A picture containing text, font, graphics, logo

Description automatically generated

**Modernizing CICS Applications**

Session 4140

Lab Exercise Guide

Andy Armstrong, Anthony Papageorgiou

Technical Enablement Specialists

[andy.armstrong@uk.ibm.com](mailto:andy.armstrong@uk.ibm.com), [a.papageorgiou@uk.ibm.com](mailto:a.papageorgiou@uk.ibm.com)

Using the web browser, access the lab using the following URL: <https://emitchj.github.io/WSC-CICSzVA-Registration/>

Using the web browser, access the lab guide using the following URL. This lab guide is in Git Hub.

Lab exercise Guide: https://github.com/IBMTechSales/klp-think2022-labs/tree/master/1152-Mono2Micro-refactorJavaAppsToMicroservices

***Available Lab Exercises***

* Implementing Web Services in CICS
  + L20 – SOAP based web services.
  + L90 – JSON based web services.
* Java Applications in CICS
  + L34 – How to deploy a CICS application program coded in Java using the OSGi JVM
  + L72 – Java Servlet with LINK to COBOL program
  + L93 – RESTful JSON with LINK to COBOL program using JAX-RS, JSON4J, JZOS
* z/OS Connect (CICS in Provider mode)
  + L97 – zOS Connect EE calling a COBOL application.

***Lab Exercise Key:***

***(z/OS Userids, Password, Port, etc)***

The labs have symbols which you will need to substitute. Symbols look like this:

<zOS\_userid>.CICSLAB.UTIL

You will substitute using one of these values:

* <zOS\_userid> : USER1
* <zOS\_Password> : USER1
* <CICS\_Port : 1423 (used for all labs except Liberty labs)
* <CICS\_APPLID> : CICS1
* <Servlet\_JSP\_Port> : 1424 (used for Liberty labs)
* <zOS\_DNS\_Name> : ws31.washington.ibm.com

Results:

**USER1.CICSLAB.UTIL**

**Note:** Screen shots are NOT 100% accurate. Follow the text explanations rather than what you see in the screen shot.

**Personal Communications Emulator Tips**

**Note:** The 3270-terminal sessions for TSO and OMVS screen shots in these exercises are shown in reverse video simply for printing purposes.

There are 3270-emulator keyboard mapping issues encountered when using a web browser to access Windows desktop. As a result, not all keyboard mappings described in this document may not work as intended.

Below is a guide to help you resolve issues if you have inconsistent or non-working 3270 keys.

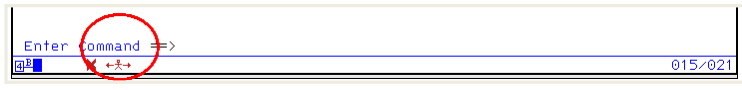
Any references to the Enter key in non-3270 windows, OMVS terminal session, etc. refers to the key labeled Enter on the keyboard.

* The 3270-emulator used for this workshop (IBM Personal Communication) maps the 3270 *enter* key to the right Ctrl key (see below). If the right Ctrl key does not work, try using the key labeled Enter or Return on your keyboard.
* The 3270 *newline* key has been mapped to the Shift-Enter key sequence.
* The 3270 *clear* key has been mapped to the Cntl-Enter key sequence.
* If all else fails, select *Actions* on the 3270-emulator tool bar and then select the Display *Popup Keypad* option. This displays the popup below where you select various keys.

A screenshot of a computer

Description automatically generated

* z/OS Updates Different 3270-terminal emulators will display an icon like the ‘*Personal Communications’* icon below at the bottom of the screen when the keyboard is locked. If this occurs use the left **Ctrl** key to reset or free the keyboard.



**Configuring the Keyboard with Mac keyboards**

* To reconfigure the IBM Personal Communications keyboard when using a Mac keyboard, click on *View* on the tool bar and select the Tool Bar option. Next click on the Remap Keyboard functions icon (see below).

A screenshot of a computer

Description automatically generated

* To set the **Return** key to perform the *enter* function, click on the Return (1) key, use the pull down arrow to select *Enter* from the function list menu (2), and then in the *Change Current Actions* *for Selected Key* section, click on the arrow to the left of *Base (3)* to set the default function of pressing the **Return** key to [*enter*] (refer to the picture below)

A computer screen shot of a keyboard

Description automatically generated

* To set the **Return** key to perform the *clear* the screen function, click on the **Return** (1) key, use the pull down arrow to select *Clear Screen* from the function list menu (2), and then in the *Change Current Actions* for Selected Key section, click on the arrow to the left of *Shift* (3) to set the default function of pressing the **Shift-Return** key sequence [*clear*].
* To set the **Return** key to perform the *new line* function, click on the **Return** (1) key, use the pull down arrow to select **New Line** from the function list menu (2), and then in the *Change Current Actions* for Selected Key section, click on the arrow to the left of *Ctrl* (3) to set the default function of pressing the **Ctrl-Return key** sequence to move the cursor down a line [*newline*].
* To set the **Return** key to reset a locked keyboard, click on the **Return** (1) key, use the pull down arrow to select **Reset** from the function list menu (2), and then in the *Change Current Actions* for Selected Key section, click on the arrow to the left of *CtrlShift* (3) to set the default function of pressing the **Ctrl-Shift-Return key** sequence to reset or unlock a key board [*reset*].
* When finished, your keyboard actions for the Return key should match the picture above.
* To set the **Ctrl-C** key sequence to perform the *copy* function, click on the **C** key (1) and then in the *Change Current Actions for Selected Key* section, click on the arrow to the left of *Ctrl* (2) to set the default function of pressing the **Ctrl-C** key sequence to [*edit-copy*] (3) (*refer to the picture below*).

A screenshot of a computer

Description automatically generated

* To set the **Ctrl-V** key sequence to perform the *paste* function, click on the **V** key (1) and then in the *Change Current Actions for Selected Key* section, click on the arrow to the left of *Ctrl* (2) to set the default function of pressing the **Ctrl-V** key sequence to [*edit-paste*] (3) (*refer to the picture below*).

A computer screen shot of a keyboard

Description automatically generated