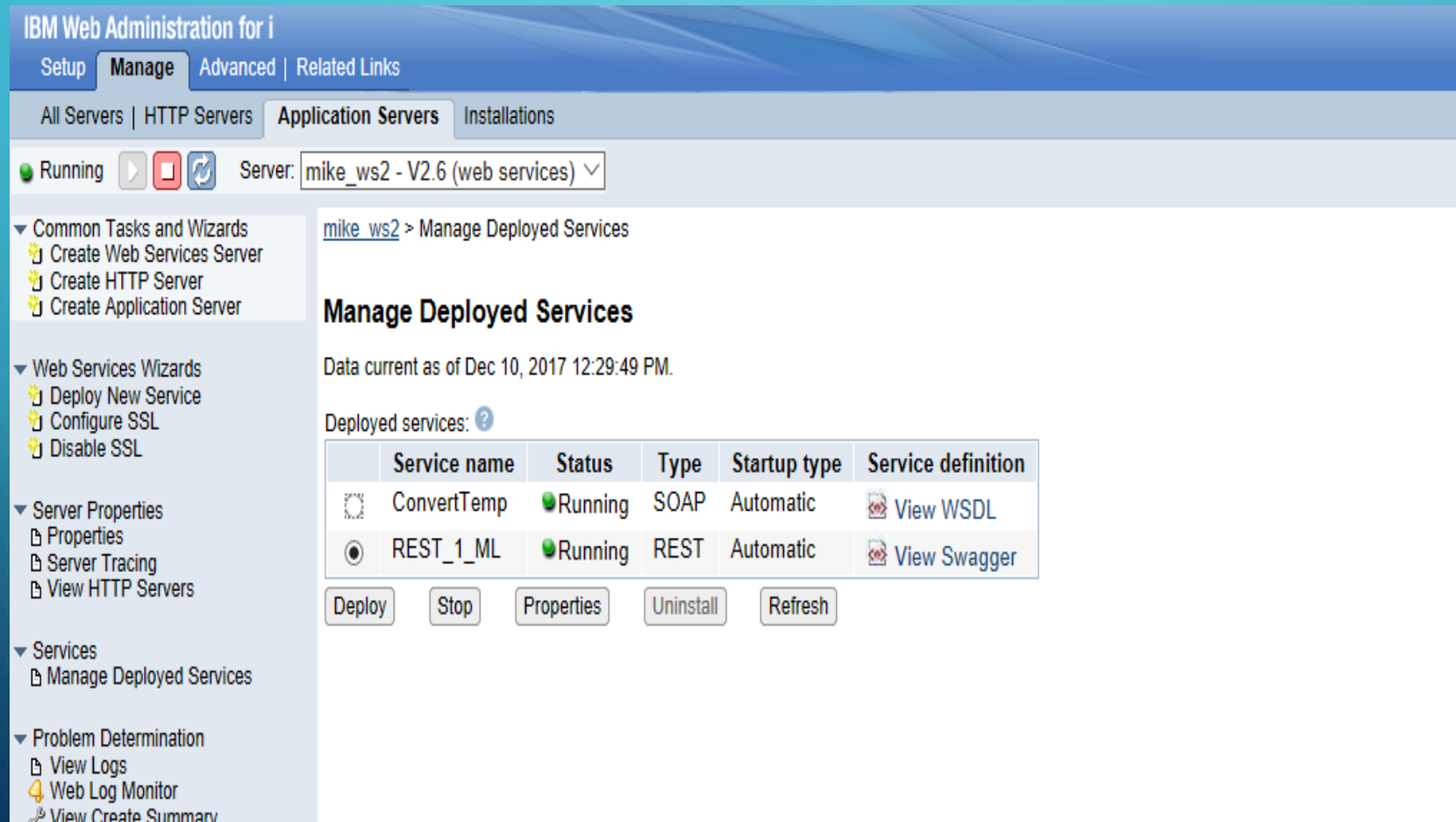
- Click 'Finish'.

The screenshot shows the IBM Web Administration for i console. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and the server 'mike_ws2 - V2.6 (web services)' is selected. The left sidebar contains a tree view with categories like 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', 'Problem Determination', and 'Tools'. The main content area displays the 'Deploy New Service' wizard, which is at 'Step 9 of 9'. The wizard title is 'Deploy New Service' and the summary is 'Summary - Step 9 of 9'. A message states: 'When you click Finish the web service is deployed.' Below this, there are three tabs: 'Service', 'Methods', and 'Request Information'. The 'Service' tab is active, showing the following details:

- Resource name: REST_1_ML
- Resource description: Rest lab 1
- Service install path: /www/mike_ws2/webservices/services/REST_1_ML
- URI path template: /
- User ID for service: *SERVER (MLARSEN)
- Program: /QSYS.LIB/MLARSEN.LIB/REST_1_ML.PGM
- Library list for service: MLARSEN

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Note: you may need to click the 'refresh' button.






The screenshot displays the IBM Web Administration for i console. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The current server is identified as 'mike_ws2 - V2.6 (web services)'. The left sidebar contains a tree view with categories like 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', and 'Problem Determination'. The main content area is titled 'mike_ws2 > Manage Deployed Services' and shows a table of deployed services. The table has columns for 'Service name', 'Status', 'Type', 'Startup type', and 'Service definition'. Two services are listed: 'ConvertTemp' (SOAP, Automatic) and 'REST_1_ML' (REST, Automatic). Below the table are buttons for 'Deploy', 'Stop', 'Properties', 'Uninstall', and 'Refresh'.

IBM Web Administration for i

Setup **Manage** Advanced | Related Links

All Servers | HTTP Servers **Application Servers** Installations

Running    Server: mike_ws2 - V2.6 (web services) ▼

▼ Common Tasks and Wizards

- ▶ Create Web Services Server
- ▶ Create HTTP Server
- ▶ Create Application Server

▼ Web Services Wizards

- ▶ Deploy New Service
- ▶ Configure SSL
- ▶ Disable SSL

▼ Server Properties

- ▶ Properties
- ▶ Server Tracing
- ▶ View HTTP Servers

▼ Services

- ▶ Manage Deployed Services


▼ Problem Determination






- ▶ View Logs
- ▶ Web Log Monitor
- ▶ View Create Summary






mike_ws2 > Manage Deployed Services

Manage Deployed Services

Data current as of Dec 10, 2017 12:29:49 PM.

Deployed services: 

	Service name	Status	Type	Startup type	Service definition
	ConvertTemp	 Running	SOAP	Automatic	 View WSDL
<input checked="" type="radio"/>	REST_1_ML	 Running	REST	Automatic	 View Swagger

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Click on 'Properties' to review the Web Service.

10.2.5.250:2001/HTTPAdmin

IBM Web Administration for i

Setup **Manage** Advanced | Related Links

All Servers | HTTP Servers **Application Servers** Installations

Running Server: mike_ws2 - V2.6 (web services) ▾

Common Tasks and Wizards

- Create Web Services Server
- Create HTTP Server
- Create Application Server

Web Services Wizards

- Deploy New Service
- Configure SSL
- Disable SSL

Server Properties

- Properties
- Server Tracing
- View HTTP Servers

Services

- Manage Deployed Services

mike_ws2 > Manage Deployed Services

Manage Deployed Services

Data current as of Dec 10, 2017 1:16:46 PM.

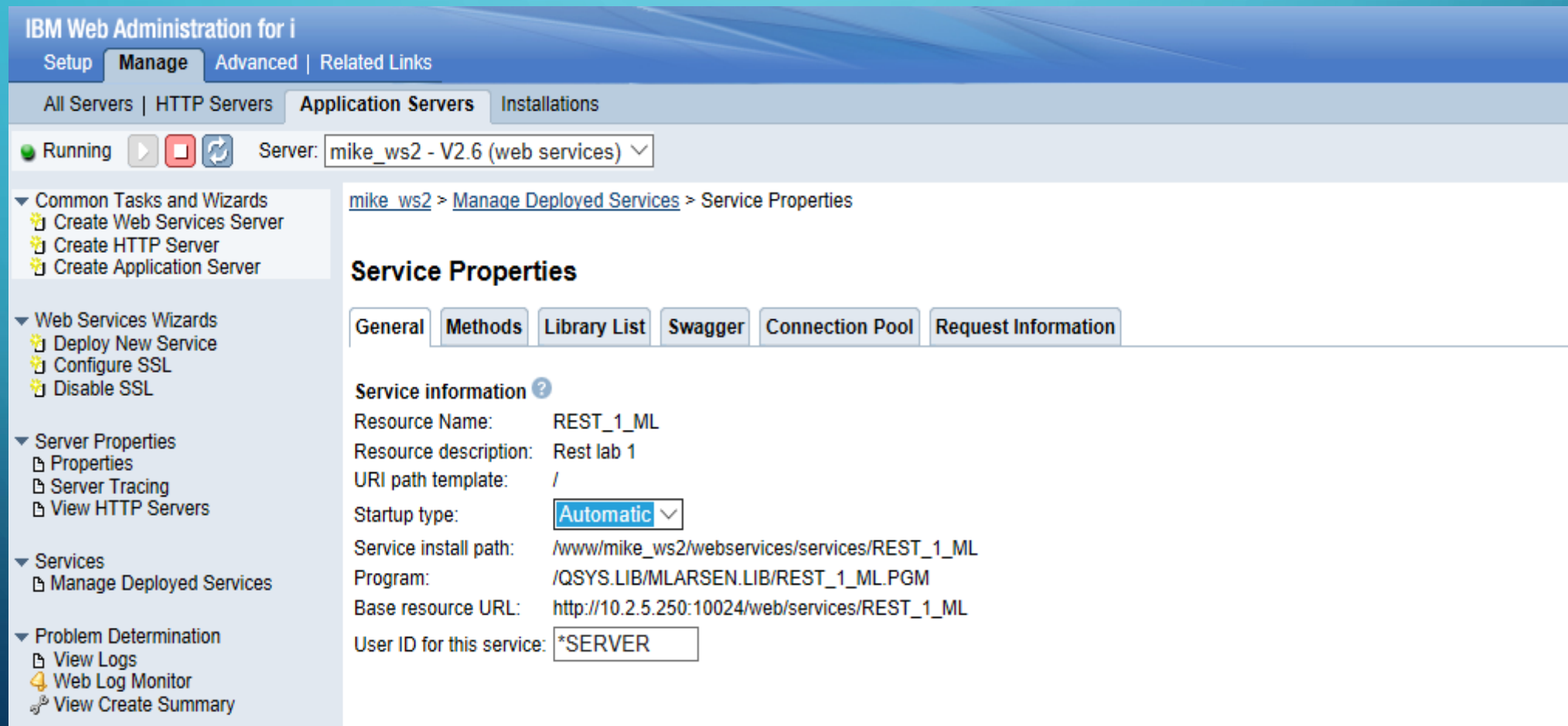
Deployed services: ?

	Service name	Status	Type	Startup type	Service definition
<input type="checkbox"/>	ConvertTemp	Running	SOAP	Automatic	View WSDL
<input checked="" type="radio"/>	REST_1_ML	Running	REST	Automatic	View Swagger

Deploy Stop Properties Uninstall Refresh

LAB 1 – CREATE A SIMPLE WEB SERVICE

- Make note of the of base resource URL.

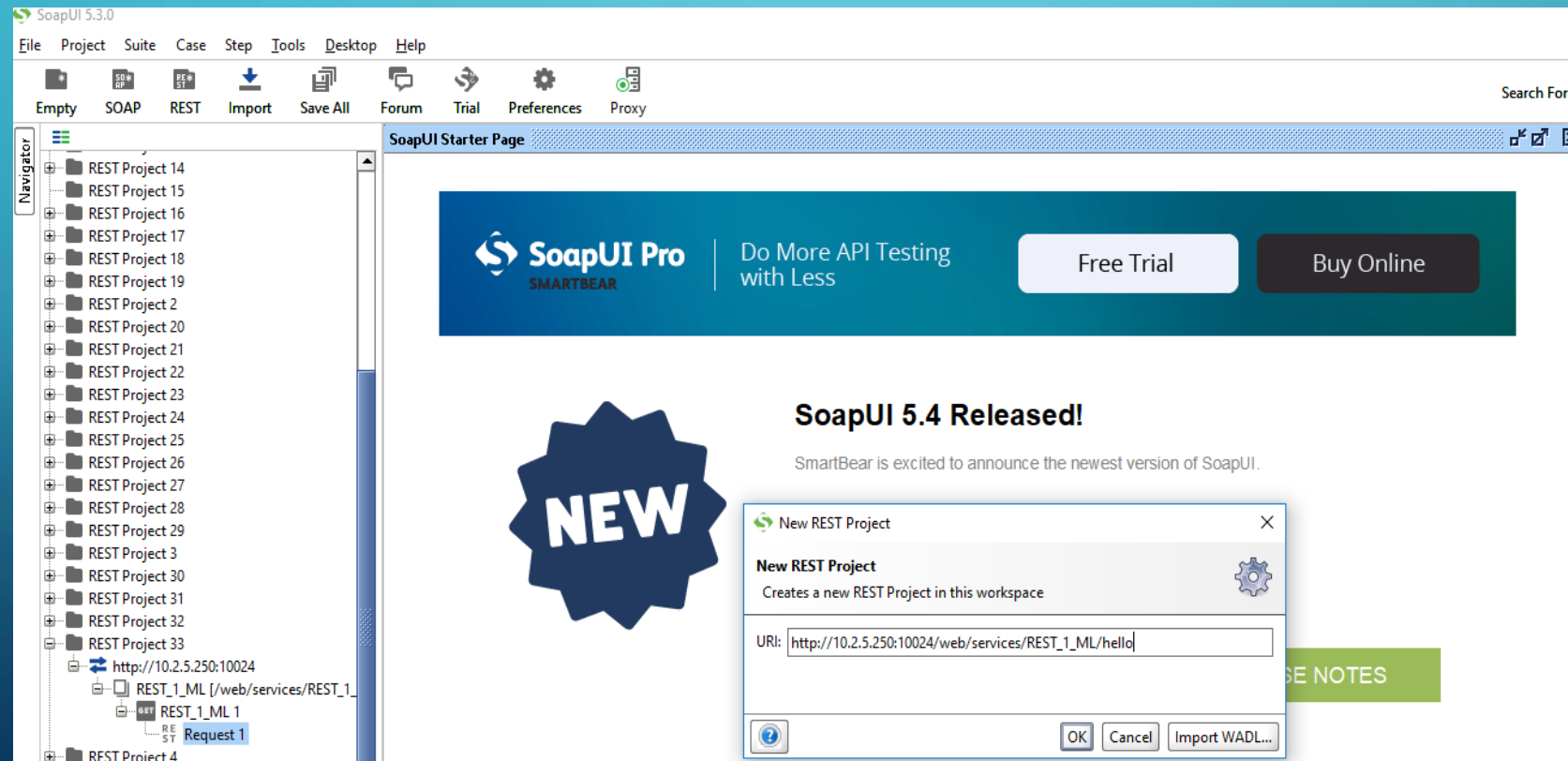


The screenshot displays the IBM Web Administration for i console. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The 'Application Servers' tab is active, and the server 'mike_ws2 - V2.6 (web services)' is selected. The left sidebar contains a tree view with categories like 'Common Tasks and Wizards', 'Web Services Wizards', 'Server Properties', 'Services', and 'Problem Determination'. The main content area shows the 'Service Properties' for 'REST_1_ML' under the 'Manage Deployed Services' path. The 'General' tab is selected, showing the following details:

Service information ?	
Resource Name:	REST_1_ML
Resource description:	Rest lab 1
URI path template:	/
Startup type:	Automatic
Service install path:	/www/mike_ws2/webservices/services/REST_1_ML
Program:	/QSYS.LIB/MLARSEN.LIB/REST_1_ML.PGM
Base resource URL:	http://10.2.5.250:10024/web/services/REST_1_ML
User ID for this service:	*SERVER

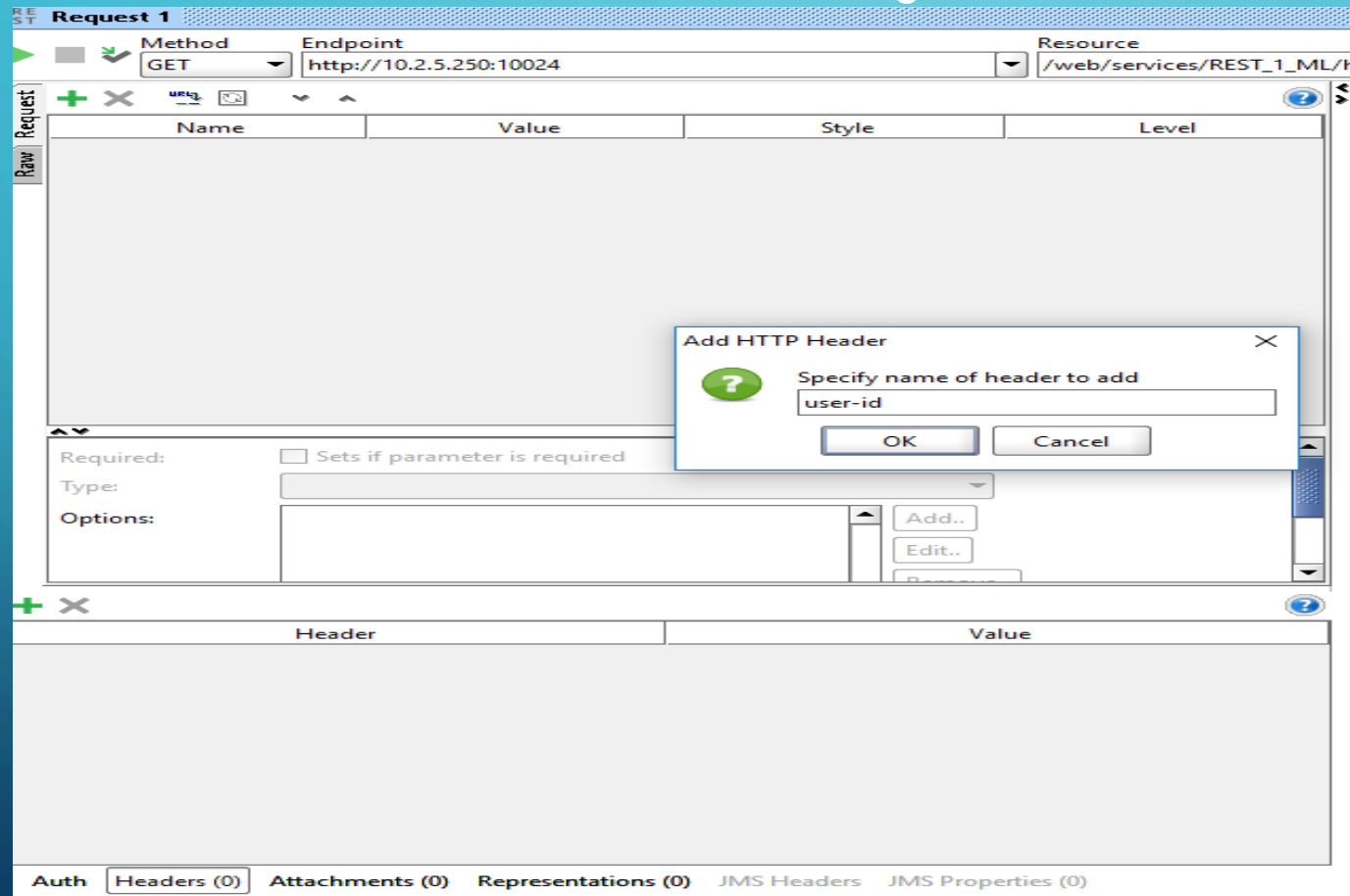
LAB 1 – TESTING THE WEB SERVICE

- Launch SoapUi and create a new REST project. The URI is the base resource URL you just noted, plus the parameter being sent.

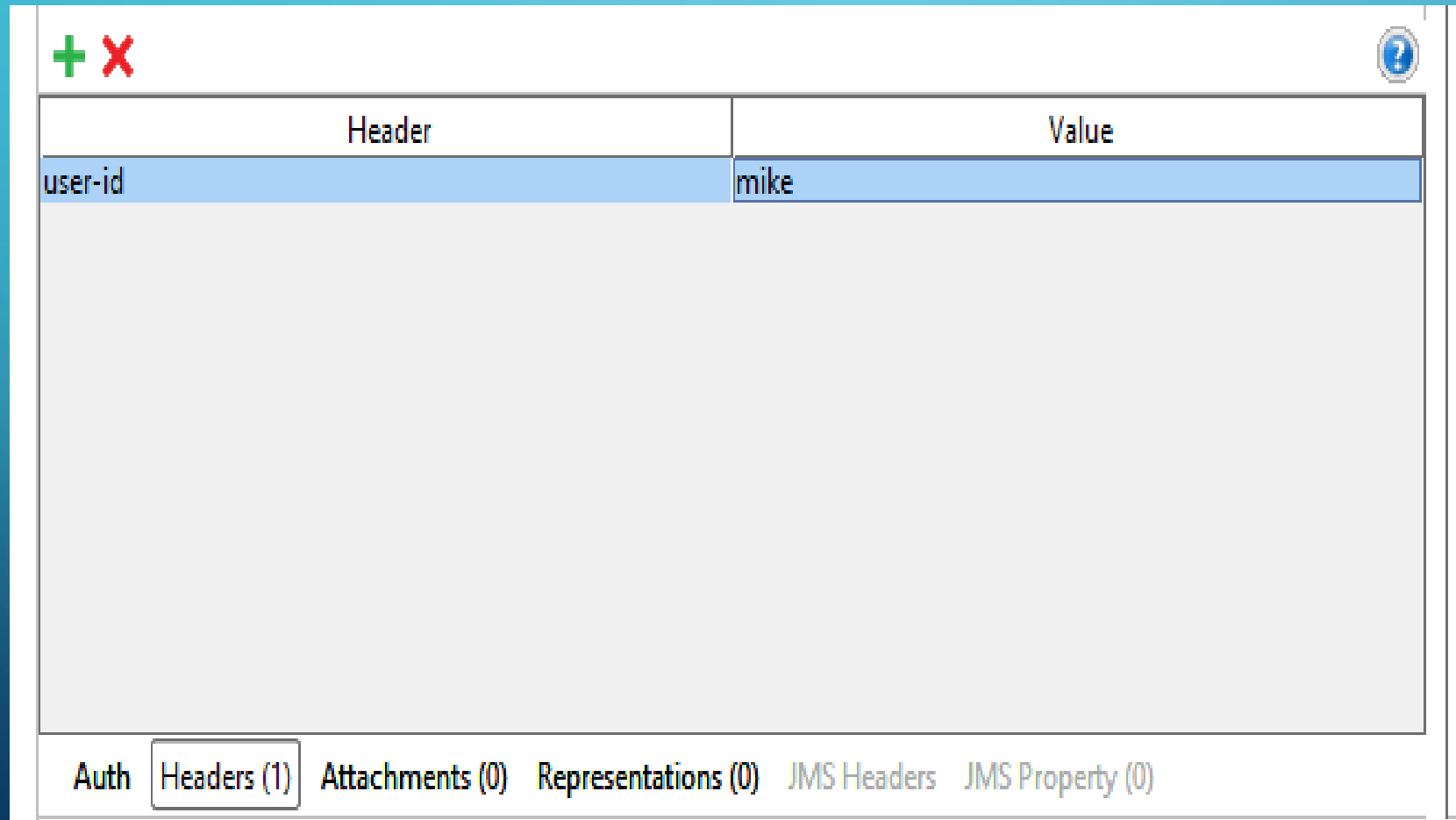


LAB 1 – TESTING THE WEB SERVICE

- Add the HTTP header 'user-id' along with a value of 'mike'.



LAB 1 – TESTING THE WEB SERVICE



The screenshot shows a window with a title bar containing a green plus sign, a red X, and a help icon. The main area is a table with two columns: 'Header' and 'Value'. The first row is highlighted in blue and contains the text 'user-id' and 'mike'. At the bottom of the window, there is a navigation bar with several tabs: 'Auth', 'Headers (1)', 'Attachments (0)', 'Representations (0)', 'JMS Headers', and 'JMS Property (0)'. The 'Headers (1)' tab is currently selected.

Header	Value
user-id	mike

Auth Headers (1) Attachments (0) Representations (0) JMS Headers JMS Property (0)

LAB 1 – TESTING THE WEB SERVICE

- Execute the web service using the green play button.

The screenshot displays a REST client interface for a request named "Request 1". The method is set to GET, the endpoint is http://10.2.5.250:10024, and the resource is /web/services/REST_1_ML/hello. The response is shown in the Raw tab as JSON:

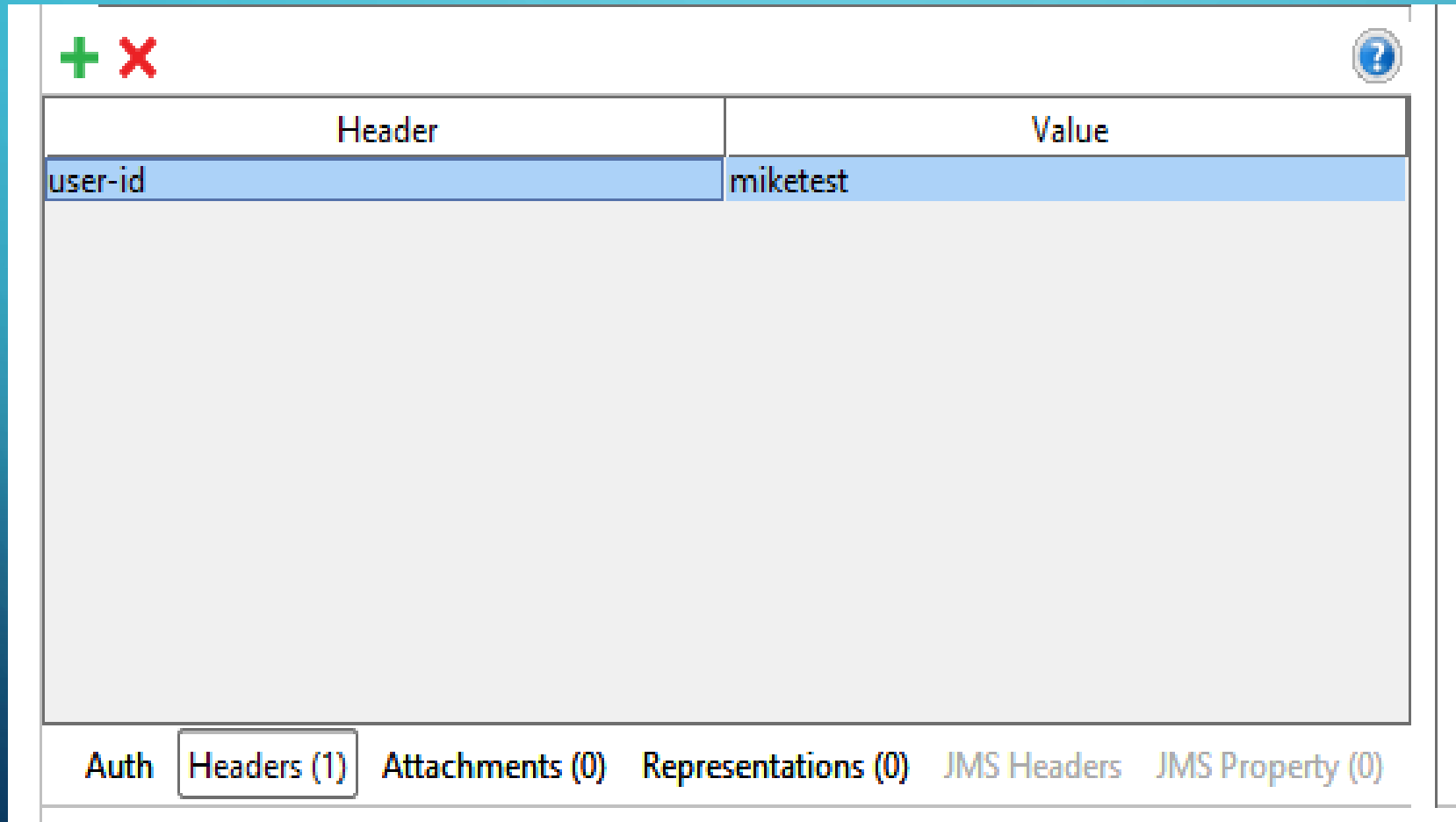
```
1 {"ParmOutMessage": {
2   "MessageSentIn": "You said hello",
3   "MessageSentOut": "I said goodbye"
4 }}
```

The interface also includes a table for request parameters and a "Required" checkbox labeled "Sets if parameter is required".

Name	Value	Style	Level
------	-------	-------	-------

LAB 1 – TESTING THE WEB SERVICE

- Change the HTTP header value.



The screenshot shows a window with a title bar containing a green plus sign, a red minus sign, and a blue question mark icon. Below the title bar is a table with two columns: "Header" and "Value". The table contains one row with the header "user-id" and the value "miketest". At the bottom of the window, there is a navigation bar with several tabs: "Auth", "Headers (1)", "Attachments (0)", "Representations (0)", "JMS Headers", and "JMS Property (0)". The "Headers (1)" tab is currently selected and highlighted.

Header	Value
user-id	miketest

Auth Headers (1) Attachments (0) Representations (0) JMS Headers JMS Property (0)

LAB 1 – TESTING THE WEB SERVICE

- Execute the service again.

The screenshot displays a web service testing interface. At the top, the method is set to GET, the endpoint is http://10.2.5.250:10024, and the resource is /web/services/REST_1_ML/hello. The response is shown in JSON format, indicating a successful call with an empty message.

Name	Value	Style	Level
------	-------	-------	-------

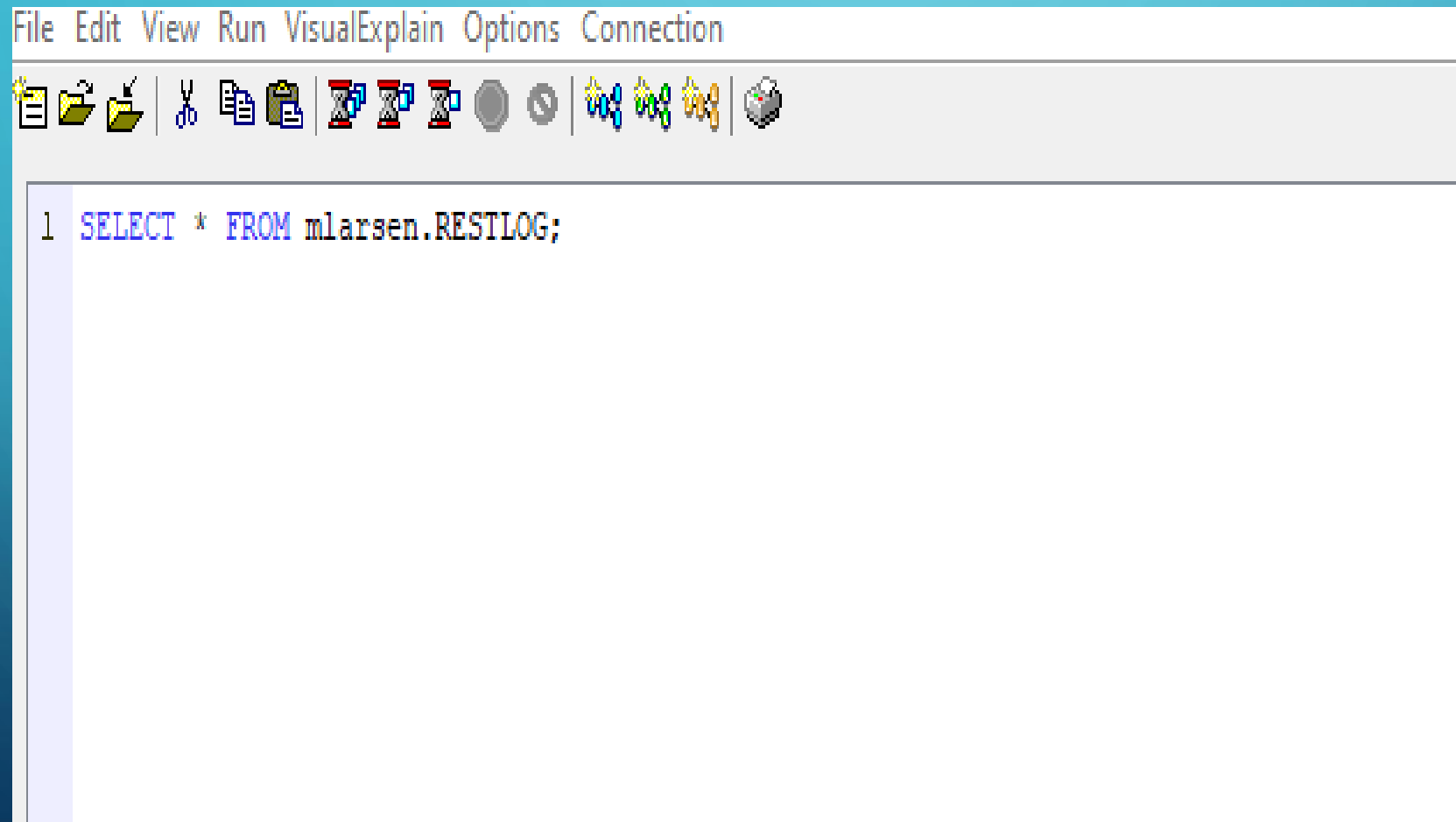
```
1 {"ParmOutMessage": {
2   "MessageSentIn": "",
3   "MessageSentOut": ""
4 }}
```

LAB 1 – TESTING THE WEB SERVICE

- The program is validating the HTTP header value for 'user-id' and only executing the 'GetOutMessage' subroutine if the value is 'mike'.

LAB 1 – CHECK THE REST SERVICE LOG

- Use ACS or STRSQL to view the REST log file.




The screenshot shows a SQL editor window with a menu bar (File, Edit, View, Run, Visual Explain, Options, Connection) and a toolbar containing various icons for file operations, execution, and debugging. The main text area contains the following SQL query:

```
1 SELECT * FROM mlarsen.RESTLOG;
```


LAB 1 – CHECK THE REST SERVICE LOG

- Use ACS or STRSQL to view the REST log file.

 SELECT * FROM mlarsen.RESTLOG

File Edit View

USERNAME	SERVICE	PROGRAMNAME	HTTPSTATUS	STARTTIME	ENDTIME	QUERYNAME
QUSER	REST example 1	REST_1_ML	200 OK	2017-12-10 13:10:08.464000	2017-12-10 13:10:08.464000	
QUSER	REST example 1	REST_1_ML	200 OK	2017-12-10 13:30:08.742000	2017-12-10 13:30:08.742000	
QUSER	REST example 1	REST_1_ML	401 Unauthorized	2017-12-10 13:33:49.628000	2017-12-10 13:33:49.628000	
QUSER	REST example 1	REST_1_ML	200 OK	2017-12-10 13:36:58.072000	2017-12-10 13:36:58.072000	

 SELECT * FROM mlarsen.RESTLOG

File Edit View

URI	URL_WS
... /web/services/REST_1_ML/hello	... http://10.2.5.250:10024/web/services/REST_1_ML/hello
... /web/services/REST_1_ML/hello	... http://10.2.5.250:10024/web/services/REST_1_ML/hello
... /web/services/REST_1_ML/hello	... http://10.2.5.250:10024/web/services/REST_1_ML/hello
... /web/services/REST_1_ML/hello	... http://10.2.5.250:10024/web/services/REST_1_ML/hello







LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Click 'Deploy New Service' or 'Deploy' button.

Manage Deployed Services

Data current as of Dec 10, 2017 3:09:40 PM.

Deployed services: ?

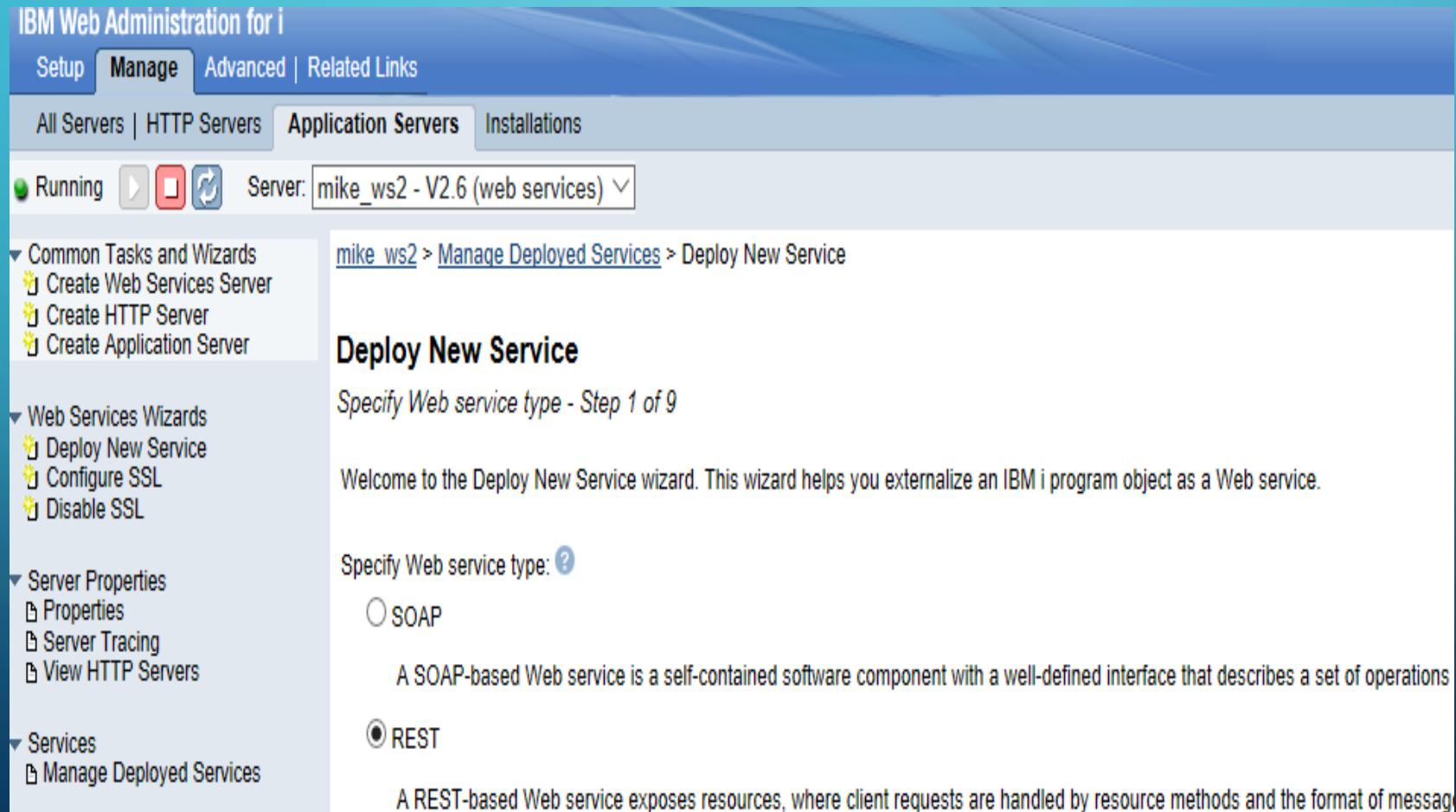
	Service name	Status	Type	Startup type	Service definition
	ConvertTemp	 Running	SOAP	Automatic	 View WSDL
	REST_1_ML	 Running	REST	Automatic	 View Swagger

Deploy

Refresh

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Select 'REST' and click 'Next'.



The screenshot displays the IBM Web Administration for i interface. The top navigation bar includes 'Setup', 'Manage', 'Advanced', and 'Related Links'. Below this, there are tabs for 'All Servers', 'HTTP Servers', 'Application Servers', and 'Installations'. The current server is identified as 'mike_ws2 - V2.6 (web services)'. The left sidebar contains a tree view with categories: 'Common Tasks and Wizards' (Create Web Services Server, Create HTTP Server, Create Application Server), 'Web Services Wizards' (Deploy New Service, Configure SSL, Disable SSL), 'Server Properties' (Properties, Server Tracing, View HTTP Servers), and 'Services' (Manage Deployed Services). The main content area shows the 'Deploy New Service' wizard, titled 'Specify Web service type - Step 1 of 9'. The wizard text reads: 'Welcome to the Deploy New Service wizard. This wizard helps you externalize an IBM i program object as a Web service.' Under the heading 'Specify Web service type: ?', there are two radio button options: 'SOAP' and 'REST'. The 'REST' option is selected. Below the 'SOAP' option, a description states: 'A SOAP-based Web service is a self-contained software component with a well-defined interface that describes a set of operations'. Below the 'REST' option, a description states: 'A REST-based Web service exposes resources, where client requests are handled by resource methods and the format of message'.

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Populate the library and ILE object name. This time choose *SRVPGM.

Deploy New Service

Specify Location of IBM i Program Object - Step 2 of 9

The IBM i object to be externalized as a Web service must be an existing ILE program (*PGM) or service program (*SRVPGM) located on the system.

Specify the program object for the Web service. ?

- Specify IBM i library and ILE program object name (Recommended)

You can specify the program object location by entering the name of the library that contains the program object, as well as the name of the program object.

Library name:

ILE Object name:

ILE Object type: *SRVPGM *PGM

- Browse the integrated file system for the IBM i program object

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Populate resource name and service description, then click ‘Next’.

[mike ws2](#) > [Manage Deployed Services](#) > [Deploy New Service](#)

Deploy New Service

Specify Name for Service - Step 3 of 9

The Web service to be externalized is a resource. The URI path template identifies matching patterns for incoming HTTP requests. You can further restrict what is allowed. [?](#)

Resource name:

Service description:

x

URI path template:

e.g. /temperature, /temperature/{temp:\d+}

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Uncheck 'detect field lengths' and fill out per screen shot.

Deploy New Service

Select Export Procedures to Externalize as a Web Service - Step 4 of 9

Exported procedures are entry points to a program object and are mapped to Web service operations. A procedure is a set of self-contained more procedures. A program contains only one procedure.

The table below lists all the exported procedures found in the program object that can be externalized through this Web service. Expand the by clients and what is returned by the Web service.

Detect length fields

Use parameter name as element name for data structures

Export procedures: [?](#)

Select	Procedure name/Parameter name	Usage	Data type	Count
<input checked="" type="checkbox"/>	▼ CREATEINVOICE			
	ININVOICE	input ▼	struct	
	HTTPSTATUS	output ▼	int	
	HTTPHEADERS	output ▼	char	10 ▼
<input checked="" type="checkbox"/>	▼ GETALL			
	INVOICESL_LENGTH	output ▼	int	
	CHANNEL	input ▼	char	
	STORE	input ▼	char	
	INVDATA	input ▼	char	
	INVOICES	output ▼	struct	INVOICESL_LENGTH ▼
	HTTPSTATUS	output ▼	int	

Select All

Deselect All

Expand All

Collapse All

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Populate the 'CreateInvoice' method. Note The HTTP method is 'POST'.

Deploy New Service

Specify Resource Method Information - Step 5 of 9

Procedures are mapped to resource methods. Each resource method needs to be defined to handle client requests by mapping an HTTP request method to

Specify resource method information. ?

Procedure name: CREATEINVOICE
URI path template for resource: /
HTTP request method: POST
URI path template for method: *NONE or...
HTTP response code output parameter: HTTPSTATUS
HTTP header array output parameter: HTTPHEADERS
Allowed input media types: *JSON or...
Returned output media types: *JSON or...

Whether to wrap input parameters:

- Wrap input parameters
 Do not wrap input parameters

Input parameter mappings:

Parameter name	Data type	Input source	Identifier	Default Value
ININVOICE	struct	*NONE		

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Populate the 'GetAll' method. Note The HTTP method is 'GET'.

Deploy New Service

Specify Resource Method Information - Step 5 of 9

Procedures are mapped to resource methods. Each resource method needs to be defined to handle client requests by mapping an HTTP request method

Specify resource method information. ?

Procedure name: GETALL
URI path template for resource: /
HTTP request method: GET
URI path template for method: /{Channel:w+}/{Store:ld} or...
HTTP response code output parameter: HTTPSTATUS
HTTP header array output parameter: *NONE
Allowed input media types: *ALL or...
Returned output media types: *JSON or...
Whether to wrap input parameters:
 Wrap input parameters
 Do not wrap input parameters

Input parameter mappings:

Parameter name	Data type	Input source	Identifier	Default Value
CHANNEL	char	*PATH_PARAM	Channel	*NONE or...
STORE	char	*PATH_PARAM	Store	*NONE or...
INVDATA	char	*PATH_PARAM	InvDate	*NONE or...

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Click 'Next'.

Deploy New Service

Specify User ID for this Service - Step 6 of 9

The service requires an IBM i user ID to run the program object that contains the Web service business logic.

Specify User ID for this Service: ?

Use server's user ID

The server's user ID must have the necessary authority to this program object and any other additional program objects.

Specify an existing user ID

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Click 'Next'.

Deploy New Service

Specify Library List - Step 7 of 9

The functionality of the IBM i program you want to externalize as a Web service may depend upon other IBM i programs in the system used.

Specify library list position for this Web service:

- Insert libraries in front of user library portion of the library list
- Insert libraries at the end of user library portion of the library list

Library list entries: [?](#)

	Library name
<input type="radio"/>	MLARSEN

Add

Remove All

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Click 'Next'.

Deploy New Service

Specify Transport Information to Be Passed - Step 8 of 9

Specify transport information to be passed to the web service implementation code.

Information to be passed to web service implementation code [?](#)

Specify Transport Metadata:

	Transport Metadata
<input type="checkbox"/>	QUERY_STRING
<input type="checkbox"/>	REMOTE_ADDR
<input type="checkbox"/>	REMOTE_USER
<input type="checkbox"/>	REQUEST_METHOD
<input type="checkbox"/>	REQUEST_URI
<input type="checkbox"/>	REQUEST_URL
<input type="checkbox"/>	SERVER_NAME
<input type="checkbox"/>	SERVER_PORT

Specify HTTP Headers:

HTTP Headers	
There are no entries for this table.	

LAB 2 – A WEB SERVICE WITH SELECT AND INSERT

- Click 'Finish'.

Deploy New Service

Summary - Step 9 of 9

When you click **Finish** the web service is deployed.

Service

Methods

Request Information

Resource name: REST_2_ML

Resource description: Rest lab 2

Service install path : /www/mike_ws2/webservices/services/REST_2_ML

URI path template: /

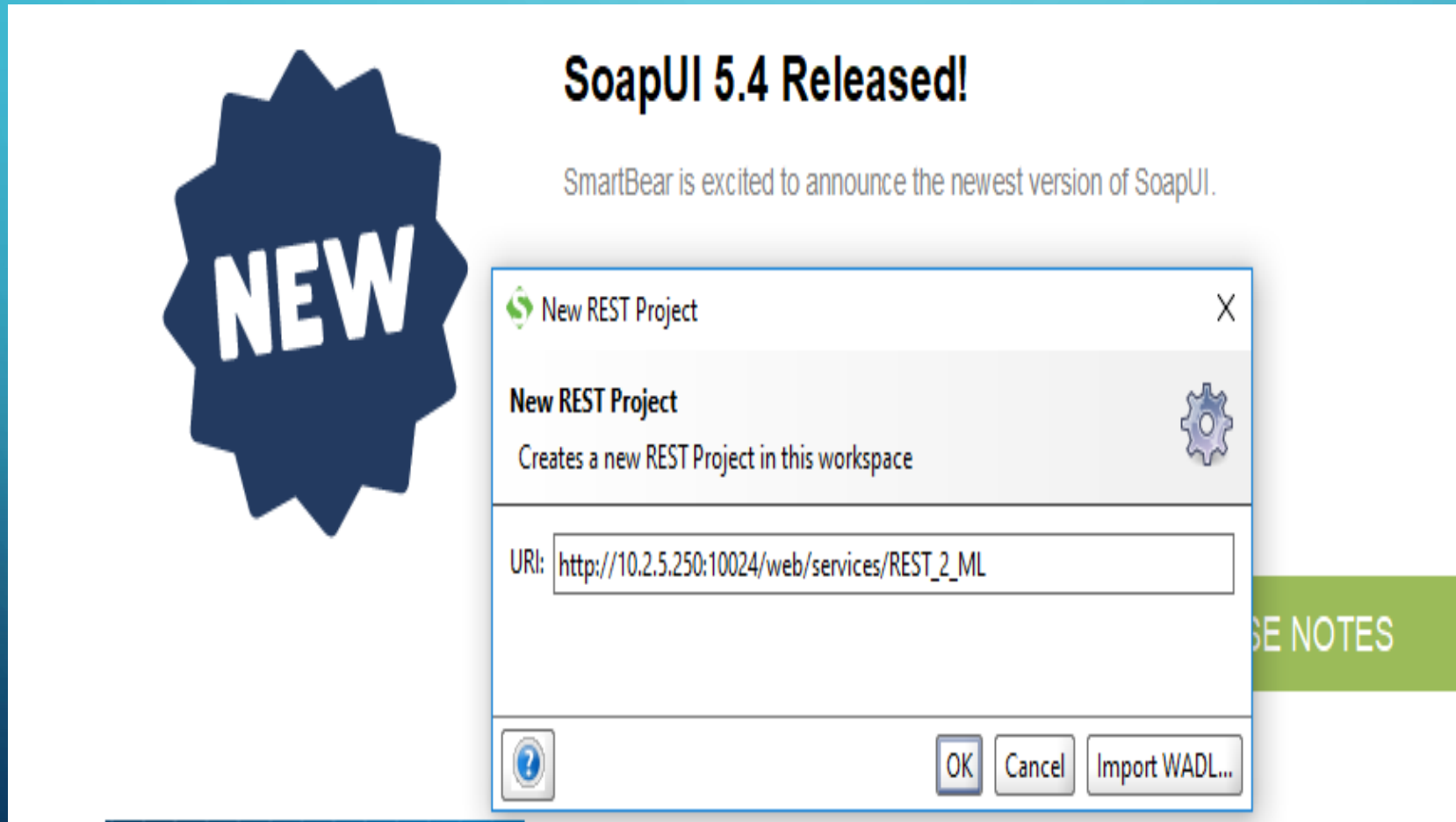
User ID for service: *SERVER (MLARSEN)

Program: /QSYS.LIB/MLARSEN.LIB/REST_2_ML.SRVPGM

Library list for service: MLARSEN

LAB 2 – TESTING THE WEB SERVICE – ‘CREATEINVOICE’

- Launch SoapUi and create a new REST project.



The image shows a screenshot of the SoapUI 5.4 release announcement. On the left, there is a dark blue starburst graphic with the word "NEW" in white. To the right, the text reads "SoapUI 5.4 Released!" followed by "SmartBear is excited to announce the newest version of SoapUI." Below this, a "New REST Project" dialog box is open. The dialog box has a title bar with the SoapUI logo and the text "New REST Project". The main content area shows "New REST Project" with a gear icon and the description "Creates a new REST Project in this workspace". Below this is a text input field for the URI, containing the text "http://10.2.5.250:10024/web/services/REST_2_ML". At the bottom of the dialog box, there are three buttons: a help button (question mark in a square), "OK", and "Cancel". To the right of the dialog box, a green banner with the text "SE NOTES" is partially visible.

LAB 2 – TESTING THE WEB SERVICE – ‘CREATEINVOICE’

- Change the method to ‘POST’ and populate the JSON being passed.

The screenshot shows a REST client interface for 'Request 1'. The configuration is as follows:

- Method:** POST
- Endpoint:** http://10.2.5.250:10024
- Resource:** /web/services/REST_2_ML

The request body is a JSON object with the following fields:

```
{
  "INCHANNEL": "RT",
  "INSTORE": "00099",
  "ININVDATE": "20171126",
  "ININVNUM": "000990000100022",
  "INTRNT": "01",
  "INTRNAMT": "200.00",
  "INTAXAMT": "6.00"
}
```

The 'Raw' tab is selected, and the 'JSON' format is chosen. The 'Media Type' is set to 'application/ison' and 'Post QueryString' is unchecked.

LAB 2 – TESTING THE WEB SERVICE – 'CREATEINVOICE'

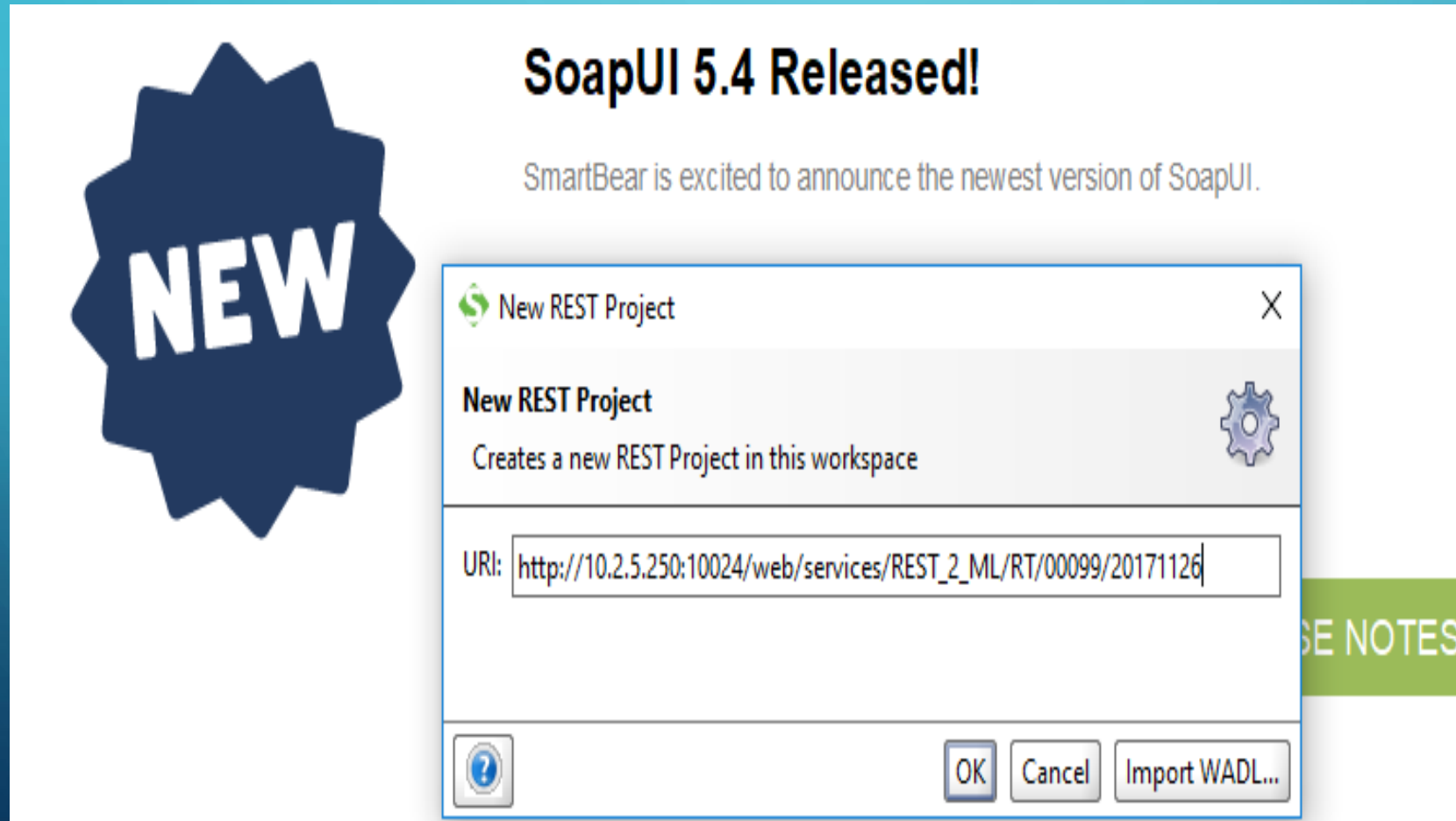
- Click the 'play' button, then check the status on the 'Headers' tab.

The screenshot displays an API testing interface for a REST client. The top section shows the request configuration: Method is POST, Endpoint is http://10.2.5.250:10024, and Resource is /web/services/REST_2_ML. The request body is a JSON object with the following fields: INCHANNEL, INSTORE, ININVDATA, ININNUM, INTRNT, INTRNAMT, and INTAXAMT. The response headers are visible in the bottom right section.

Header	Value
Content-Language	en-US
Date	Sun, 10 Dec 2017 20:52:00 GMT
Content-Length	0
#status#	HTTP/1.1 201 Created
Connection	close
Server	Apache
X-Powered-By	IBM i

LAB 2 – TESTING THE WEB SERVICE – ‘GETALL’

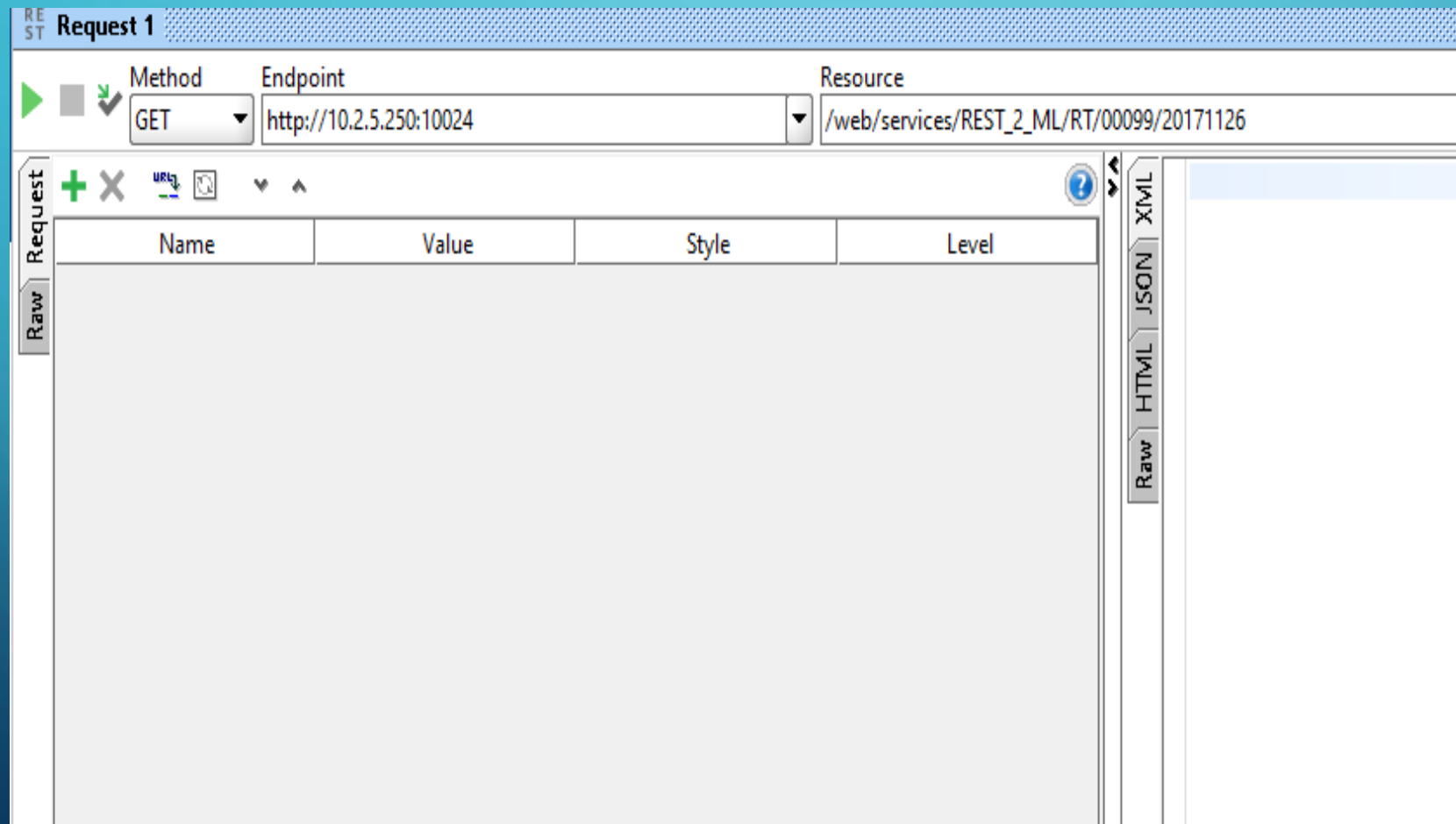
- Create a new REST project.



The screenshot shows a SoapUI 5.4 release announcement. On the left is a dark blue starburst with the word 'NEW' in white. The main text reads 'SoapUI 5.4 Released!' followed by 'SmartBear is excited to announce the newest version of SoapUI.' Overlaid on this is a 'New REST Project' dialog box. The dialog has a title bar with a close button, a gear icon, and the text 'New REST Project' and 'Creates a new REST Project in this workspace'. A text field labeled 'URI:' contains the URL 'http://10.2.5.250:10024/web/services/REST_2_ML/RT/00099/20171126'. At the bottom are buttons for '?', 'OK', 'Cancel', and 'Import WADL...'. A green banner with the text 'SE NOTES' is partially visible on the right side of the dialog.

LAB 2 – TESTING THE WEB SERVICE – ‘GETALL’

- Click the ‘play’ button, then check the JSON tab for output.



The screenshot shows a REST client interface for 'Request 1'. The configuration is as follows:

Method	Endpoint	Resource
GET	http://10.2.5.250:10024	/web/services/REST_2_ML/RT/00099/20171126

Below the configuration, there is a table with the following columns: Name, Value, Style, and Level. The table is currently empty.

On the right side, there are tabs for 'Raw', 'HTML', 'JSON', and 'XML'. The 'JSON' tab is currently selected.

LAB 2 – TESTING THE WEB SERVICE – ‘GETALL’

The screenshot displays a REST client interface for a request. The request details are as follows:

Method	Endpoint	Resource	Parameters
GET	http://10.2.5.250:10024	/web/services/REST_2_ML/RT/00099/20171126	

The response is shown in JSON format:

```
1 {
2   "INVOICESL_LENGTH": 3,
3   "INVOICES": [
4     {
5       "DSCHANNEL": "RT",
6       "DSSTORE": "00099",
7       "DSINVDATE": "20171126",
8       "DSINVNUM": "000990000100016",
9       "DSTRNT": "01",
10      "DSTRNAMT": "200.00",
11      "DSTAXAMT": "6.00"
12    },
13    {
14      "DSCHANNEL": "RT",
15      "DSSTORE": "00099",
16      "DSINVDATE": "20171126",
17      "DSINVNUM": "000990000100017",
18      "DSTRNT": "01",
19      "DSTRNAMT": "210.00",
20      "DSTAXAMT": "7.00"
21    },
22    {
23      "DSCHANNEL": "RT",
24      "DSSTORE": "00099",
25      "DSINVDATE": "20171126",
26      "DSINVNUM": "000990000100022",
27      "DSTRNT": "01",
28      "DSTRNAMT": "200.00",
29      "DSTAXAMT": "6.00"
30    }
31  ]
32 }
```

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE







- Click 'Deploy New Service' or 'Deploy' button.

mike ws2 > Manage Deployed Services

Manage Deployed Services

Data current as of Dec 10, 2017 4:07:02 PM.

Deployed services: ?

	Service name	Status	Type	Startup type	Service definition
<input type="checkbox"/>	ConvertTemp	 Running	SOAP	Automatic	 View WSDL
<input type="checkbox"/>	REST_1_ML	 Running	REST	Automatic	 View Swagger
<input checked="" type="checkbox"/>	REST_2_ML	 Running	REST	Automatic	 View Swagger

Deploy

Stop

Properties

Uninstall

Refresh

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Select 'REST' and click 'Next'.

[mike_ws2](#) > [Manage Deployed Services](#) > [Deploy New Service](#)

Deploy New Service

Specify Web service type - Step 1 of 9

Welcome to the Deploy New Service wizard. This wizard helps you externalize an IBM i program object as a Web service.

Specify Web service type: ?

SOAP

A SOAP-based Web service is a self-contained software component with a well-defined interface that describes a set of operations that are accessible over a network.

REST

A REST-based Web service exposes resources, where client requests are handled by resource methods and the format of messages that are exchanged is based on the representation of the resources.

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Populate the library, ILE object name, and choose *SRVPGM.

Deploy New Service

Specify Location of IBM i Program Object - Step 2 of 9

The IBM i object to be externalized as a Web service must be an existing ILE program (*PGM) or service program (*SRVPGM).

Specify the program object for the Web service. ?

Specify IBM i library and ILE program object name (Recommended)

You can specify the program object location by entering the name of the library that contains the program object, as well as the object name.

Library name:

ILE Object name:

ILE Object type: *SRVPGM *PGM


Browse the integrated file system for the IBM i program object

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE


- Populate resource name and service description, then click ‘Next’.

Deploy New Service

Specify Name for Service - Step 3 of 9

The Web service to be externalized is a resource. The URI path template identifies matching patterns for incoming HTTP requests. The [URI path template](#) can be used to further restrict what is allowed. 

Resource name:

Service description: 

URI path template:

e.g. /temperature, /temperature/{temp:id+}

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Uncheck 'detect field lengths' and fill out per screen shot.

Deploy New Service

Select Export Procedures to Externalize as a Web Service - Step 4 of 9

Exported procedures are entry points to a program object and are mapped to Web service operations. A procedure is a set of self-contained more procedures. A program contains only one procedure.

The table below lists all the exported procedures found in the program object that can be externalized through this Web service. Expand the by clients and what is returned by the Web service.

Detect length fields

Use parameter name as element name for data structures

Export procedures: ?

Select	Procedure name/Parameter name	Usage	Data type	Count
<input checked="" type="checkbox"/>	▼ DELETEINVOICE			
	CHANNEL	input ▼	char	
	STORE	input ▼	char	
	INVDATA	input ▼	char	
	INVOICENUMBER	input ▼	char	
	HTTPSTATUS	output ▼	int	
	HTTPHEADERS	output ▼	char	10 ▼
<input checked="" type="checkbox"/>	▼ UPDATEINVOICE			
	UPINVOICE	input ▼	struct	
	HTTPSTATUS	output ▼	int	
	HTTPHEADERS	output ▼	char	10 ▼

Select All

Deselect All

Expand All

Collapse All

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Populate the 'DeleteInvoice' method. Note The HTTP method is 'DELETE'.

Deploy New Service

Specify Resource Method Information - Step 5 of 9

Procedures are mapped to resource methods. Each resource method needs to be defined to handle client requests by mapping an HTTP request method to a resource method.

Specify resource method information. ?

Procedure name: DELETEINVOICE
URI path template for resource: /
HTTP request method: DELETE ▾
URI path template for method: /{Channel:\w+}/{Store:\d} or... ▾
HTTP response code output parameter: HTTPSTATUS ▾
HTTP header array output parameter: HTTPHEADERS ▾
Allowed input media types: *ALL or... ▾
Returned output media types: *JSON or... ▾
Whether to wrap input parameters:
 Wrap input parameters
 Do not wrap input parameters

Input parameter mappings:

Parameter name	Data type	Input source	Identifier	Default Value
CHANNEL	char	*PATH_PARAM ▾	Channel ▾	*NONE or... ▾
STORE	char	*PATH_PARAM ▾	Store ▾	*NONE or... ▾
INVDATE	char	*PATH_PARAM ▾	InvDate ▾	*NONE or... ▾
INVOICENUMBER	char	*PATH_PARAM ▾	InvoiceNumber ▾	*NONE or... ▾

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Populate the 'UpdateInvoice' method. Note The HTTP method is 'PUT'.

Deploy New Service

Specify Resource Method Information - Step 5 of 9

Procedures are mapped to resource methods. Each resource method needs to be defined to handle client requests by mapping an HTTP request

Specify resource method information. ?

Procedure name: UPDATEINVOICE
URI path template for resource: /
HTTP request method: PUT
URI path template for method: *NONE or...
HTTP response code output parameter: HTTPSTATUS
HTTP header array output parameter: HTTPHEADERS
Allowed input media types: *JSON or...
Returned output media types: *JSON or...
Whether to wrap input parameters:
 Wrap input parameters
 Do not wrap input parameters

Input parameter mappings:

Parameter name	Data type	Input source	Identifier	Default Value
UPINVOICE	struct	*NONE		

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Click 'Next'.

Deploy New Service

Specify User ID for this Service - Step 6 of 9

The service requires an IBM i user ID to run the program object that contains the Web service business logic.

Specify User ID for this Service: ?

Use server's user ID

The server's user ID must have the necessary authority to this program object and any other additional program objects.

Specify an existing user ID

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Click 'Next'.

Deploy New Service

Specify Library List - Step 7 of 9

The functionality of the IBM i program you want to externalize as a Web service may depend upon other IBM i programs in the system. Specify all used.

Specify library list position for this Web service:

- Insert libraries in front of user library portion of the library list
- Insert libraries at the end of user library portion of the library list

Library list entries: [?](#)

	Library name
<input type="radio"/>	MLARSEN

Add

Remove All

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Click 'Next'.

Deploy New Service

Specify Transport Information to Be Passed - Step 8 of 9

Specify transport information to be passed to the web service implementation code.

Information to be passed to web service implementation code [?](#)

Specify Transport Metadata:

	Transport Metadata
<input type="checkbox"/>	QUERY_STRING
<input type="checkbox"/>	REMOTE_ADDR
<input type="checkbox"/>	REMOTE_USER
<input type="checkbox"/>	REQUEST_METHOD
<input type="checkbox"/>	REQUEST_URI
<input type="checkbox"/>	REQUEST_URL
<input type="checkbox"/>	SERVER_NAME
<input type="checkbox"/>	SERVER_PORT

Specify HTTP Headers:

	HTTP Headers
<i>There are no entries for this table.</i>	

Add

Remove All

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

- Click 'Finish'.

Deploy New Service

Summary - Step 9 of 9

When you click **Finish** the web service is deployed.

Service

Methods

Request Information

Resource name: REST_3_ML

Resource description: Rest Lab 3

Service install path : /www/mike_ws2/webservices/services/REST_3_ML

URI path template: /

User ID for service: *SERVER (MLARSEN)

Program: /QSYS.LIB/MLARSEN.LIB/REST_3_ML.SRVPGM









Library list for service: MLARSEN

LAB 3 – A WEB SERVICE WITH UPDATE AND DELETE

Manage Deployed Services

Data current as of Dec 10, 2017 4:28:02 PM.

Deployed services: ?

	Service name	Status	Type	Startup type	Service definition
<input type="checkbox"/>	ConvertTemp	 Running	SOAP	Automatic	 View WSDL
<input type="checkbox"/>	REST_1_ML	 Running	REST	Automatic	 View Swagger
<input type="checkbox"/>	REST_2_ML	 Running	REST	Automatic	 View Swagger
<input checked="" type="checkbox"/>	REST_3_ML	 Running	REST	Automatic	 View Swagger

Deploy

Stop

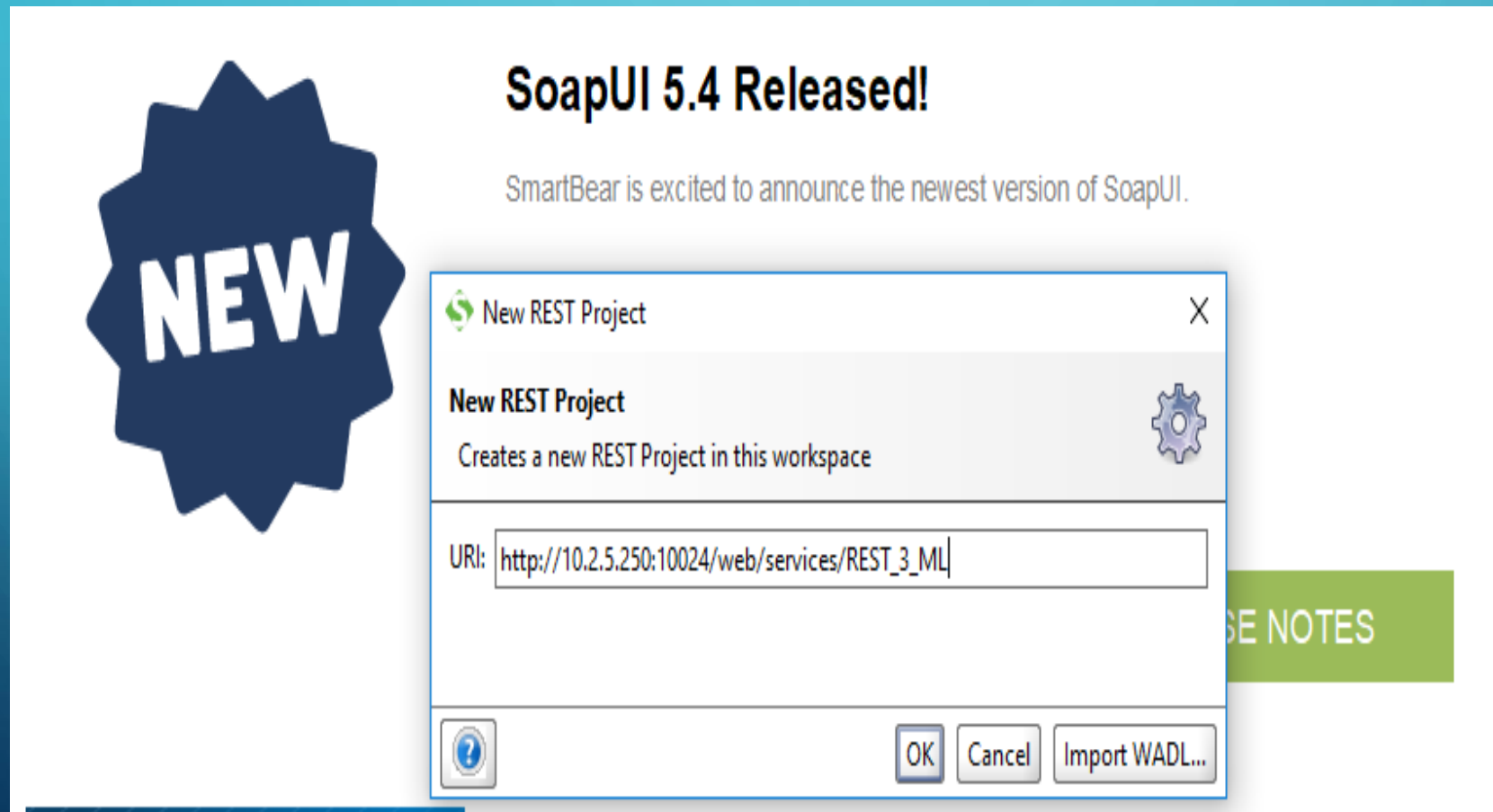
Properties

Uninstall

Refresh

LAB 3 – TESTING THE WEB SERVICE – ‘UPDATEINVOICE’

- Launch SoapUI and create a new REST project.



The screenshot shows the SoapUI 5.4 release announcement page. On the left is a dark blue starburst with the word "NEW" in white. The main text reads "SoapUI 5.4 Released!" followed by "SmartBear is excited to announce the newest version of SoapUI." Below this is a "New REST Project" dialog box. The dialog box has a title bar with a green icon and the text "New REST Project" and a close button. The main area contains the text "New REST Project" with a gear icon, "Creates a new REST Project in this workspace", and a text input field for the URI containing "http://10.2.5.250:10024/web/services/REST_3_ML". At the bottom are buttons for "?", "OK", "Cancel", and "Import WADL...". A green button labeled "SEE NOTES" is partially visible on the right side of the dialog box.

LAB 3 – TESTING THE WEB SERVICE – ‘UPDATEINVOICE’

- Update Amount and Tax Amount on Invoice ‘000990000100022’.

```
SELECT * FROM mlarsen.Sales_History
```

File Edit View

SH_CHANNEL	SH_STORE	SH_INVOICECENTURY	SH_INVOICEYEAR	SH_INVOICEMONTH	SH_INVOICEDAY	SH_INVOICENUMBER	SH_TRANSACTION_TYPE	SH_AMOUNT	SH_TAX_AMOUNT
RT	99	20	17	11	26	000990000100016	01	200.00	6.00
RT	99	20	17	11	26	000990000100017	01	210.00	7.00
RT	99	20	17	11	26	000990000100022	01	200.00	6.00

LAB 3 – TESTING THE WEB SERVICE – ‘UPDATEINVOICE’

- Populate the JSON document and execute the Web Service.

The screenshot shows a REST client interface for a request named "Request 1". The configuration is as follows:

- Method:** PUT
- Endpoint:** http://10.2.5.250:10024
- Resource:** /web/services/REST_3_ML

The request body is a JSON document with the following content:

```
{
  "UPCHANNEL": "RT",
  "UPSTORE": "00099",
  "UPINVDATE": "20171126",
  "UPINVNUM": "000990000100022",
  "UPTRANTYPE": "01",
  "UPTRANAMT": "999.00",
  "UPTAXAMT": "99.99"
}
```

The interface also shows a table for parameters, a "Required" checkbox, a "Type" dropdown, and "Options" with an "Add.." button. The "Media Type" is set to "application/ison" and "Post QueryString" is unchecked.

LAB 3 – TESTING THE WEB SERVICE – ‘UPDATEINVOICE’

- Check the Headers tab for the result.

The screenshot shows a REST client interface for a PUT request. The Method is PUT and the Endpoint is http://10.2.5.250:10024. The Resource is /web/services/REST_3_ML. The Request body is a JSON object with the following fields:

```
{
  "UPCHANNEL": "RT",
  "UPSTORE": "00099",
  "UPINVDATE": "20171126",
  "UPINVNUM": "000990000100022",
  "UPTRANTYPE": "01",
  "UPTRANAMT": "999.00",
  "UPTAXAMT": "99.99"
}
```

The Media Type is application/ison. The Response is a 204 No Content status, indicating a successful update. The Headers tab shows the following response headers:

Header	Value
Content-Language	en-US
Date	Sun, 10 Dec 2017 21:42:55 GMT
Content-Length	0
#status#	HTTP/1.1 204 No Content
Connection	close
Server	Apache
X-Powered-By	IBM i

LAB 3 – TESTING THE WEB SERVICE – ‘UPDATEINVOICE’

- Query the Sales History table to see the update.

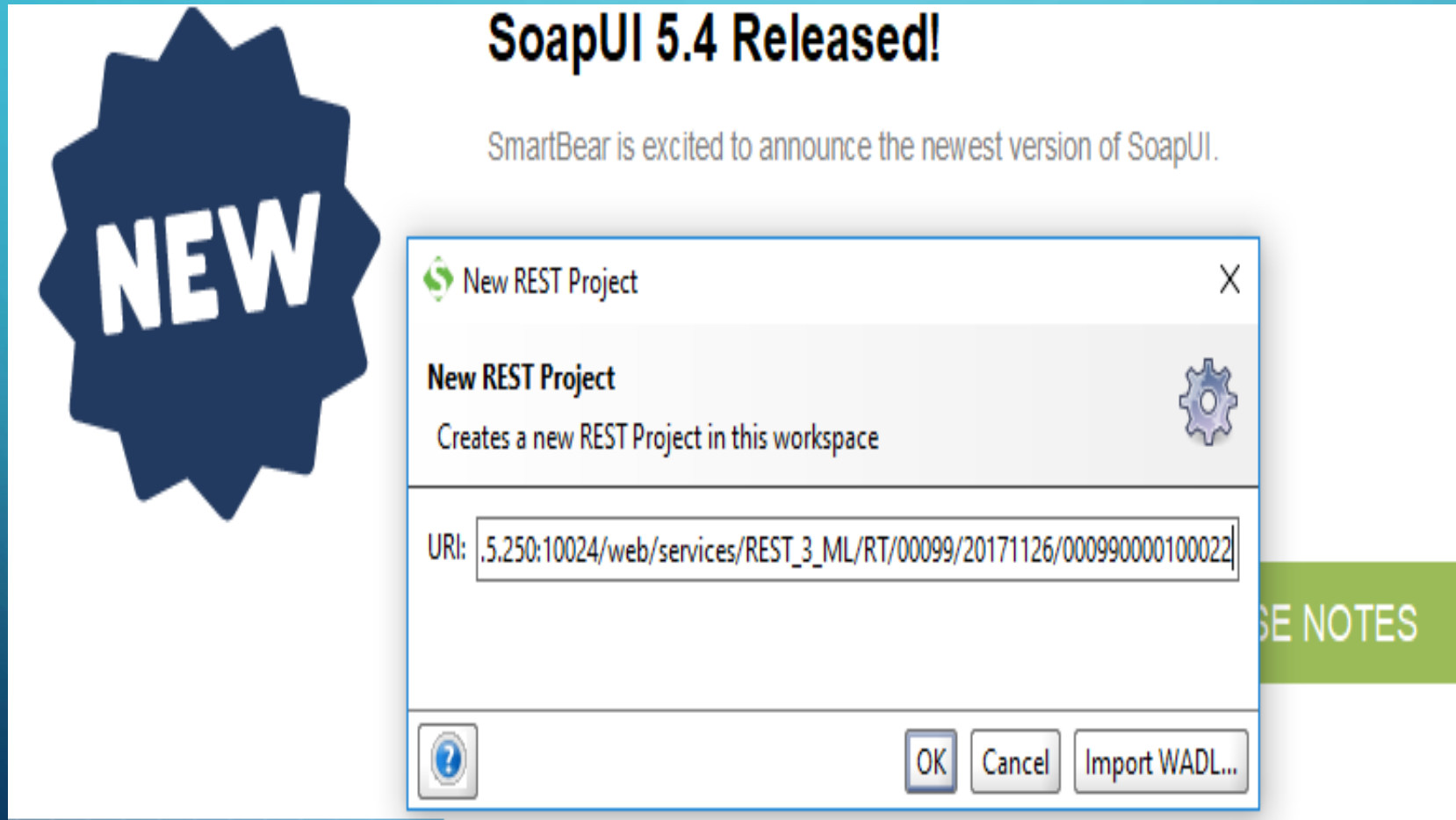
```
SELECT * FROM mlarsen.Sales_History
```

File Edit View

SH_CHANNEL	SH_STORE	SH_INVOICECENTURY	SH_INVOICEYEAR	SH_INVOICEMONTH	SH_INVOICEDAY	SH_INVOICENUMBER	SH_TRANSACTION_TYPE	SH_AMOUNT	SH_TAX_AMOUNT
RT	99	20	17	11	11	26 000990000100016	01	200.00	6.00
RT	99	20	17	11	11	26 000990000100017	01	210.00	7.00
RT	99	20	17	11	11	26 000990000100022	01	999.00	99.99

LAB 3 – TESTING THE WEB SERVICE – ‘DELETEINVOICE’

- Launch SoapUI and create a new REST project.



LAB 3 – TESTING THE WEB SERVICE – ‘DELETEINVOICE’

- Change the method to ‘DELETE’ and click the ‘Play’ button.

The screenshot displays a REST client interface for configuring a request. The top bar shows 'Request 1' with a 'Play' button and a status icon. Below this, the 'Method' is set to 'DELETE', the 'Endpoint' is 'http://10.2.5.250:10024', and the 'Resource' is '/web/services/REST_3_ML/RT/00099/20171126/000990000100022'. A table with columns 'Name', 'Value', 'Style', and 'Level' is currently empty. Below the table, there are settings for 'Required' (checkbox), 'Type' (dropdown), and 'Options' (text input with 'Add..' button). At the bottom, the 'Media Type' is set to 'application/ison' and the 'Post QueryString' checkbox is unchecked.

Name	Value	Style	Level
------	-------	-------	-------

Required: Sets if parameter is required

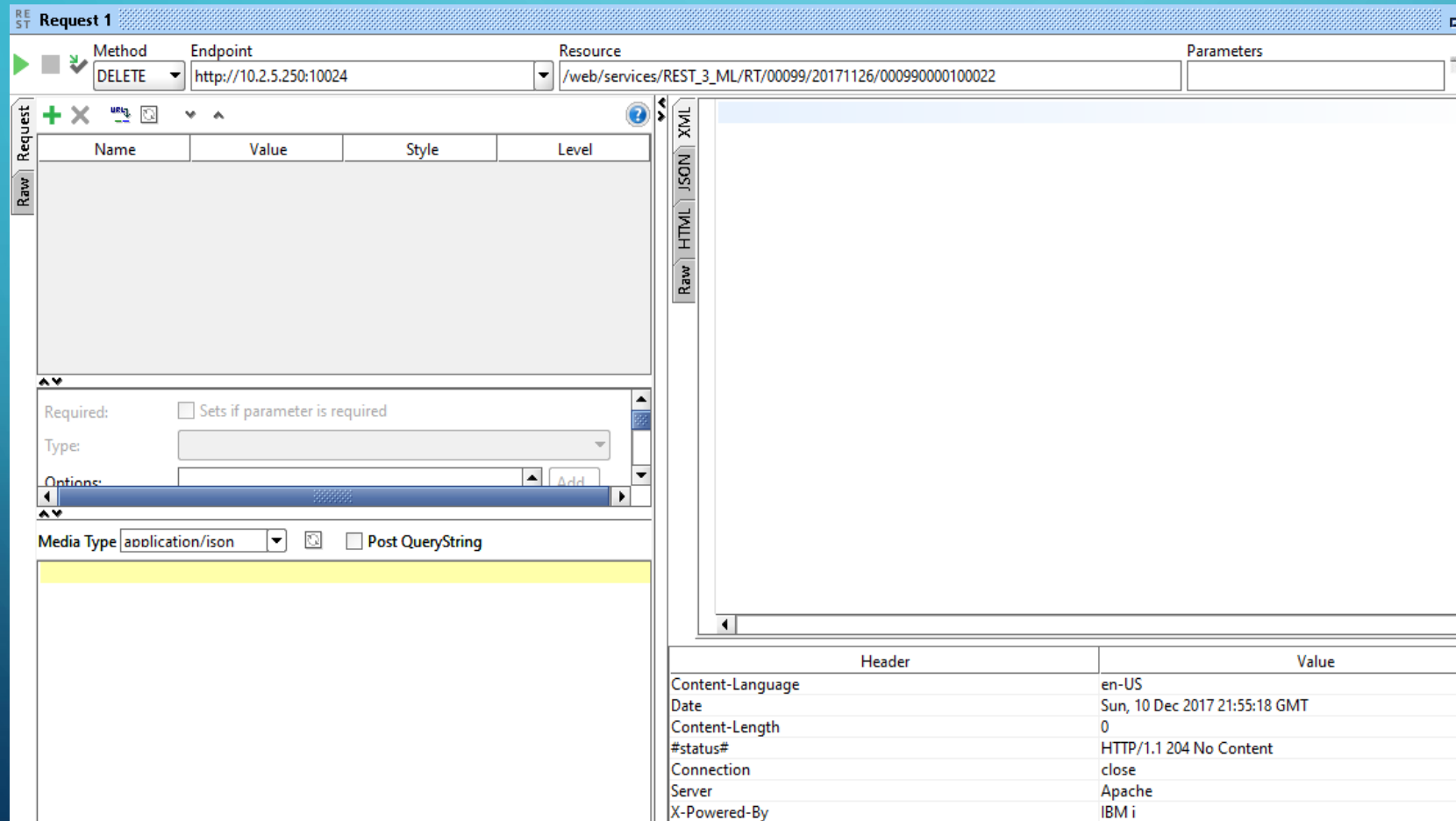
Type:

Options: Add..

Media Type: application/ison Post QueryString

LAB 3 – TESTING THE WEB SERVICE – ‘DELETEINVOICE’

- Click on the ‘Headers’ tab to check the result.



The screenshot shows a REST client interface with the following details:

- Request 1** (Method: DELETE, Endpoint: http://10.2.5.250:10024, Resource: /web/services/REST_3_ML/RT/00099/20171126/000990000100022)
- Request Tab:** A table with columns Name, Value, Style, and Level. Below it are fields for Required, Type, and Options.
- Media Type:** application/json
- Response Tab:** A table showing the following headers and values:

Header	Value
Content-Language	en-US
Date	Sun, 10 Dec 2017 21:55:18 GMT
Content-Length	0
#status#	HTTP/1.1 204 No Content
Connection	close
Server	Apache
X-Powered-By	IBM i

LAB 3 – TESTING THE WEB SERVICE – ‘DELETEINVOICE’

- Run a query in ACS to make sure the invoice was deleted.

SELECT * FROM mlarsen.Sales_History

File Edit View

SH_CHANNEL	SH_STORE	SH_INVOICECENTURY	SH_INVOICEYEAR	SH_INVOICEMONTH	SH_INVOICEDAY	SH_INVOICENUMBER	SH_TRANSACTION_TYPE	SH_AMOUNT	SH_TAX_AMOUNT
RT	99	20	17	11	26	000990000100016	01	200.00	6.00
RT	99	20	17	11	26	000990000100017	01	210.00	7.00

The background is a dark blue gradient. In the corners, there are decorative white line-art patterns resembling circuit traces or neural network connections. These patterns consist of straight lines of varying lengths and angles, ending in small white circles. The patterns are located in the top-left, top-right, bottom-left, and bottom-right corners.

Questions or comments?



Thank you!