

IBM – CICS and DevOps Workshop



## **Lab – Git-based DevOps Pipeline for Z Application**

Wazi Developer for VS Code

*Lab Version V1.1*

---

**October 2022**

Please send any comments on this lab exercise to:

George Ge: [ypge@au1.ibm.com](mailto:ypge@au1.ibm.com)

Lauren Li: [lauren.k.li@ibm.com](mailto:lauren.k.li@ibm.com)

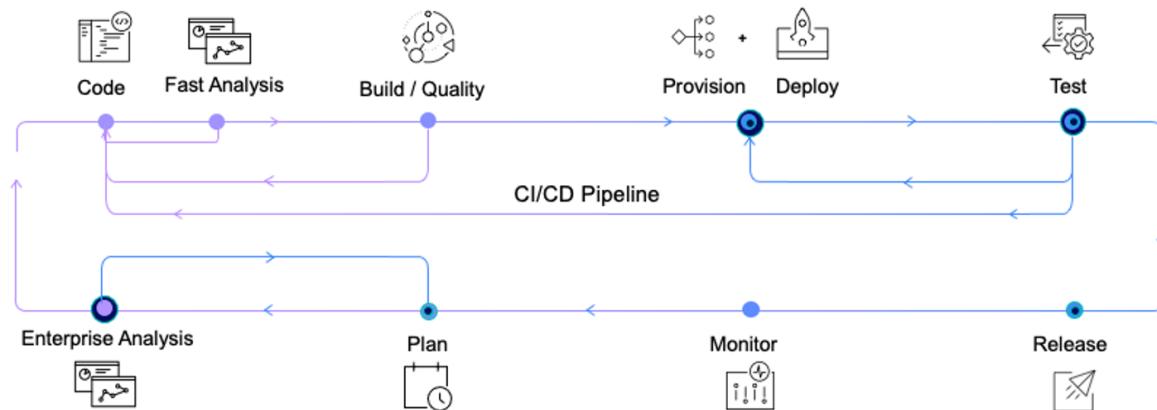
## Overview

DevOps is the enterprise capability of continuous software delivery to end-users. Its goal is to make business and IT operations more agile. It enables IT to deliver business requirements quickly and efficiently so that business operations can incorporate customer feedback and react swiftly to faster-changing market, economic and social conditions. DevOps is a union of people, processes, platform capabilities and tools that work together to make continuous software delivery possible.

The term “DevOps” is derived from the words “Development” and “Operations”. Originally, the term emphasized the importance of addressing one of the most critical impediments to application delivery process – lack of collaboration and insufficient automation in the processes between Development and Operations teams. Creating smoother integration and more efficient collaboration between Development and Operations teams are still one of the main goals of a DevOps solution, but DevOps methodology and solutions have gradually evolved into something much broader.

Enterprise digital transformation efforts look to capitalize on the hybrid cloud, the API economy, microservices and containers. This transformation is powered by software, and therefore, drives an urgent focus on innovation in the delivery of business value.

It requires software development teams to make use of agile, modern DevOps practices, and deliver applications via an integrated pipeline, the backbone of the value stream. Using the capabilities offered by the IBM Z DevOps portfolio, z/OS development teams can adopt standard software delivery practices, modernize core applications, and deliver a seamlessly integrated pipeline.



You can use a single heterogeneous pipeline to orchestrate the development, integration, and deployment of an application across multiple target platforms and environments. This pipeline must handle all stages of the process, from build tasks such as compilation and binds to artifact packaging, deployment, and larger scale integration tests. The pipeline can enforce gating mechanisms at each stage, ensuring quality.

## Advantages

Including IBM z/OS applications in DevOps transformation supports product teams in a single way of working, which simplifies the process and breaks down silos. It brings common tools, reducing the skills problem, and enables new talent to work with IBM Z assets by using tools and processes that they already know.

Focusing on a single heterogeneous pipeline allows for enterprise-wide visibility of the entire software delivery value chain. The business can deploy parts of an application to the best platform for the type of application. A single pipeline also allows the delivery organization to focus on common compliance, security, and audit controls, which can reduce cost and allow for comprehensive control across the lifecycle.

## Lab introduction

This lab will walk you through a sample DevOps workflow that covers the code, build, test and deploy stages using a GitLab CI/CD pipeline for a traditional CICS application. The application code is managed in a Git-based repository on the GitLab platform where the multiple developers can work in parallel.

### **Acknowledgements:**

We would like to thank Wilbert Kho for generously allowing us to use the DevOps on Z Proof of Technology (PoT) workbook as a basis for this lab document.

## Scenario

As a developer, you have been assigned to work on a GitLab issue that documents a defect that is causing incorrect prices for the items to be returned from the CICS Catalog Manager application for the APIs. You choose to use a modern IDE like Wazi for VS Code or IBM Developer for z/OS (IDz) to check out the code from a Git repository, create a new branch to work on, and use the debugging tool to interactively trace the execution. After identifying the root cause, you make the code change and attempt a user build in your own workspace and perform an early integration test. After it has been successful, you then commit the changes to the master branch on the remote Git repo which automatically triggers a pipeline to build the entire application and deploy the changes to the SIT environment. Lastly, you run an integration test to show the correct cost has been rectified in the API response payload.

## Lab Requirements

This lab uses GitLab as the source code repository and leverages GitLab CI/CD's pipeline capability to automate the different tasks that are associated with the build. However, the GitLab repository is interchangeable with other Git providers such as GitHub and Bitbucket. The GitLab CI/CD pipeline orchestrator is also interchangeable, with Jenkins being another alternative that is widely used.

The IDE we use in the lab can be **Wazi for VS Code** or the Eclipse-based **IBM Developer for z/OS**. We have two different versions of this lab document to cover each flavor of the IDE. Both documents will complete the same task and steps are equivalent between the two, just with a different user interface. You can choose either one based on your preference.

With the VS Code option, there are a few extensions that have been preinstalled for supporting IBM Z application development.

- Git for Windows
- Zowe CLI and Zowe Explorer
- IBM Z Open Editor
- IBM Z Open Debug

On the z/OS side, we leverage IBM z/OS Debugger for interactive debugging sessions, IBM z/OS Dependency Based Build as the build framework, IBM Virtual Test Platform for early integration test, and of course, our target application Catalog Manager which is a CICS-supplied sample has been set up on CICS Transaction Server on z/OS.

We have also configured z/OS OpenSSH and RSEAPI for the integration with pipeline tasks.

We use z/OS Connect APIs to drive the CICS program for testing, but you can also use EGUI transaction which is the traditional 3270 BMS interface to run it too.

If you have any questions about the technology, please contact one of the lab instructors for assistance.

## **Lab Step Overview**

### **Part 1: Exploring the GitLab project and initiating a merge request for the issue**

Getting yourself familiar with the GitLab interface, confirm the defect with the API URL link attached. Then submit a merge request which creates a new branch to work in parallel just for this task.

### **Part 2: Create a clone of the new branch into your IDE**

In this step you will make a copy of the new branch into your IDE workspace. Navigate the project structure to understand the different components and configurations that are defined in the project as code.

### **Part 3: Debugging the application at statement level**

Enable the debugging option to intercept the program execution, then initiate another request to start the interactive debugging session, to exam the COBOL execution and identify the root cause for the issue.

### **Part 4: Make the code change**

---

In this part of the lab, you will learn how to explore the code editing capability in the IDE and fix the code.

## **Part 5: Run a user build to build the program in your own workspace**

In this part of the lab exercise, you will run a user build that will upload only the changed code with all its dependencies the z/OS environment and create a load module in your personal PDS dataset, with detail build result transferred back to your IDE for inspection.

## **Part 6: Early integration test of the application before deployment**

In this part of the lab exercise you will verify that your code change is now working and fixed the issue on the application level, without deploying into CICS.

## **Part 7: Commit the code and push to the remote Git repo to close the merge request**

In this step you are confident the code is ready, so you will commit it to your local Git repository and push it to the remote Git repo to submit the work you've done. This step will automatically trigger a full pipeline build.

You will dive into the different stages of the pipeline, review the build report that's published on the DBB webapp, deployment status and finally the integration test result from the API call.

## **Part 8: Summary**

This is a recap of the steps performed in this lab exercise.

## **Part 1: Exploring the GitLab project and initiating a merge request for the issue**

Getting yourself familiar with the GitLab interface, confirm the defect with the API URL link attached. Then submit a merge request which creates a new branch to work in parallel just for this task.

### **Log on to the GitLab dashboard**

1. From the desktop, double-click the Google Chrome icon to start the browser if it is not already running.



2. Click on the GitLab bookmark which opens the log in page.

A screenshot of the GitLab login page. At the top, there is a red header bar with the text "IBM CICS TS and DevOps workshop - October 2022". Below the header is the GitLab logo, which is a stylized orange fox head. The main page has a light gray background. On the left, there is a sidebar with the title "A complete DevOps platform" and a brief description: "GitLab is a single application for the entire software development lifecycle. From project planning and source code management to CI/CD, monitoring, and security." Below this, it says "This is a self-managed instance of GitLab.". The right side contains the login form. It includes fields for "Username or email" and "Password", a "Remember me" checkbox, a "Forgot your password?" link, and a large blue "Sign in" button. At the bottom of the form, there is a link "Don't have an account yet? Register now".

3. Use username `ibmdev` and password `Password` to login, you will see the dashboard for all the projects.

The screenshot shows the GitLab 'Projects' page. At the top, there's a navigation bar with a search bar labeled 'Search GitLab'. Below the navigation bar, the title 'Projects' is displayed, along with a 'New project' button. A filter bar allows filtering by 'Name'. The main area shows three projects:

- c** IBM Devs / **cics-catalog-manager** Owner DevOps CI/CD Pipeline Lab - CICS Catalog Manager. Last updated 1 minute ago.
- D** IBM Devs / **dbb-zappbuild** Owner. Last updated 1 month ago.
- Z** IBM Devs / **zcon-catalog-manager-api3** Owner z/OS Connect EE OpenAPI3 project for CICS Catalog Manager application. Last updated 1 week ago.

4. Click on the **cics-catalog-manager** project and you will be taken to the project homepage.

The screenshot shows the homepage of the 'cics-catalog-manager' project. The header includes the project name, a 'Project ID: 10' badge, and social sharing buttons for GitHub, Star, and Fork. Below the header, project statistics are listed: 20 Commits, 2 Branches, 0 Tags, and 5.3 MB Project Storage. The description 'DevOps CI/CD Pipeline Lab - CICS Catalog Manager' is present. The repository navigation bar shows 'master' and a '+' button. Below the repository info, a commit history is shown for 'Update COBOL/DFH0XVDS.cbl' by 'IBM Devs' 1 week ago. A list of available actions includes 'README', 'Apache License 2.0', 'CI/CD configuration', 'Add CHANGELOG', 'Add CONTRIBUTING', 'Auto DevOps enabled', 'Add Kubernetes cluster', and 'Configure Integrations'. A table at the bottom lists files with their last commit details:

Name	Last commit	Last update
.vscode	Initial commit for CICS & DevOps Workshop	2 weeks ago
ASM	Initial commit for CICS & DevOps Workshop	2 weeks ago
ASMCOPY	Initial commit for CICS & DevOps Workshop	2 weeks ago

## View the project issues and confirm the defect

5. Click on **Issues** from the sidebar, you might need to expand the side bar to see the full menu.

6. On the right, click the "Incorrect item cost" issue that was created earlier.

Item	Description	Cost	Order
0010	Ball.Pens.Black.24pk.....	999.99	-
0020	Ball.Pens.Blue.24pk.....	999.99	-
0030	Ball.Pens.Red.24pk.....	999.99	-
0040	Ball.Pens.Green.24pk.....	999.99	-
0050	Pencil.with.eraser.12pk.....	999.99	-
0060	Highlighters.Assorted.5pk.....	999.99	-
0070	Laser.Paper.28-lb.108.Bright.500/ream...	999.99	-
0080	Laser.Paper.28-lb.108.Bright.2500/case..	999.99	-
0090	Blue.Laser.Paper.20lb.500/ream.....	999.99	-
0100	Green.Laser.Paper.20lb.500/ream.....	999.99	-
0110	IBM.Network.Printer.24.-.Toner.cart.....	999.99	-
0120	Standard.Diary;.Week.to.view.8.1/4x5.3/4	999.99	-
0130	Wall.Planner;.Erasable.36x24.....	999.99	-

7. Use the link included in the description to try the Catalog Manager APIs which shows all the item costs were wrongly marked as **999.99**

8. Go back to **GitLab** browser tab and scroll down the issue and click the **Create Merge Request** button to start a new branch called ***n-incorrect-item-cost*** (branch name is generated automatically and it can be a different from the screenshot) and a draft merge request that will be used to merge the work into the master branch when complete.

## New merge request

From 5-incorrect-item-cost into master [Change branches](#)

Title (required)

Draft: Resolve "Incorrect item cost"

Remove the `Draft` prefix from the title to allow this merge request to be merged when it's ready.

Add [description templates](#) to help your contributors to communicate effectively!

Description

[Write](#) [Preview](#)

---

Closes #5

B I § |≡ </> ⚡ |≡ |≡ |≡ |≡ |≡ |≡

---

Supports [Markdown](#). For [quick actions](#), type /.

Attach a file

9. Click **Assign to me** link for assignee and click the **Create merge request** button.

Assignee  
IBM Devs

Reviewer  
Unassigned

Milestone  
Milestone

Labels  
Labels

Merge options

Delete source branch when merge request is accepted.

Squash commits when merge request is accepted. (?)

**Create merge request** **Cancel**

- 10.** Now you have created a new branch in the remote Git repository, you can now start working on this branch which is isolated from any other development activities happening in the same repository. When your work is done, you will submit this merge request (which currently sits in draft state) to merge the changes to the master branch.

IBM Devs > cics-catalog-manager > Merge requests > !6

## Draft: Resolve "Incorrect item cost"

**Edit**

**Code** ▼

⋮

Open IBM Devs requested to merge [5-incorrect-item-cost](#) into [master](#) just now

[Overview](#) 0   [Commits](#) 0   [Pipelines](#) 1   [Changes](#) 0

Closes #5

This merge request contains no changes.

Use merge requests to propose changes to your project and discuss them with your team. To make changes, push a commit or edit this merge request to use a different branch. With [CI/CD](#), automatically test your changes before merging.



**Create file**

0

0

Oldest first ▼

Show all activity ▼

[Write](#) [Preview](#)

B I § |≡ </> ⚡ |≡ |≡ |≡ |≡ |≡ |≡

### Note:

What is Git?

Git is a free and open source software for distributed version control: tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows (thousands of parallel branches running on different systems).

Git is very popular in the distributed world. In early 2017, Rocket Software ported Git into the mainframe – with the necessary checks to handle EBCDIC to UTF-8 conversions and vice-versa.

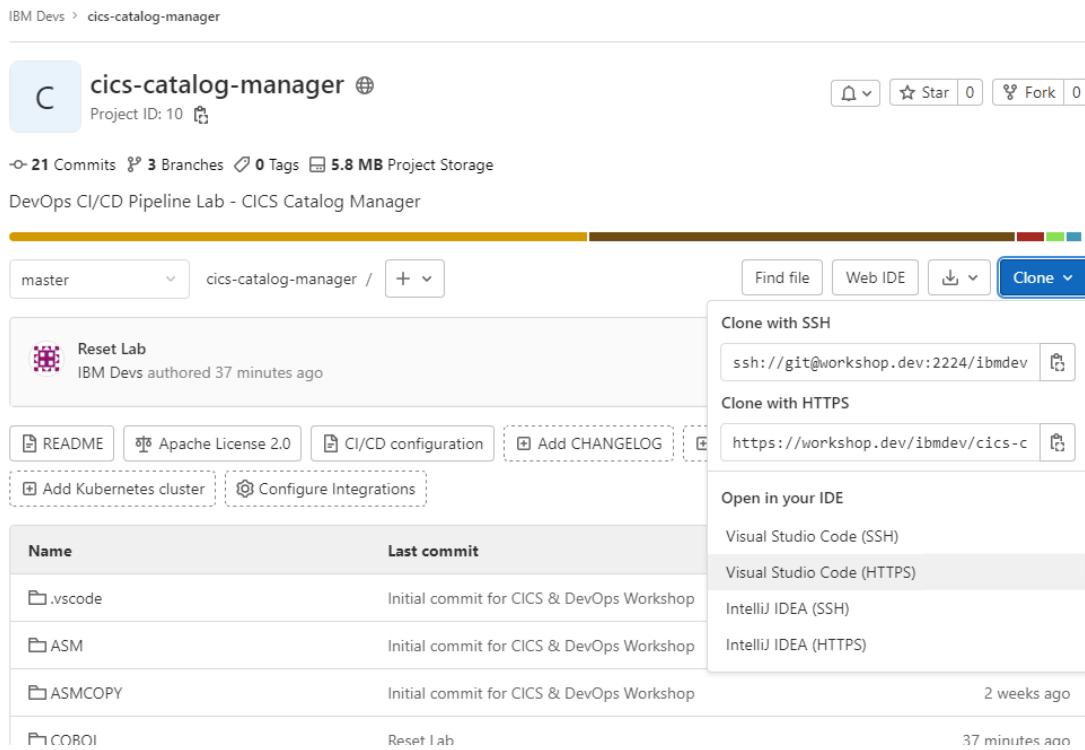
Git is not to be confused with GitLab or GitHub, which are commercial DevOps offerings consisting of Git-based SCM repositories with added features like bug tracking, continuous integration and team collaboration tools.

## **Part 2: Create a clone of the new branch into your IDE**

In this step you will make a copy of the new branch into your IDE workspace. Navigate the project structure to understand the different components and configurations that are defined in the project as code.

### **Clone the git repository into your workstation using VS Code**

- 1. Go back to the project homepage in **GitLab** by clicking on the  icon at the top of the navbar.
- 2. Click on the **Clone** button on the right and select **Open in your IDE, Visual Studio Code (HTTPS)** option.

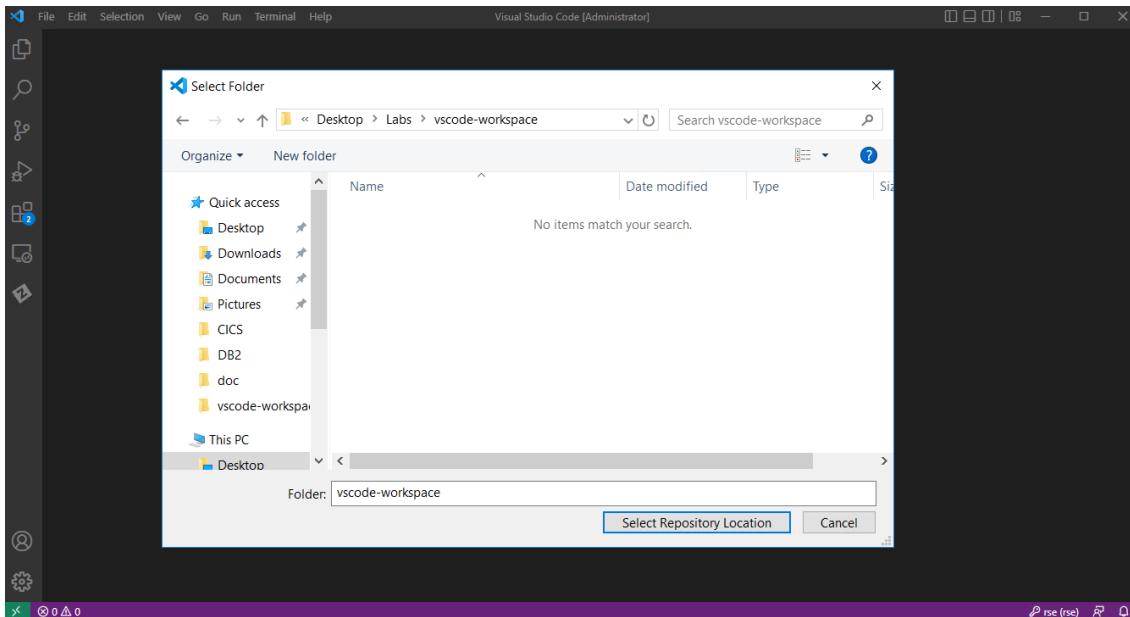


The screenshot shows the GitLab project page for 'cics-catalog-manager'. At the top, there's a navigation bar with a 'C' icon. Below it, the project name 'cics-catalog-manager' and a 'Project ID: 10' link. To the right are icons for issues (down arrow), stars (star), and forks (fork). The main area shows commit history with a 'master' dropdown, a search bar, and buttons for 'Find file', 'Web IDE', and 'Clone'. A dropdown menu is open under 'Clone', listing options for SSH and HTTPS, and specifically highlighting 'Open in your IDE' with 'Visual Studio Code (HTTPS)' selected. Below the dropdown, a table lists recent commits, including 'Initial commit for CICS & DevOps Workshop' and 'Reset Lab'.

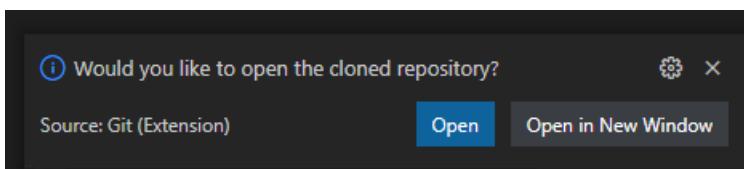
Name	Last commit
.vscode	Initial commit for CICS & DevOps Workshop
ASM	Initial commit for CICS & DevOps Workshop
ASMCOPY	Initial commit for CICS & DevOps Workshop
COROI	Reset Lab

- \_\_\_3. VS Code should automatically open (If prompted, click **Open** to confirm the operation).
- \_\_\_4. In the **Select Folder** window, Open the **Desktop → Labs → vscode-workspace** folder on and click **Select Repository Location** button. This should be an empty directory.

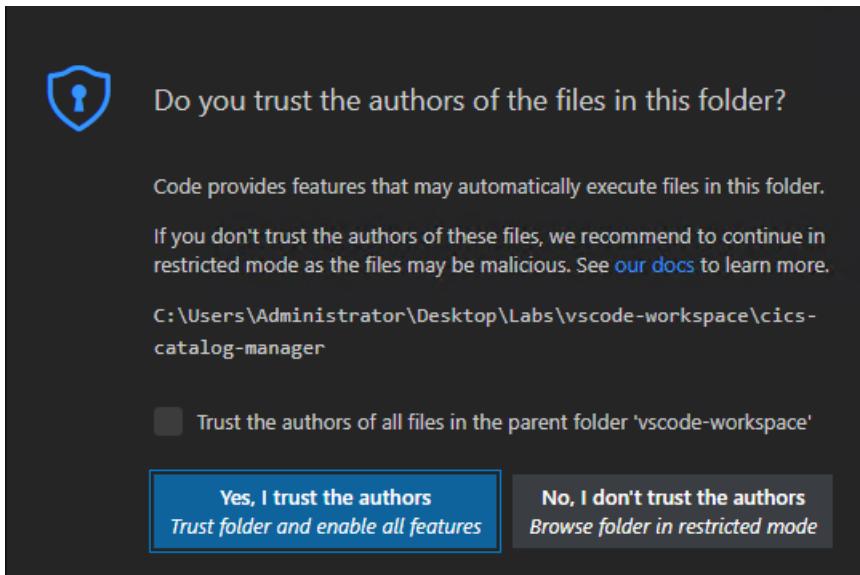
(The full path is C:\Users\Administrator\Desktop\Labs\vscode-workspace)



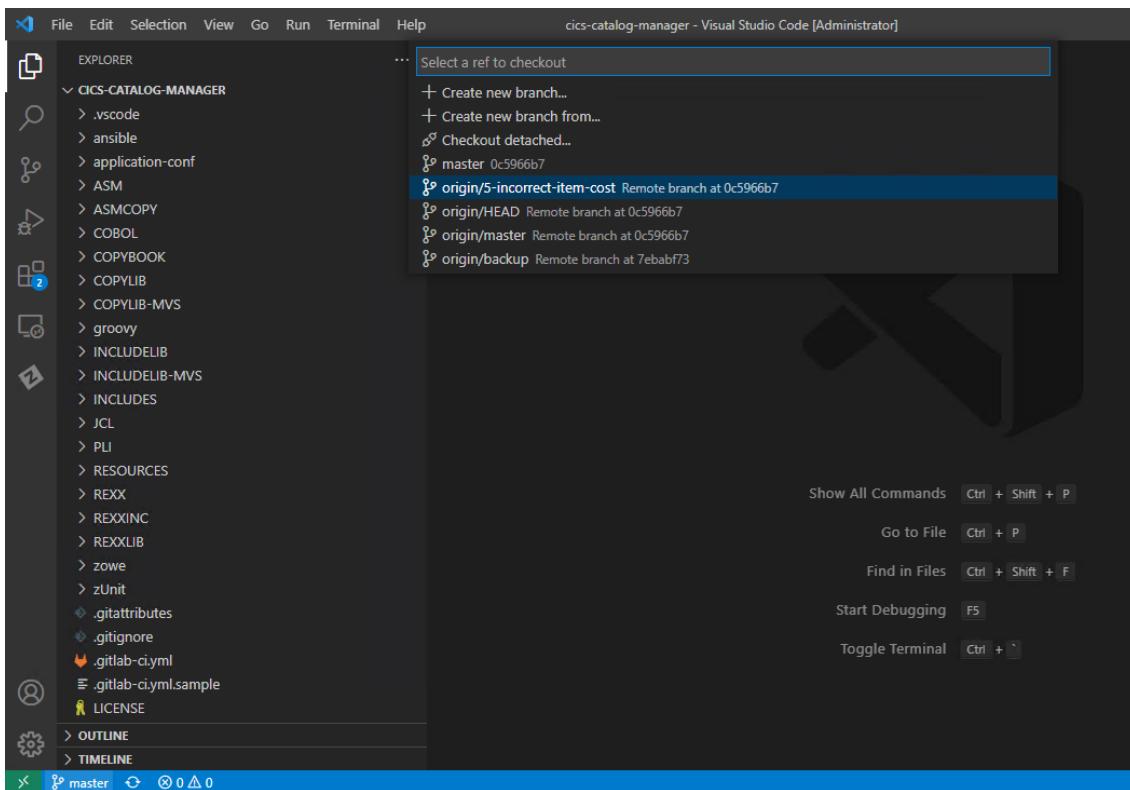
- \_\_\_5. Click **Open** button at the bottom pop-up dialog to open the local cloned project directory.



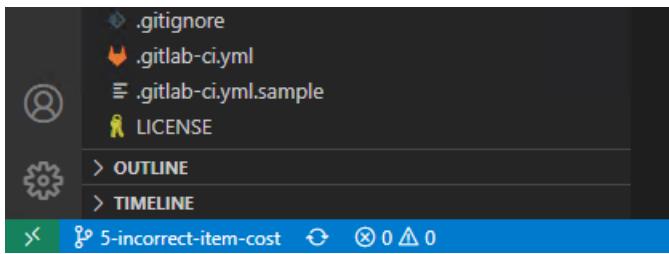
- 
- \_\_\_ 6. Click **Trust** button to trust the source in the project.



- \_\_\_ 7. VS Code has now created a local copy of the Git repository, now very **importantly** as you want to work on the new branch rather than the master, select **origin/n-incorrect-item-cost** to checkout as the current branch to work on.



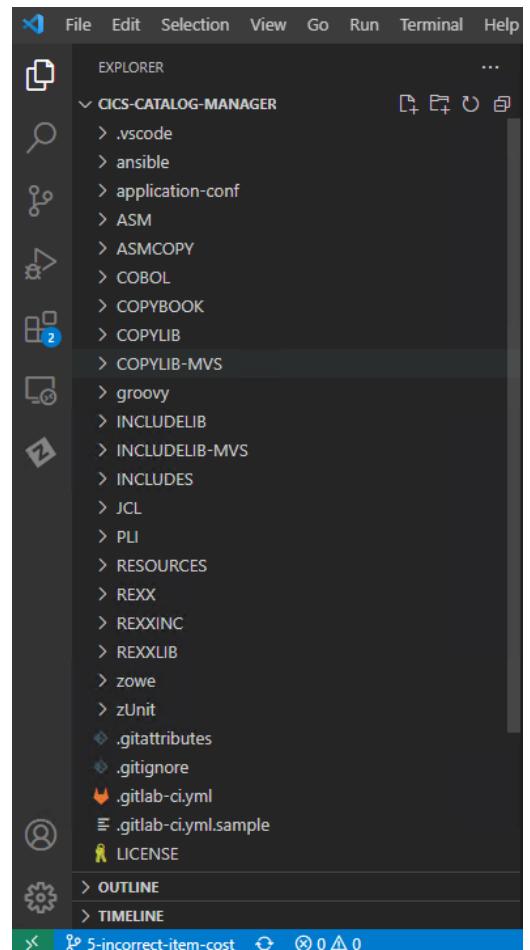
- 
- \_\_\_ 8. **Make sure** the bottom status bar shows the new branch name which means you are now working in an isolated code branch now rather than the main development stream.



## Understanding the structure of the z application project

9. Feel free to navigate and explore the content of the project in VS Code and get familiar with the way it's structured.
10. Some of the key elements in the project includes:

- **.vscode** contains the project specified settings for the VS Code environment so that all the team members will automatically inherit IDE settings and project specific configurations e.g. copybook resolving search order, file type mapping, also environmental settings like remote z/OS debugging port numbers and user-build configurations.
- **application-conf** contains the project-specific setting for zAppBuild framework, which leverages IBM Dependency Build for z/OS for the build, it includes configuration on how to build the project, more specifically for our project it defines things like the folders of the COBOL source code, copybooks etc. And impact resolution rules, and last but not least the compiler options for each type of the program as well as the link editor options.
- The application source is located in **language-specific folders** e.g. Assembler, COBOL and PLI, JCL, REXX. In our lab, we only focus on COBOL and COPYBOOK, but you can see how they can be organized for any real project.
- **.gitlab-ci.yml** file is the main CI/CD pipeline definition file for GitLab, which will be actioned whenever changes are committed. This is a YAML document that defines the different stages of the DevOps pipeline and will be executed automatically by the runner when triggered. Similarly, if you use Jenkins for pipeline orchestration, you will have a **Jenkinsfile** located in the project repository managed by the SCM too.



- 
- \_\_\_ **11.** As you can see the whole concept for the way the project is structured is to have all the configuration managed in the repository like source code (configuration-as-code) so that it becomes portable, and the moment developer clone the project into the IDE they will have a workspace ready for all the tasks they need.
  - \_\_\_ **12.** Without making any changes, you can close any open files now.

**Note:**

IBM Dependency Based Build (DBB) is an intelligent build system for traditional z/OS applications written in languages such as COBOL and PL/I that allows the analysis of build dependencies between objects.

The goal of DBB is to provide automation capabilities that can be used on z/OS. IBM DBB is a standalone framework (it does not require a specific source code manager or automation tool) to simplify the process of building code on z/OS based on a modern scripting language.

z/OS development teams have the freedom to choose a modern software configuration management (SCM) tool, such as Git, and continuous integration tools, such as Jenkins or GitLab, to build traditional z/OS applications written in COBOL or PL/I.

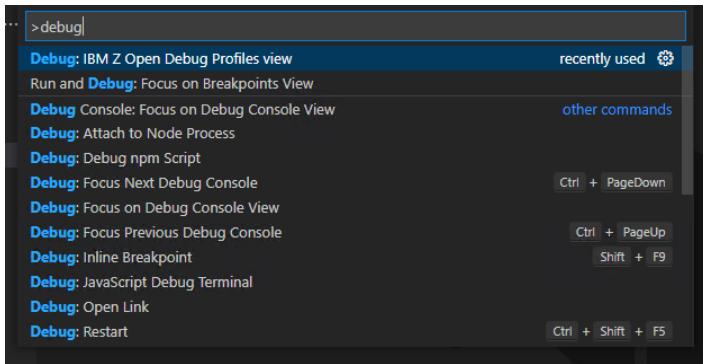
DBB allows you to standardize DevOps processes and practices across multiple platforms

## ***Part 3: Debugging the application at statement level***

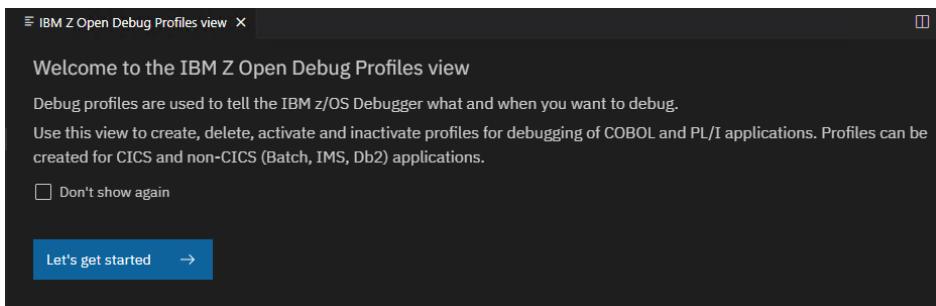
We know our program has some defects with the incorrect item cost, but the best way to identify the root cause is to use the debugger and trace the program line-by-line when in execution. In this part of the lab we will use the z Open Debug to connect to IBM z/OS Debugger, set up a debug profile which intercept any call to the main catalog program DFH0XCMN and “park” its execution, so that developer can use VS Code to connect to the "parked" debug session to inspect and control the execution.

### **Creating the Debug Profile**

- \_\_\_ **13.** In the VS Code window, press **Control-Shift-p** to bring up the **command palette** and start typing **debug profile** and select **Debug: IBM Z Open Debug Profiles** view from the list.

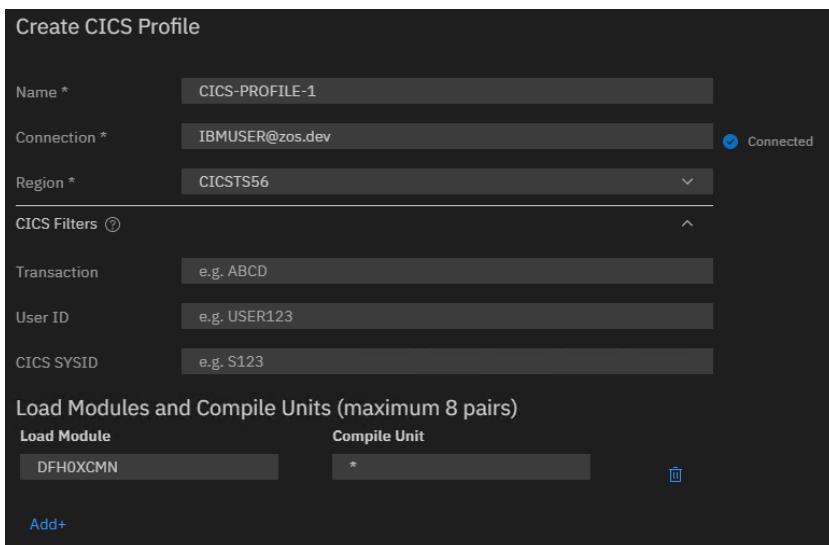


14. In the welcome screen read the description and click **Let's get started** button.



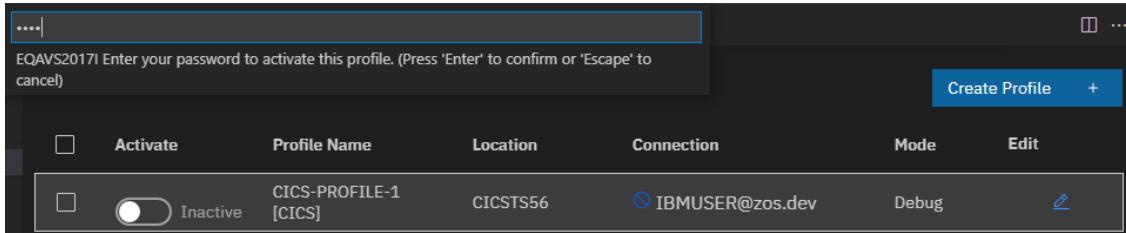
15. Next step click **Create Profile** button and select **CICS**

16. On the next screen you will be able to select what you want to debug. you will notice that it's now connected to the z/OS and listing all available CICS regions, select **CICSTS56**. In the CICS filters section you can define a few criteria of the CICS application you want to debug, things like transaction ID, user ID, SYSID, as well as load modules or a combination. We will click **Add+** button under **Load Modules and Compile Units** and specify load module name **DFH0XCMN** so that anytime our catalog manager program is invoked, it will be suspended and parked aside for you to start take control and debug interactively.



17. Leave all the other fields as default and click **Create** button to finish, you will be taken back to the previous page with **CICS-PROFILE-1** defined with **inactive** status.

18. Click on the **Activate** toggle switch, you will be asked to type your password, type **SYS1** in the pop-up window and press **Enter** and now it should be successfully activated.



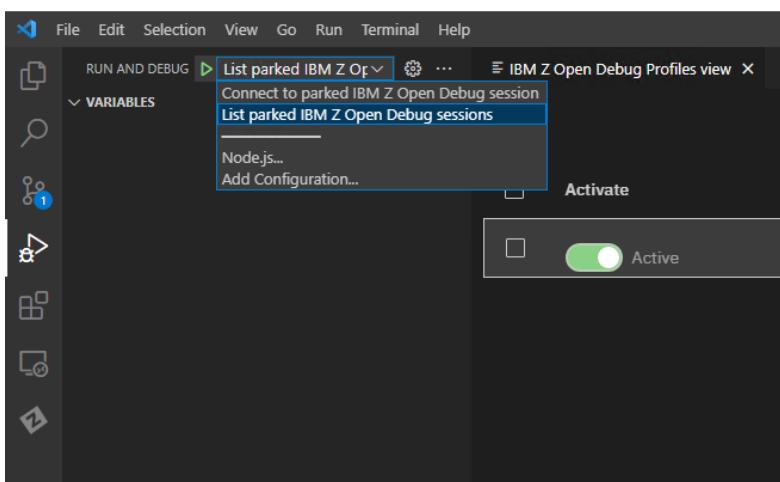
### Invoke the program to start debugging

19. Now that your debug profile for **DFH0XCMN** is activated. Go back to the browser and trigger the API call once again using the URL:

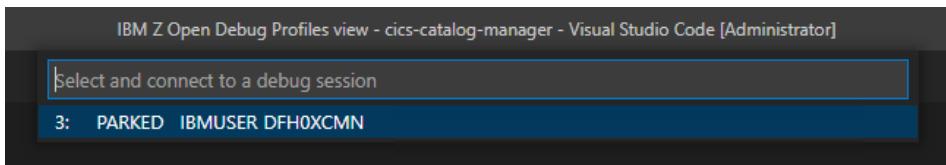
```
https://workshop.dev:9443/items?startItemID=0
```

You will notice the page is loading as the API is not responding this time since the backend program has been intercepted.

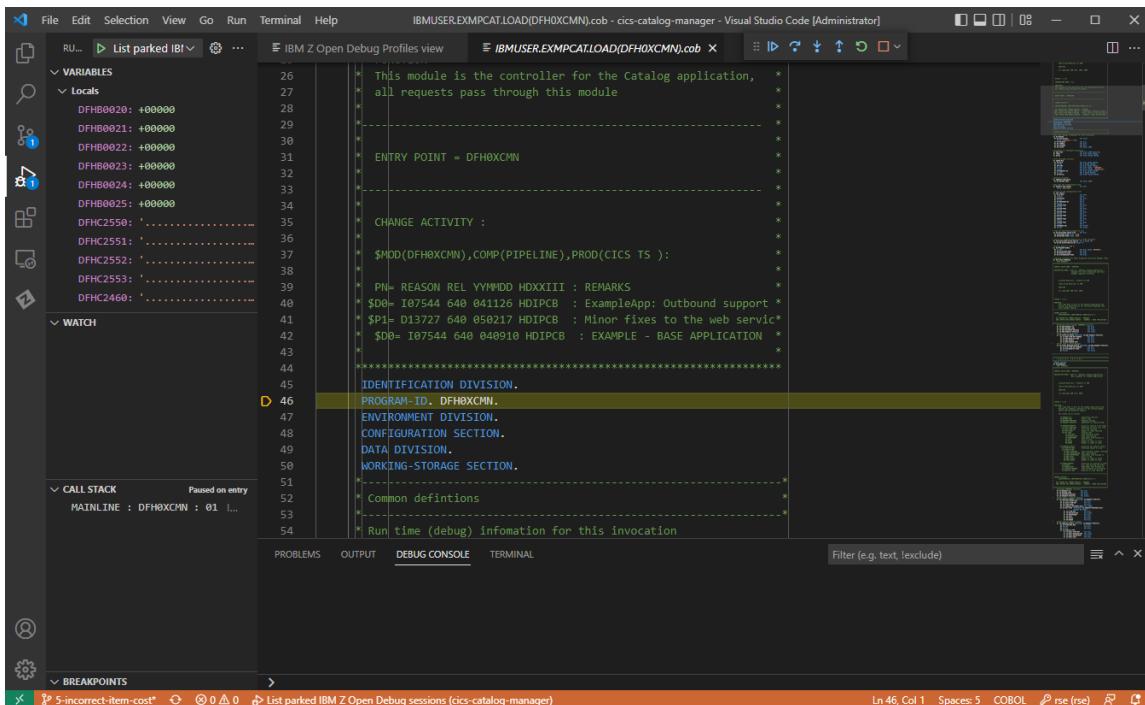
20. Go back to the VS Code and click **Debug** icon on the navbar, and at the top dropdown list choose **List parked IBM Z Open Debug sessions** and click the green **go** button.



21. There should be a pop-up window list the current debug session for **DFH0XCMN**, click on the session name to start the debug session.



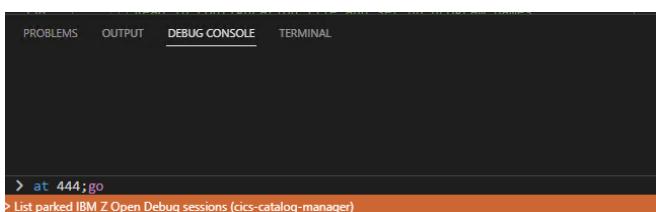
22. In the debugging session, you could find source code inside the editor, see variables, call stack and watch expressions as you step. Similar to the debugging experience for other programming languages, you can step through a running program, check variable values, set breakpoints etc.



23. You can explore the interface by:

- Try to press **F10** key multiple time to step over the program line by line.
- Hover the mouse over any variable names to monitor its real time value, or right-click to add into watch window.
- Click in the area before the line number you can set or clear breakpoint on that line.

24. type **at 444;go** and **Enter** in the **Debug Console** at the bottom of the screen to set a breakpoint on line 444 and continue.



25. The program will resume execution and pause again just before linking into another CICS program that handles the catalog inquire function.

```

439      *=====
440      * Procedure to link to Datastore program to inquire
441      *   on the catalog data
442      *=====
443      CATALOG-INQUIRE.
444      MOVE 'EXCATMAN: CATALOG-INQUIRE' TO CA-RESPONSE-MESSAGE
445      EXEC CICS LINK PROGRAM(WS-DATASTORE-PROG)
446      COMMAREA(DFHCOMMAREA)
447      END-EXEC
448      EXIT.

```

26. Press F11 or click **Step-Into** button from the top bar to trace into the subprogram, and we will see we are now debugging in the program DFH0XVDS.

```

28      * ENTRY POINT = DFH0XVDS
29
30      *
31      * CHANGE ACTIVITY :
32
33      *MOD(DFH0XVDS),COMP(SAMPLES),PROD(CICS TS):
34
35      * PN= REASON REL YYMMDD HDXXIII : REMARKS
36      * $P0= D13727 640 050217 HDIPCB : Minor fixes to the web service
37      * $P1= D20555 660 080415 HDFFCMS : Typos in Web Service Example
38      * $D0= I07544 640 040910 HDIPCB : EXAMPLE - BASE APPLICATION
39
40      ****
41      IDENTIFICATION DIVISION.
42      PROGRAM-ID. DFH0XVDS.
43      ENVIRONMENT DIVISION.
44      CONFIGURATION SECTION.
45      DATA DIVISION.
46      WORKING-STORAGE SECTION.
47
48      * Common definitions
49
50      * Run time (debug) information for this invocation
51      01 WS-HEADER.
52          03 WS-EYECATCHER      PIC X(16)
53                               VALUE 'DFH0XVDS-----WS'.
54          03 WS-TRANSID         PIC X(4).
55          03 WS-TERMINAL        PIC X(4).
56          03 WS-TASKNUM        PIC 9(7).

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Filter (e.g. text, exclude)

→ at 444;go

COMMAND:go;  
Program was stopped due to line/statement breakpoint at statement 444.

27. Type `at change WS-COST` and **Enter** in the Debug Console at the bottom of the screen to define a dynamic breakpoint only when the content of WS-COST variable is modified.

Press F5 or the **continue** button about 3 times until reaching **line 401**.

Where you can clearly see the **WS-COST** has been overwritten with a value 999.99 by a **MOVE** statement. This is the root cause of the defect which we will fix later.

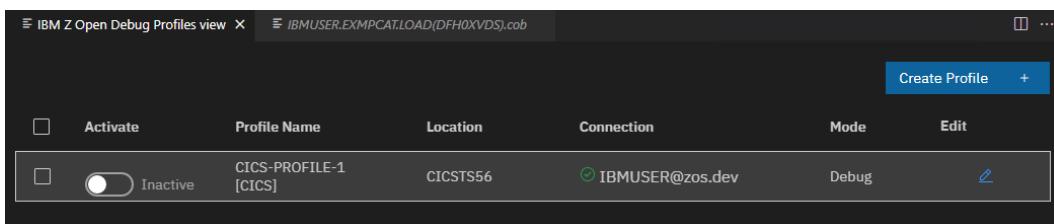
```

394
395 v EVALUATE WS-RESPONSE-CODE
396 v WHEN DFHRESP(NORMAL)
397
398 v * MOVE WS-LOOP-COUNTER TO WS-RECORD-COUNT
399   this is a bug - start
400 v * MOVE 999.99 TO WS-COST IN WS-CAT-ITEM
401   this is a bug - end
402 D MOVE WS-CAT-ITEM TO CA-CAT-ITEM(WS-LOOP-COUNTER)
403
404 MOVE WS-RECORD-COUNT TO CA-ITEM-COUNT
405 MOVE WS-CURRENT-ITEM-REF TO CA-LAST-ITEM-REF

```

## Stop the debug session and deactivate the debug profile

28. Now we know the root cause of the defect. We now stop the debugging session by press **Shift-F5** or click the stop button , now you are disconnected the debugging session and you should also be able to see the API response now returned.
29. IMPORTANT. Don't forget to deactivate the debug profile in the **IBM Z Open Debug Profile view**. Otherwise, the program will be suspended for debugging again when we later test our program. When deactivating the profile, type **SYS1** in the password window.



### Note:

IBM Z Open Debug is a remote debug interface to be used with z/OS Debugger to debug z/OS COBOL, PL/I, and assembler applications in Wazi for VS Code and Wazi for Dev Spaces IDEs.

IBM z/OS Debugger is a host component that supports various debug interfaces, like the Eclipse and Visual Studio Code IDEs.

## Part 4: Make the code change

In this part of the lab, you will learn how to leverage the code editing capability in VS Code to fix the code.

### Open the source code in the editor

1. Click on the top **Explorer** button in the left navigation bar to go back to the local project we just cloned.

2. Expand the COBOL folder and click **DFH0XVDS.cbl** to open the source code in the editor.

```

CBL CICS('COBOL3') APOST
* MODULE NAME = DFH0XVDS
* DESCRIPTIVE NAME = CICS TS (Samples) Example Application -
* VSAM Data Store
*
* Licensed Materials - Property of IBM
* "Restricted Materials of IBM"
* 5655-Y04
* (C) Copyright IBM Corp. 2004, 2021"
* DAT version
* STATUS = 7.2.0
* TRANSACTION NAME = n/a
* FUNCTION =
* This accesses the VSAM file for the example application
* to perform reads and updates of the catalog

```

3. Move the cursor to line **290** and press **Ctrl- /** to comment out this line, alternatively you can also put an asterisk on column 8 which is indicated by the tab stop indicator in the editor.

4. Press **Ctrl-S** to save the file to complete the bug fix.

## Experimenting with other cool editing features

5. Browse code by using the **Outline view**: With the COBOL program **DFH0XVDS.cbl** open, use the **Outline view** to explore the structure of the program. This view provides a complete sortable and interactive "table of contents" for the program's code. You can click on the arrow icons at the

left of each header to expand them, and quickly jump to any part of your code by clicking on it in the Outline view.

- In the **Outline** view (located in the lower left of the VS Code window), expand the headers **PROGRAM: DFH0XVDS > PROCEDURE DIVISION > MAINLINE SECTION.**
    - (You might have to click the Outline tab to activate this view.)
  - Click on the header "**CATALOG-INQUIRE**" to jump to this part of the code in the editor window.

EXPLORER

CICS-CATALOG-MANAGER

- DFH0XST2.cbl
- DFH0XVDS.cbl
- DFH0XWOD.cbl
- SAM1.cbl
- SAM1LIB.cbl

OUTLINE

- ({}) PROGRAM: DFH0XVDS
  - IDENTIFICATION DIVISION.
  - ENVIRONMENT DIVISION.
  - DATA DIVISION.
  - PROCEDURE DIVISION.
    - MAINLINE SECTION.
      - MAINLINE-EXIT.
      - WRITE-ERROR-MESSAGE.
      - CATALOG-INQUIRE. **2**

DFH0XVDS.cbl X

COBOL > DFH0XVDS.cbl > {} PROGRAM: DFH0XVDS > PROCEDURE DIVISION.

```
227 END-EXEC.  
228 EXIT.  
229 *=====  
230 * Procedure to link to Datastore program to inquire  
231 * on the catalog data  
232 *=====  
233 CATALOG-INQUIRE.  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1198  
1199  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1279  
1280  
1281  
1282  
1283  
1284  
1285  
1286  
1287  
1288  
1289  
1289  
1290  
1291  
1292  
1293  
1294  
1295  
1296  
1297  
1298  
1299  
1299  
1300  
1301  
1302  
1303  
1304  
1305  
1306  
1307  
1308  
1309  
1309  
1310  
1311  
1312  
1313  
1314  
1315  
1316  
1317  
1318  
1319  
1319  
1320  
1321  
1322  
1323  
1324  
1325  
1326  
1327  
1328  
1329  
1329  
1330  
1331  
1332  
1333  
1334  
1335  
1336  
1337  
1338  
1339  
1339  
1340  
1341  
1342  
1343  
1344  
1345  
1346  
1347  
1348  
1349  
1349  
1350  
1351  
1352  
1353  
1354  
1355  
1356  
1357  
1358  
1359  
1359  
1360  
1361  
1362  
1363  
1364  
1365  
1366  
1367  
1368  
1369  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457  
1458  
1459  
1459  
1460  
1461  
1462  
1463  
1464  
1465  
1466  
1467  
1468  
1469  
1469  
1470  
1471  
1472  
1473  
1474  
1475  
1476  
1477  
1478  
1479  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1489  
1490  
1491  
1492  
1493  
1494  
1495  
1496  
1497  
1498  
1499  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508  
1509  
1509  
1510  
1511  
1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1559  
1560  
1561  
1562  
1563  
1564  
1565  
1566  
1567  
1568  
1569  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1659  
1660  
1661  
1662  
1663  
1664  
1665  
1666  
1667  
1668  
1669  
1669  
1670  
1671  
1672  
1673  
1674  
1675  
1676  
1677  
1678  
1679  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716  
1717  
1718  
1719  
1719  
1720  
1721  
1722  
1723  
1724  
1725  
1726  
1727  
1728  
1729  
1729  
1730  
1731  
1732  
1733  
1734  
1735  
1736  
1737  
1738  
1739  
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749  
1749  
1750  
1751  
1752  
1753  
1754  
1755  
1756  
1757  
1758  
1759  
1759  
1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1769  
1770  
1771  
1772  
1773  
1774  
1775  
1776  
1777  
1778  
1779  
1779  
1780  
1781  
1782  
1783  
1784  
1785  
1786  
1787  
1788  
1789  
1789  
1790  
1791  
1792  
1793  
1794  
1795  
1796  
1797  
1798  
1799  
1799  
1800  
1801  
1802  
1803  
1804  
1805  
1806  
1807  
1808  
1809  
1809  
1810  
1811  
1812  
1813  
1814  
1815  
1816  
1817  
1818  
1819  
1819  
1820  
1821  
1822  
1823  
1824  
1825  
1826  
1827  
1828  
1829  
1829  
1830  
1831  
1832  
1833  
1834  
1835  
1836  
1837  
1838  
1839  
1839  
1840  
1841  
1842  
1843  
1844  
1845  
1846  
1847  
1848  
1849  
1849  
1850  
1851  
1852  
1853  
1854  
1855  
1856  
1857  
1858  
1859  
1859  
1860  
1861  
1862  
1863  
1864  
1865  
1866  
1867  
1868  
1869  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889  
1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1939  
1940  
1941  
1942  
1943  
1944  
1945  
1946  
1947  
1948  
1949  
1949  
1950  
1951  
1952  
1953  
1954  
1955  
1956  
1957  
1958  
1959  
1959  
1960  
1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997  
1998  
1999  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017  
2018  
2019  
2019  
2020  
2021  
2022  
2023  
2024  
2025  
2026  
2027  
2028  
2029  
2029  
2030  
2031  
2032  
2033  
2034  
2035  
2036  
2037  
2038  
2039  
2039  
2040  
2041  
2042  
2043  
2044  
2045  
2046  
2047  
2048  
2049  
2049  
2050  
2051  
2052  
2053  
2054  
2055  
2056  
2057  
2058  
2059  
2059  
2060  
2061  
2062  
2063  
2064  
2065  
2066  
2067  
2068  
2069  
2069  
2070  
2071  
2072  
2073  
2074  
2075  
2076  
2077  
2078  
2079  
2079  
2080  
2081  
2082  
2083  
2084  
2085  
2086  
2087  
2088  
2089  
2089  
2090  
2091  
2092  
2093  
2094  
2095  
2096  
2097  
2098  
2099  
2099  
2100  
2101  
2102  
2103  
2104  
2105  
2106  
2107  
2108  
2109  
2109  
2110  
2111  
2112  
2113  
2114  
2115  
2116  
2117  
2118  
2119  
2119  
2120  
2121  
2122  
2123  
2124  
2125  
2126  
2127  
2128  
2129  
2129  
2130  
2131  
2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147  
2148  
2149  
2149  
2150  
2151  
2152  
2153  
2154  
2155  
2156  
2157  
2158  
2159  
2159  
2160  
2161  
2162  
2163  
2164  
2165  
2166  
2167  
2168  
2169  
2169  
2170  
2171  
2172  
2173  
2174  
2175  
2176  
2177  
2178  
2179  
2179  
2180  
2181  
2182  
2183  
2184  
2185  
2186  
2187  
2188  
2189  
2189  
2190  
2191  
2192  
2193  
2194  
2195  
2196  
2197  
2198  
2199  
2199<br
```

6. Code completion: Use IBM Z Open Editor code completion feature to view and select from a list of commands, defined variable names, and code snippets when you start typing a command, variable, or paragraph name.

- At line 242 of the COBOL program DFH0XVDS.cbl, take note of the variable "CA-CAT-ITEM", and then place your cursor at the end of the line and press "Enter" to start a new line.
  - Place your cursor at column 15 of the new line (if it is not already there) and begin CCA typing the variable name: "CA-C". A selection list automatically appears with matching commands, variable names, and code snippets, allowing you to quickly and accurately complete what you were typing by simply choosing the relevant item from the list.
    - The list dynamically narrows down if you continue typing the variable name (for example, ""CA-CAT") while it is displayed.
    - You can use the up and down arrow keys to move up and down the displayed list.

```

242          INITIALIZE CA-CAT-ITEM(WS-LOOP-COUNTER)
243          CA-CAT
244          END-PERFO [e] CA-CAT-ITEM           CA-CAT-ITEM
245          [e] CA-CAT-ITEM IN CA-INQUIRE-REQUEST
246          MOVE 'EXD' [e] CA-CHARGE-DEPT
247          [e] CA-CHARGE-DEPT IN CA-ORDER-REQUEST
248          MOVE CA-L [e] CA-COST IN CA-CAT-ITEM

```

7. Preview copybooks: Preview the contents of a copybook in the COBOL program.

- At line 129 in DFH0XVDS.cbl, hover your mouse cursor over the copybook name "DFH0XCP1" in the COPY statement to view the copybook preview.
- You can also open the copybook in a separate editor by pressing **Ctrl+Click** (Windows) or **Cmd+Click** (Mac) on the hover pop-up. (You can also do this by scrolling to the bottom of the hover pop-up and clicking "Follow link".)

```

127          LINKAGE SECTION.
128          01 DFHCOMMAREA.
129          COPY DFH0XCP1.
130
131          ****
132          * P R O          * ***** CONTROL BLOCK NAME = DFH0XCP1
133          ****
134          PROCEDURE        * DESCRIPTIVE NAME = CICS TS (Samples) Example Application -
135                      * Main copybook for example application
136          *
137          MAINLINE
138          *
139          *----- * Licensed Materials - Property of IBM
140          * Common c
141          *----- * "Restricted Materials of IBM"
142          * initiali
143          DISPLAY 'Hello World!'.

```

8. Hover for declaration: View a data declaration by hovering over the variable or paragraph name.

- At line 144, hover your cursor over the variable name "WS-HEADER". Note the hover pane that appears, showing you the working storage definition or DCL definition and the parent group of the variable name.
  - Note: To quickly jump to the variable's declaration, you can click the "WS-HEADER" variable to select it, and then press F3 (or right-click and select "Go to Definition").

```

142          * initialize working storage variables
143          DISPLAY 'Hello World!'.
144          INITIALIZE WS-HEADER.
145          INITIALIZE 01 WS-HEADER.
146          03 WS-EYECATCHER PIC X(16) VALUE 'DFH0XVDS-----WS'.
147          * set up general
148          MOVE EIBTRN
149          MOVE EIBTRM
150          MOVE EIBTAS

```

9. Close the file without saving it.

**Tech-Tip:**

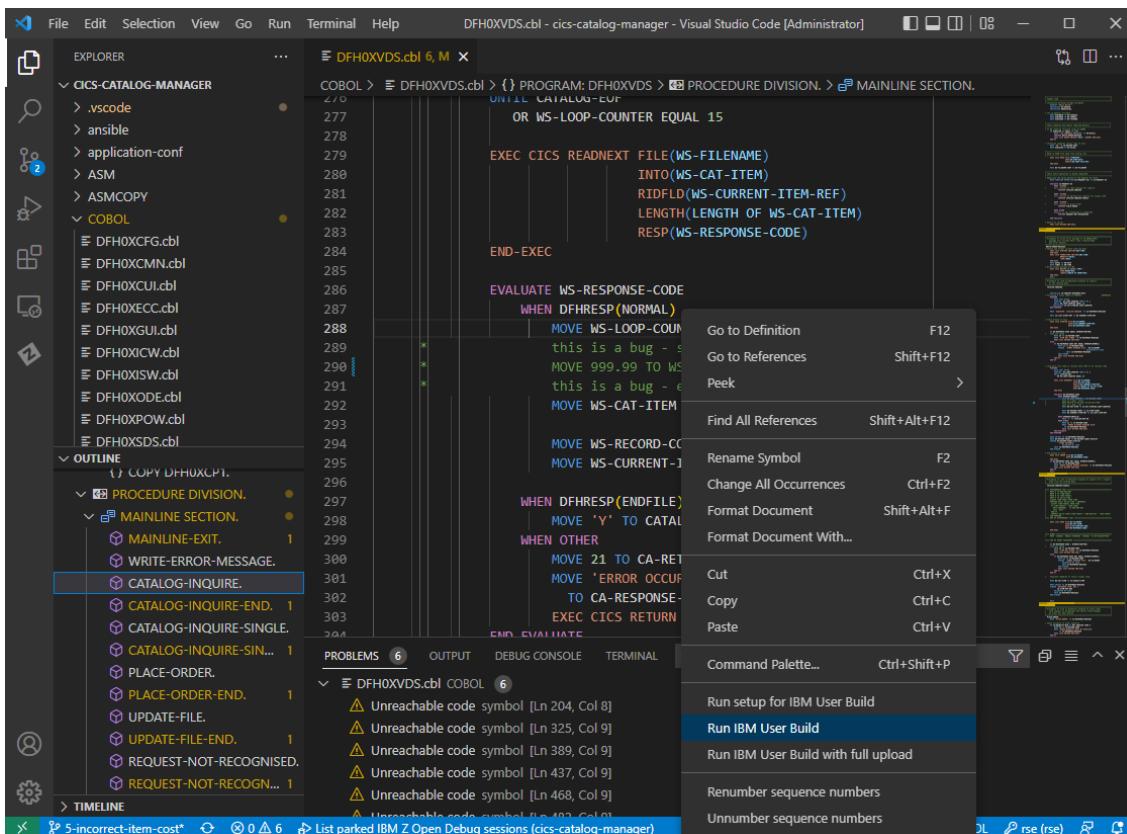
For additional editing features, you can check out the IBM Z Open Editor documentation:

<https://www.ibm.com/docs/en/cloud-paks/z-modernization-stack/2022.3?topic=files-making-cobol-pli-code-changes>

## Part 5: Run a user build to build the program in your own workspace

In this part of the lab exercise, you will run a user build that will upload only the changed code with all its dependencies to the z/OS environment and use IBM DBB to build a load module in your personal PDS load library dataset. And detail build report is then transferred back to your workstation IDE for review.

1. With the **DFH0XVDS.cbl** program saved, right-click within the editor area and choose "Run IBM User Build"



2. Switch to the **Output** panel down below and monitor the output information for **IBM User Build process** until you see the **Build State: CLEAN**

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under 'CICS-CATALOG-MANAGER' with files like DFH0XCFG.cbl, DFH0XCMN.cbl, DFH0XCUI.cbl, etc.
- Editor:** Displays the COBOL code for DFH0XVDS.cbl, specifically the MAINLINE SECTION. The code includes logic for reading a file, evaluating responses, and moving data between WS-CAT-ITEM and CA-CAT-ITEM.
- Output Panel:** Shows the build logs for 'IBM User Build'. The logs indicate the build process, including invoking build scripts (groovy, LinkEdit.groovy), building files (COBOL/DFH0XVDS.cbl), writing build reports, and finally stating 'Build State : CLEAN'.
- Bottom Status Bar:** Shows the current line (Ln 290), column (Col 51), spaces (Spaces: 5), encoding (UTF-8), line feed (LF), COBOL, and the 'rse (rse)' command.

3. Scroll up if you want to understand the process, take note of the high-level qualifier of the user build dataset library which will be used in the next step which is **IBMUSER.USERBLD**

```
BGZTK0108I Process, 65888: /u/ibmuser/userbuild/dbb-zappbuild/build.groovy --
userBuild --workspace /u/ibmuser/userbuild --application . --hlq
IBMUSER.USERBLD --outDir /u/ibmuser/userbuild/log --dependencyFile
/u/ibmuser/userbuild/log/.userbuilddependencies
/u/ibmuser/userbuild/COBOL/DFH0XVDS.cbl
```

4. Optionally you can review the build output log files including the compiler listing under the project directory **/logs**
- It is a known issue that sometimes you might need to rerun the user build to see the **logs** directory.

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "CICS-CATALOG-MANAGER".
- Editor:** Displays the log file "DFH0XVDS.log" with the following content:
 

```

PP 5655-EC6 IBM Enterprise COBOL for z/OS 6.3.0 P211108
Date 10/06/2022
Invocation parameters:
LIB(CICS)

IGY0S4090-I The "LIB" option specification is no longer required. COBOL ]
```
- Terminal:** Shows the command "ibmuser@ibmuser:~/userbuild\$" and the output of the build command, which includes:
 

```

RC=0
** Build finished
/u/ibmuser/userbuild>
Looking for user build logs in "/u/ibmuser/userbuild/log".
Downloading logs to c:/Users/Administrator/Desktop/Labs/vscode-workspace/cics-catalog-manager/logs
Downloading log file /u/ibmuser/userbuild/log/.userbuilddependencies. Binary: false
```

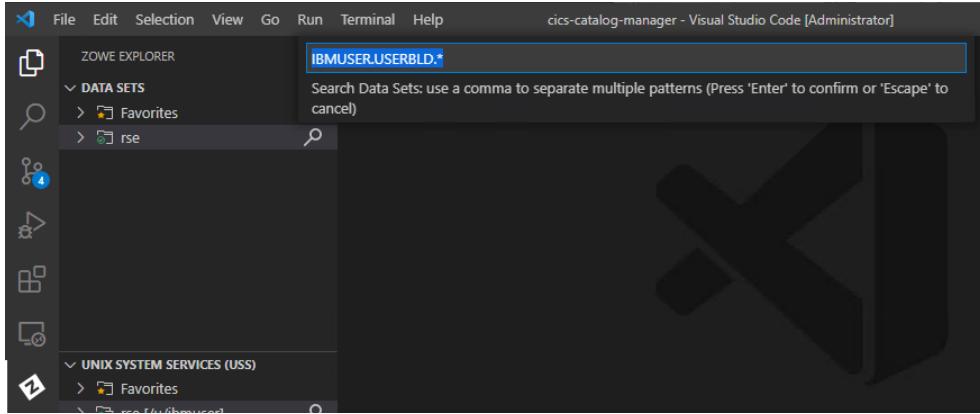
5. As you can see in the log, while we ran the user build for program **DFH0XVDS.cbl**, the **DFH0XCP1.cpy** is also uploaded to z/OS for the build because it has been identified as a dependent include file that's required for building the program.

```

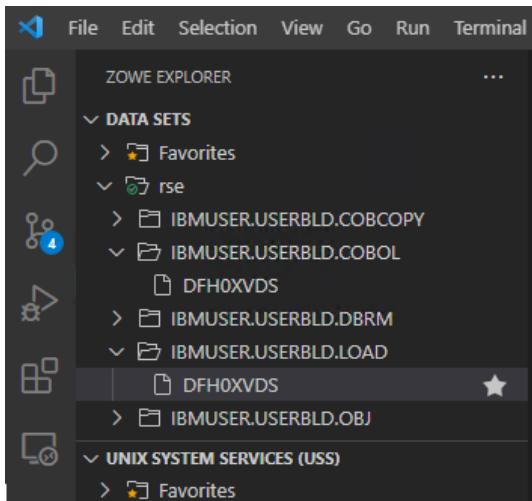
Checking if include files were updated since the last build...
Updating 1/1 local include files.
Uploading file C:\Users\Administrator\Desktop\Labs\vscode-workspace\cics-
catalog-manager\COPYBOOK\DFH0XCP1.cpy to
/u/ibmuser/userbuild/COPYBOOK/DFH0XCP1.cpy with encoding ibm-1047

```

6. Optionally use the **Zowe Explorer** icon in the navbar to connect to the remote z/OS to check the load module from the user build.
7. Under the **DATA SETS** section, click the pre-configured connection profile **rse** and click the magnifier icon to create a dataset filter called **IBMUSER.USERBLD.\*** and hit **Enter** TWICE.



8. You can see the load module has been compiled into **IBMUSER.USERBLD.LOAD**. In the next part of the lab we will run early integration test against this load module.



**Note:**

The User Build capability of the IBM Z Open Editor VS Code extension helps COBOL developers to leverage IBM Dependency Based Build (DBB) toolkit right from their local VS Code or Eclipse Che development environment. A developer who is working on a COBOL application can run a user build to compile and link programs before the code is ready to be exposed to the repository for others to use. With user build, you can compile your program without having to perform commits or pushes.

IBM User Build automatically does the following when you click to run a build:

- Uploads only the necessary files and folders to z/OS
- Resolves and uploads COBOL application's copybooks
- Executes DBB user build script on z/OS
- Downloads log folders to your local git directory so you don't have to navigate to the remote z/OS system to view them.

IBM User Build allows developers to do their COBOL development locally using modern tools without context switching.

## **Part 6: Early integration test of the application before deployment**

In this part of the lab exercise, you will perform an early integration test to verify that your code change did fix the issue and it's now working and playing well with the rest of the CICS application, before deploying it into a CICS environment.

**Note:**

IBM Z Virtual Test Platform (ZVTP) provides capabilities for developers and testers to perform batch and transactional testing without the need for data or supporting subsystem (CICS, Db2, IMS, or MQ) products.

ZVTP works by intercepting calls made by an application to subsystems or other programs and recording details about those calls. After the data is recorded, those same application programs can be re-run in batch without the need for the original test environment by returning values from the recorded data. This allows users to automate the testing process of online transactions and batch programs, whether they are testing a single program, hundreds of programs, or thousands of transactions. Additionally, the application programs can be modified with required changes and then rerun to help ensure they perform without adverse effects. Subsystem responses to program calls can also be altered during the recording by user commands or during replay by using exit points. This allows users to test execution paths that programs do not normally take without having to make any program changes.

The product is composed of three parts: Dynamic Test Runner (for recording and replay), the optional ZVTP Viewer, and the optional ZVTP Server Extension. None of these parts requires any changes to user application code. There is no need for recompilation, relinking, or rebinding. The process does not require access to a debugger. All of the call interceptions are performed by using the user's original load modules, whether the application runs in batch, CICS, or IMS.

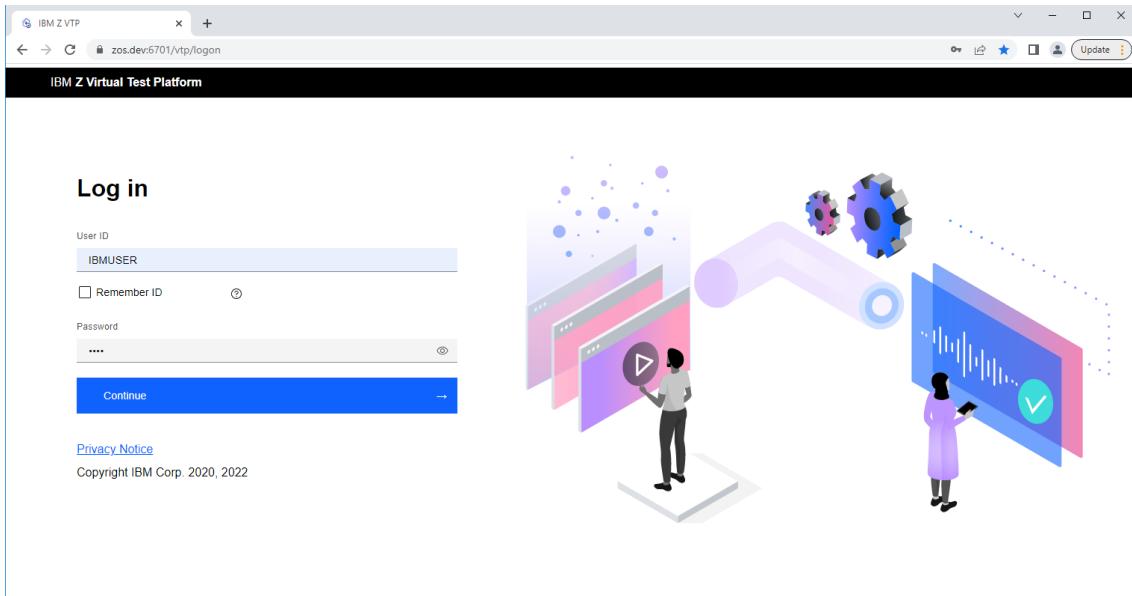
ZVTP can be used independently or in cooperation with other products. For more information, see product document. <https://www.ibm.com/docs/en/zvtp/2.0?topic=overview>

ZVTP is intended to be used by development and testing. It is not intended for a high-throughput or production environment. This product intercepts and records data contained in the arguments of various calls and this data is not "sanitized". Do not record production or sensitive data because this information can easily be viewed in the sequential file created during the recording process.

## Explore the IBM Z Virtual Test Platform web interface

1. Open a new browser tab and click **IBM VTP** from the bookmark or navigate to

<https://zos.dev:6701/vtp/logon>



2. Log in use **IBMUSER** and password **SYS1** and click **Continue** to go to the **Test cases** overview page.

Name	Type	Transactions	Programs	Recording time	Replay time	Result history	Recent result
Catalog Manager BMS 3270	CICS	EGUI	DFH0XSSM +4	9/7/2022, 8:33:14 PM	-	-	-
Catalog Manager call from API	CICS	CSMI	DFH0XCMN +1	9/19/2022, 8:31:29 PM	-	-	-

3. Click on the **Catalog Manager Call from API** test case which is a previous Virtual Test Platform recording of the execution path from API before the defect is fixed. It recorded all the CICS activity and Input/Output from all the CICS calls that's initiated when running a catalog inquire call from z/OS Connect API. By clicking on the different **Programs** name you can check the **Statement** of the EXEC CICS calls made from the program.

4. Click on the **Timeline** which show all the CICS calls involved, you can see the API invokes **DFH0XCMN** which subsequence LINK into **DFH0XVDS** where the actual VSAM **READNEXT** requests were captured before returning the control back to **DFH0XCMN**.

### Replay the test case using the new load module built by user build

5. Now we can replay the same test case using the user build load module we built in previous step. Click **Run test case** button at the top.
6. Provide the user build load module PDS name **IBMUSER.USERBLD.LOAD** in the **User library** section.

7. Click **Run** button to submit the replay of the test case, IBM Z VTP will replay the test case use updated user build load module and it will compare the test result.
- If you have been idle for too long since the log in, your user session will expire, and the run test case will fail. Just login the Virtual Test Platform again and retry the operation.

[Test cases](#)

### Catalog Manager call from API

Replay time: 10/5/2022, 6:12:24 PM

[Run test case](#) [Save changes](#) [1 result history](#)

[About test case](#) [Test results \(RC=4\)](#)

The test case was run successfully

Test case named Catalog Manager call from API was run successfully. The table and history will be refreshed.

8. When finished you will see a new entry of the **test result** is recorded with **RC=4** which indicates some discrepancies were found, click on the **test results** to look at the details.
9. Any mismatch between **recorded** vs **replayed** is highlighted with yellow icon, click on the **DFH0XVDS** program you will notice up until the (EXEC CICS) **RETURN** everything was a perfect match. Then click on the **RETURN** it then highlights the COMMAREA differences. While the recorded price was **999.99**, now during the replay it's now **002.90** with the new user build load module. It shows your code change has now fixed the defect.

Transaction	Program	Statement	Record #	Argument	Offset	Commarea
CSMI	DFH0XVDS	RETURN	42			COMMAREA
CSMI	DFH0XVDS	READ				
CSMI	DFH0XVDS	STARTBR				
CSMI	DFH0XVDS	READNEXT				
CSMI	DFH0XVDS	ENDBR				
CSMI	DFH0XVDS	RETURN	145			

Statement details

Recorded	Replayed
-3-----4-----5-----6-----7-----8-----9-----0----- 44444444444444FF <del>FF</del> F4FFFFFFFFFFC8994D9A4C9A54FF99444444444444444444FF BBBBBBBBBBBBB010995B9901330000020133B7552B2345B2472BBBBBBBBBBBBB01	-3-----4-----5-----6-----7-----8-----9-----0----- 44444444444444FF <del>FF</del> F4FFFFFFFFFFC8994D9A4C9A54FF99444444444444444444FF BBBBBBBBBBBBB01002B9001330000020133B7552B2345B2472BBBBBBBBBBBBB01

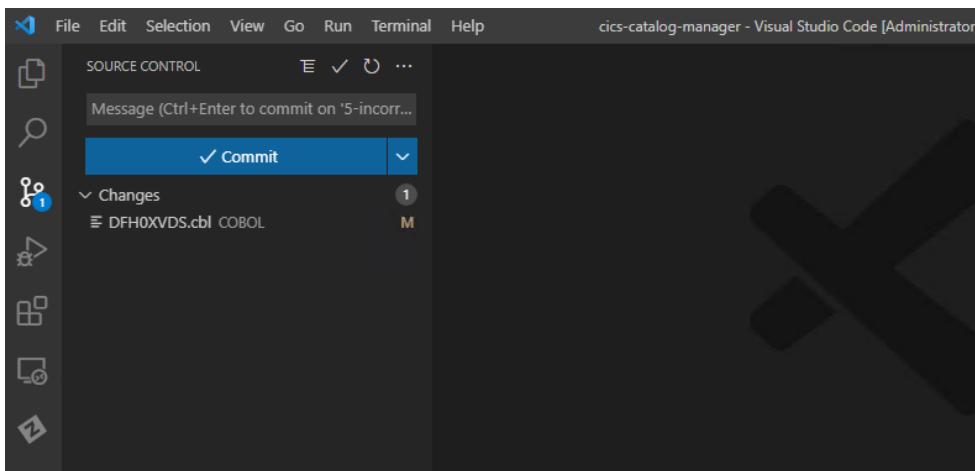
10. What happened in the background is the replay process kicks off a batch job to run your load module, without the need of CICS environment (or any other middleware your application may require, e.g. DB2, VSAM, IMS or MQ). While executing your program, VTP takes over all the external system calls and returns pre-recorded data to your program, and your program won't even know that it's not running in a real CICS region but in a virtual environment (hence the name Virtual Test Platform). VTP then capture any changes in its behavior – whether it's intended or not and report it back.
11. This is what we call an **early integration test** so that you can shift left the testing before the deployment, removing dependencies to the external resources so that you can test earlier and more often. But obviously, this is not a replacement for a full integration test which is still required at a later stage.

## Part 7: Commit the code and push to the remote Git repo to close the merge request

After testing, you are confident the code is ready so you will commit it to your local working branch of the Git repository and merge it to the remote GitLab repo to submit the work you've done. In this lab it will automatically trigger the pipeline for a full application build on the branch, and when the build is successful, you can then safely merge the changes into the master branch.

### Commit the code change and push it to the remote repository on GitLab

1. Go to the **Source Control** panel using the button  on the left navbar in VS Code.



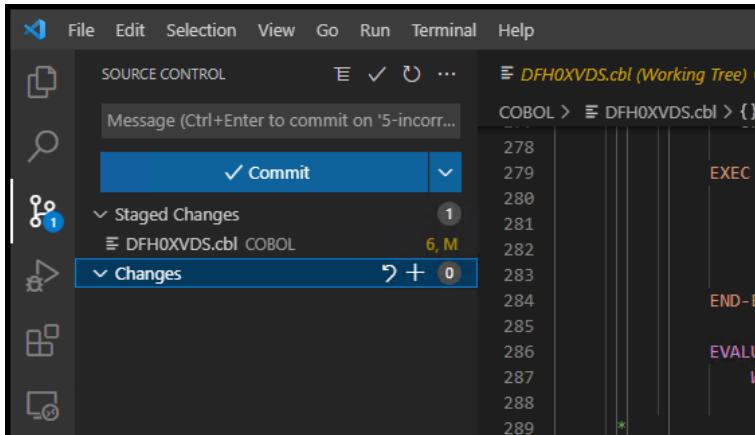
2. Click on the program **DFH0XVDS.cbl** under the **changes** to see the diff between the local copy and the Git branch. You will see the line you have been commented out with the MOVE statement is highlighted.

```

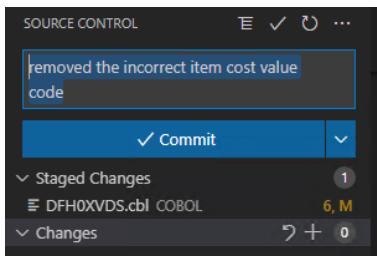
278 EXEC CICS READNEXT FILE(WS-FILENAME)          278
279   INTO(WS-CAT-ITEM)                           279
280   RIDFLD(WS-CURRENT-ITEM-REF)                280
281   LENGTH(LENGTH OF WS-CAT-ITEM)               281
282   RESP(WS-RESPONSE-CODE)                     282
283                                         283
284 END-EXEC                                     284
285                                         285
286 EVALUATE WS-RESPONSE-CODE                   286
287 WHEN DFHRESP(NORMAL)                         287
288   MOVE WS-LOOP-COUNTER TO WS-RECORD-COUNT    288
289   *                                           289
290   * this is a bug - start                    290+
291   MOVE 999,99 TO WS-COST IN WS-CAT-ITEM       291
292   * this is a bug - end                      292
293   MOVE WS-CAT-ITEM TO CA-CAT-ITEM(WS-LOOP-COUNTER) 293
294                                         294
295   MOVE WS-RECORD-COUNT TO CA-ITEM-COUNT      295
296   MOVE WS-CURRENT-ITEM-REF TO CA-LAST-ITEM-REF 296
297                                         297
298 WHEN DFHRESP(ENDFILE)                         298
299   MOVE 'Y' TO CATALOG-EOF-SW                 299
300 WHEN OTHER                                     300
301   MOVE 21 TO CA-RETURN-CODE                  301
302   MOVE 'ERROR OCCURRED READING FILE'        302
303   TO CA-RESPONSE-MESSAGE                     303
304                                         304
305                                         305
306                                         306
307                                         307
308                                         308
309                                         309
310                                         310
311                                         311
312                                         312
313                                         313
314                                         314
315                                         315
316                                         316
317                                         317
318                                         318
319                                         319
320                                         320
321                                         321
322                                         322
323                                         323
324                                         324
325                                         325
326                                         326
327                                         327
328                                         328
329                                         329
330                                         330
331                                         331
332                                         332
333                                         333
334                                         334
335                                         335
336                                         336
337                                         337
338                                         338
339                                         339
340                                         340
341                                         341
342                                         342
343                                         343
344                                         344
345                                         345
346                                         346
347                                         347
348                                         348
349                                         349
350                                         350
351                                         351
352                                         352
353                                         353
354                                         354
355                                         355
356                                         356
357                                         357
358                                         358
359                                         359
360                                         360
361                                         361
362                                         362
363                                         363
364                                         364
365                                         365
366                                         366
367                                         367
368                                         368
369                                         369
370                                         370
371                                         371
372                                         372
373                                         373
374                                         374
375                                         375
376                                         376
377                                         377
378                                         378
379                                         379
380                                         380
381                                         381
382                                         382
383                                         383
384                                         384
385                                         385
386                                         386
387                                         387
388                                         388
389                                         389
390                                         390
391                                         391
392                                         392
393                                         393
394                                         394
395                                         395
396                                         396
397                                         397
398                                         398
399                                         399
400                                         400
401                                         401
402                                         402
403                                         403
404                                         404
405                                         405
406                                         406
407                                         407
408                                         408
409                                         409
410                                         410
411                                         411
412                                         412
413                                         413
414                                         414
415                                         415
416                                         416
417                                         417
418                                         418
419                                         419
420                                         420
421                                         421
422                                         422
423                                         423
424                                         424
425                                         425
426                                         426
427                                         427
428                                         428
429                                         429
430                                         430
431                                         431
432                                         432
433                                         433
434                                         434
435                                         435
436                                         436
437                                         437
438                                         438
439                                         439
440                                         440
441                                         441
442                                         442
443                                         443
444                                         444
445                                         445
446                                         446
447                                         447
448                                         448
449                                         449
450                                         450
451                                         451
452                                         452
453                                         453
454                                         454
455                                         455
456                                         456
457                                         457
458                                         458
459                                         459
460                                         460
461                                         461
462                                         462
463                                         463
464                                         464
465                                         465
466                                         466
467                                         467
468                                         468
469                                         469
470                                         470
471                                         471
472                                         472
473                                         473
474                                         474
475                                         475
476                                         476
477                                         477
478                                         478
479                                         479
480                                         480
481                                         481
482                                         482
483                                         483
484                                         484
485                                         485
486                                         486
487                                         487
488                                         488
489                                         489
490                                         490
491                                         491
492                                         492
493                                         493
494                                         494
495                                         495
496                                         496
497                                         497
498                                         498
499                                         499
500                                         500
501                                         501
502                                         502
503                                         503
504                                         504
505                                         505
506                                         506
507                                         507
508                                         508
509                                         509
510                                         510
511                                         511
512                                         512
513                                         513
514                                         514
515                                         515
516                                         516
517                                         517
518                                         518
519                                         519
520                                         520
521                                         521
522                                         522
523                                         523
524                                         524
525                                         525
526                                         526
527                                         527
528                                         528
529                                         529
530                                         530
531                                         531
532                                         532
533                                         533
534                                         534
535                                         535
536                                         536
537                                         537
538                                         538
539                                         539
540                                         540
541                                         541
542                                         542
543                                         543
544                                         544
545                                         545
546                                         546
547                                         547
548                                         548
549                                         549
550                                         550
551                                         551
552                                         552
553                                         553
554                                         554
555                                         555
556                                         556
557                                         557
558                                         558
559                                         559
560                                         560
561                                         561
562                                         562
563                                         563
564                                         564
565                                         565
566                                         566
567                                         567
568                                         568
569                                         569
570                                         570
571                                         571
572                                         572
573                                         573
574                                         574
575                                         575
576                                         576
577                                         577
578                                         578
579                                         579
580                                         580
581                                         581
582                                         582
583                                         583
584                                         584
585                                         585
586                                         586
587                                         587
588                                         588
589                                         589
590                                         590
591                                         591
592                                         592
593                                         593
594                                         594
595                                         595
596                                         596
597                                         597
598                                         598
599                                         599
600                                         600
601                                         601
602                                         602
603                                         603
604                                         604
605                                         605
606                                         606
607                                         607
608                                         608
609                                         609
610                                         610
611                                         611
612                                         612
613                                         613
614                                         614
615                                         615
616                                         616
617                                         617
618                                         618
619                                         619
620                                         620
621                                         621
622                                         622
623                                         623
624                                         624
625                                         625
626                                         626
627                                         627
628                                         628
629                                         629
630                                         630
631                                         631
632                                         632
633                                         633
634                                         634
635                                         635
636                                         636
637                                         637
638                                         638
639                                         639
640                                         640
641                                         641
642                                         642
643                                         643
644                                         644
645                                         645
646                                         646
647                                         647
648                                         648
649                                         649
650                                         650
651                                         651
652                                         652
653                                         653
654                                         654
655                                         655
656                                         656
657                                         657
658                                         658
659                                         659
660                                         660
661                                         661
662                                         662
663                                         663
664                                         664
665                                         665
666                                         666
667                                         667
668                                         668
669                                         669
670                                         670
671                                         671
672                                         672
673                                         673
674                                         674
675                                         675
676                                         676
677                                         677
678                                         678
679                                         679
680                                         680
681                                         681
682                                         682
683                                         683
684                                         684
685                                         685
686                                         686
687                                         687
688                                         688
689                                         689
690                                         690
691                                         691
692                                         692
693                                         693
694                                         694
695                                         695
696                                         696
697                                         697
698                                         698
699                                         699
700                                         700
701                                         701
702                                         702
703                                         703
704                                         704
705                                         705
706                                         706
707                                         707
708                                         708
709                                         709
710                                         710
711                                         711
712                                         712
713                                         713
714                                         714
715                                         715
716                                         716
717                                         717
718                                         718
719                                         719
720                                         720
721                                         721
722                                         722
723                                         723
724                                         724
725                                         725
726                                         726
727                                         727
728                                         728
729                                         729
730                                         730
731                                         731
732                                         732
733                                         733
734                                         734
735                                         735
736                                         736
737                                         737
738                                         738
739                                         739
740                                         740
741                                         741
742                                         742
743                                         743
744                                         744
745                                         745
746                                         746
747                                         747
748                                         748
749                                         749
750                                         750
751                                         751
752                                         752
753                                         753
754                                         754
755                                         755
756                                         756
757                                         757
758                                         758
759                                         759
760                                         760
761                                         761
762                                         762
763                                         763
764                                         764
765                                         765
766                                         766
767                                         767
768                                         768
769                                         769
770                                         770
771                                         771
772                                         772
773                                         773
774                                         774
775                                         775
776                                         776
777                                         777
778                                         778
779                                         779
780                                         780
781                                         781
782                                         782
783                                         783
784                                         784
785                                         785
786                                         786
787                                         787
788                                         788
789                                         789
790                                         790
791                                         791
792                                         792
793                                         793
794                                         794
795                                         795
796                                         796
797                                         797
798                                         798
799                                         799
800                                         800
801                                         801
802                                         802
803                                         803
804                                         804
805                                         805
806                                         806
807                                         807
808                                         808
809                                         809
810                                         810
811                                         811
812                                         812
813                                         813
814                                         814
815                                         815
816                                         816
817                                         817
818                                         818
819                                         819
820                                         820
821                                         821
822                                         822
823                                         823
824                                         824
825                                         825
826                                         826
827                                         827
828                                         828
829                                         829
830                                         830
831                                         831
832                                         832
833                                         833
834                                         834
835                                         835
836                                         836
837                                         837
838                                         838
839                                         839
840                                         840
841                                         841
842                                         842
843                                         843
844                                         844
845                                         845
846                                         846
847                                         847
848                                         848
849                                         849
850                                         850
851                                         851
852                                         852
853                                         853
854                                         854
855                                         855
856                                         856
857                                         857
858                                         858
859                                         859
860                                         860
861                                         861
862                                         862
863                                         863
864                                         864
865                                         865
866                                         866
867                                         867
868                                         868
869                                         869
870                                         870
871                                         871
872                                         872
873                                         873
874                                         874
875                                         875
876                                         876
877                                         877
878                                         878
879                                         879
880                                         880
881                                         881
882                                         882
883                                         883
884                                         884
885                                         885
886                                         886
887                                         887
888                                         888
889                                         889
890                                         890
891                                         891
892                                         892
893                                         893
894                                         894
895                                         895
896                                         896
897                                         897
898                                         898
899                                         899
900                                         900
901                                         901
902                                         902
903                                         903
904                                         904
905                                         905
906                                         906
907                                         907
908                                         908
909                                         909
910                                         910
911                                         911
912                                         912
913                                         913
914                                         914
915                                         915
916                                         916
917                                         917
918                                         918
919                                         919
920                                         920
921                                         921
922                                         922
923                                         923
924                                         924
925                                         925
926                                         926
927                                         927
928                                         928
929                                         929
930                                         930
931                                         931
932                                         932
933                                         933
934                                         934
935                                         935
936                                         936
937                                         937
938                                         938
939                                         939
940                                         940
941                                         941
942                                         942
943                                         943
944                                         944
945                                         945
946                                         946
947                                         947
948                                         948
949                                         949
950                                         950
951                                         951
952                                         952
953                                         953
954                                         954
955                                         955
956                                         956
957                                         957
958                                         958
959                                         959
960                                         960
961                                         961
962                                         962
963                                         963
964                                         964
965                                         965
966                                         966
967                                         967
968                                         968
969                                         969
970                                         970
971                                         971
972                                         972
973                                         973
974                                         974
975                                         975
976                                         976
977                                         977
978                                         978
979                                         979
980                                         980
981                                         981
982                                         982
983                                         983
984                                         984
985                                         985
986                                         986
987                                         987
988                                         988
989                                         989
990                                         990
991                                         991
992                                         992
993                                         993
994                                         994
995                                         995
996                                         996
997                                         997
998                                         998
999                                         999
1000                                         1000
1001                                         1001
1002                                         1002
1003                                         1003
1004                                         1004
1005                                         1005
1006                                         1006
1007                                         1007
1008                                         1008
1009                                         1009
1010                                         1010
1011                                         1011
1012                                         1012
1013                                         1013
1014                                         1014
1015                                         1015
1016                                         1016
1017                                         1017
1018                                         1018
1019                                         1019
1020                                         1020
1021                                         1021
1022                                         1022
1023                                         1023
1024                                         1024
1025                                         1025
1026                                         1026
1027                                         1027
1028                                         1028
1029                                         1029
1030                                         1030
1031                                         1031
1032                                         1032
1033                                         1033
1034                                         1034
1035                                         1035
1036                                         1036
1037                                         1037
1038                                         1038
1039                                         1039
1040                                         1040
1041                                         1041
1042                                         1042
1043                                         1043
1044                                         1044
1045                                         1045
1046                                         1046
1047                                         1047
1048                                         1048
1049                                         1049
1050                                         1050
1051                                         1051
1052                                         1052
1053                                         1053
1054                                         1054
1055                                         1055
1056                                         1056
1057                                         1057
1058                                         1058
1059                                         1059
1060                                         1060
1061                                         1061
1062                                         1062
1063                                         1063
1064                                         1064
1065                                         1065
1066                                         1066
1067                                         1067
1068                                         1068
1069                                         1069
1070                                         1070
1071                                         1071
1072                                         1072
1073                                         1073
1074                                         1074
1075                                         1075
1076                                         1076
1077                                         1077
1078                                         1078
1079                                         1079
1080                                         1080
1081                                         1081
1082                                         1082
1083                                         1083
1084                                         1084
1085                                         1085
1086                                         1086
1087                                         1087
1088                                         1088
1089                                         1089
1090                                         1090
1091                                         1091
1092                                         1092
1093                                         1093
1094                                         1094
1095                                         1095
1096                                         1096
1097                                         1097
1098                                         1098
1099                                         1099
1100                                         1100
1101                                         1101
1102                                         1102
1103                                         1103
1104                                         1104
1105                                         1105
1106                                         1106
1107                                         1107
1108                                         1108
1109                                         1109
1110                                         1110
1111                                         1111
1112                                         1112
1113                                         1113
1114                                         1114
1115                                         1115
1116                                         1116
1117                                         1117
1118                                         1118
1119                                         1119
1120                                         1120
1121                                         1121
1122                                         1122
1123                                         1123
1124                                         1124
1125                                         1125
1126                                         1126
1127                                         1127
1128                                         1128
1129                                         1129
1130                                         1130
1131                                         1131
1132                                         1132
1133                                         1133
1134                                         1134
1135                                         1135
1136                                         1136
1137                                         1137
1138                                         1138
1139                                         1139
1140                                         1140
1141                                         1141
1142                                         1142
1143                                         1143
1144                                         1144
1145                                         1145
1146                                         1146
1147                                         1147
1148                                         1148
1149                                         1149
1150                                         1150
1151                                         1151
1152                                         1152
1153                                         1153
1154                                         1154
1155                                         1155
1156                                         1156
1157                                         1157
1158                                         1158
1159                                         1159
1160                                         1160
1161                                         1161
1162                                         1162
1163                                         1163
1164                                         1164
1165                                         1165
1166                                         1166
1167                                         1167
1168                                         1168
1169                                         1169
1170                                         1170
1171                                         1171
1172                                         1172
1173                                         1173
1174                                         1174
1175                                         1175
1176                                         1176
1177                                         1177
1178                                         1178
1179                                         1179
1180                                         1180
1181                                         1181
1182                                         1182
1183                                         1183
1184                                         1184
1185                                         1185
1186                                         1186
1187                                         1187
1188                                         1188
1189                                         1189
1190                                         1190
1191                                         1191
1192                                         1192
1193                                         1193
1194                                         1194
1195                                         1195
1196                                         1196
1197                                         1197
1198                                         1198
1199                                         1199
1200                                         1200
1201                                         1201
1202                                         1202
1203                                         1203
1204                                         1204
1205                                         1205
1206                                         1206
1207                                         1207
1208                                         1208
1209                                         1209
1210                                         1210
1211                                         1211
1212                                         1212
1213                                         1213
1214                                         1214
1215                                         1215
1216                                         1216
1217                                         1217
1218                                         1218
1219                                         1219
1220                                         1220
1221                                         1221
1222                                         1222
1223                                         1223
1224                                         1224
1225                                         1225
1226                                         1226
1227                                         1227
1228                                         1228
1229                                         1229
1230                                         1230
1231                                         1231
1232                                         1232
1233                                         1233
1234                                         1234
1235                                         1235
1236                                         1236
1237                                         1237
1238                                         1238
1239                                         1239
1240                                         1240
1241                                         1241
1242                                         1242
1243                                         1243
1244                                         1244
1245                                         1245
1246                                         1246
1247                                         1247
1248                                         1248
1249                                         1249
1250                                         1250
1251                                         1251
1252                                         1252
1253                                         1253
1254                                         1254
1255                                         1255
1256                                         1256
1257                                         1257
1258                                         1258
1259                                         1259
1260                                         1260
1261                                         1261
1262                                         1262
1263                                         1263
1264                                         1264
1265                                         1265
1266                                         1266
1267                                         1267
1268                                         1268
1269                                         1269
1270                                         1270
1271                                         1271
1272                                         1272
1273                                         1273
1274                                         1274
1275                                         1275
1276                                         1276
1277                                         1277
1278                                         1278
1279                                         1279
1280                                         1280
1281                                         1281
1282                                         1282
1283                                         1283
1284                                         1284
1285                                         1285
1286                                         1286
1287                                         1287
1288                                         1288
1289                                         1289
1290                                         1290
1291                                         1291
1292                                         1292
1293                                         1293
1294                                         1294
1295                                         1295
1296                                         1296
1297                                         1297
1298                                         1298
1299                                         1299
1300                                         1300
1301                                         1301
1302                                         1302
1303                                         1303
1304                                         1304
1305                                         1305
1306                                         1306
1307                                         1307
1308                                         1308
1309                                         1309
1310                                         1310
1311                                         1311
1312                                         1312
1313                                         1313
1314                                         1314
1315                                         1315
1316                                         1316
1317                                         1317
1318                                         1318
1319                                         1319
1320                                         1320
1321                                         1321
1322                                         1322
1323                                         1323
1324                                         1324
1325                                         1325
1326                                         1326
1327                                         1327
1328                                         1328
1329                                         1329
1330                                         1330
1331                                         1331
1332                                         1332
1333                                         1333
1334                                         1334
1335                                         1335
1336                                         1336
1337                                         1337
1338                                         1338
1339                                         1339
1340                                         1340
1341                                         1341
1342                                         1342
1343                                         1343
1344                                         1344
1345                                         1345
1346                                         1346
1347                                         1347
1348                                         1348
1349                                         1349
1350                                         1350
1351                                         1351
1352                                         1352
1353                                         1353
1354                                         1354
1355                                         1355
1356                                         1356
1357                                         1357
1358                                         1358
1359                                         1359
1360                                         1360
1361                                         1361
1362                                         1362
1363                                         1363
1364                                         1364
1365                                         1365
1366                                         1366
1367                                         1367
1368                                         1368
1369                                         1369

```

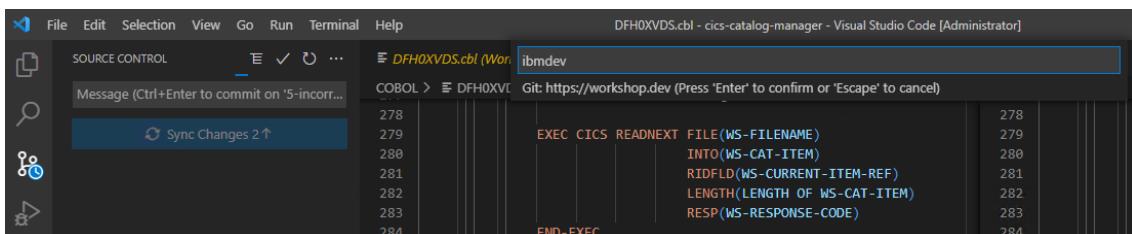
- \_\_\_ 3. Click the + (plus) sign next to the program to **stage** the changes for the commit.



- \_\_\_ 4. Put a commit message e.g. `Removed the incorrect item cost value code` and click **Commit** button to commit to local Git repository.



- \_\_\_ 5. Click **Sync Change** button to push the changes in local git repository to the GitLab remote repo, you will need to provide your GitLab credential `ibmdev` and `Passw0rd` in order to be able to push the changes.



## Review the changes on GitLab and pipeline for the full application build

- \_\_\_ 6. Go to GitLab browser window and navigate to the project homepage by clicking the project name on the top left

7. Switch the branch to our working branch **n-incorrect-item-cost**

Name	Last commit	Last update
.vscode	updated project structure	17 minutes ago
ASM	Initial commit for CICS & DevOps Workshop	2 weeks ago
ASMCOPY	Initial commit for CICS & DevOps Workshop	2 weeks ago
COBOL	removed the incorrect item cost value code	9 minutes ago
COPYBOOK	Initial commit for CICS & DevOps Workshop	2 weeks ago
COPYLIB-MVS	Initial commit for CICS & DevOps Workshop	2 weeks ago
COPYLIB	Initial commit for CICS & DevOps Workshop	2 weeks ago

8. Click on **commit message** of the latest commit we just made.

Name	Last commit	Last update
.vscode	updated project structure	17 minutes ago
ASM	Initial commit for CICS & DevOps Workshop	2 weeks ago
ASMCOPY	Initial commit for CICS & DevOps Workshop	2 weeks ago
COBOL	removed the incorrect item cost value code	9 minutes ago
COPYBOOK	Initial commit for CICS & DevOps Workshop	2 weeks ago
COPYLIB-MVS	Initial commit for CICS & DevOps Workshop	2 weeks ago
COPYLIB	Initial commit for CICS & DevOps Workshop	2 weeks ago

9. This screen shows the commit details and the merge request we submitted for fixing this defect, as well as the pipeline that's triggered by this commit with all the individual jobs' status.

removed the incorrect item cost value code

parent 6fe0b464 5-incorrect-item-cost

1 merge request | 6 Draft: Resolve "Incorrect item cost"

Pipeline #101 passed with stages (✓) in 2 minutes and 22 seconds

Changes 1 Pipelines 1

Showing 1 changed file ▾ with 1 addition and 1 deletion

Hide whitespace changes Inline Side-by-side

**COBOL/DFH0XVDS.cbl**

```

...   ... @@ -287,7 +287,7 @@
287      287           WHEN DFHRESP(NORMAL)
288      288           MOVE WS-LOOP-COUNTER TO WS-RECORD-COUNT
289      289           *          this is a bug - start
290      -             MOVE 999.99 TO WS-COST IN WS-CAT-ITEM
291      291           +             MOVE 999.99 TO WS-COST IN WS-CAT-ITEM
292      292           *          this is a bug - end
293      293           MOVE WS-CAT-ITEM TO CA-CAT-ITEM(WS-LOOP-COUNTER)
...

```

10. Click on the Pipeline id #101 (your id will be different) to inspect the execution of the jobs in this pipeline.

IBM Devs > cics-catalog-manager > Pipelines > #101

passed Pipeline #101 triggered 20 minutes ago by IBM Devs Delete

removed the incorrect item cost value code

4 jobs for 5-incorrect-item-cost in 2 minutes and 22 seconds (queued for 3 seconds)

latest

- 2b7945a0

1 related merge request: | 6 Draft: Resolve "Incorrect item cost"

Pipeline Needs Jobs 4 Tests 0

Build	Report	Deploy	Test
DBB Build	Build Report	CICS Deploy	Integration Test

11. Click on the **DBB Build** you can review the detail log of the **IBM Dependency Build** on the z/OS. In there you can also find a link to the DBB Web App for a consolidated view of the build result.

```

16 Fetching changes with git depth set to 20...
17 Reinitialized existing Git repository in /u/ibmuser/builds/egvgArRC/0/ibmdev/cics-catalog-manager/.git/
18 Checking out 2b7945a0 as 5-incorrect-item-cost...
19 Removing builds/
20 Skipping Git submodules setup
21 Executing "step_script" stage of the job script
22 Set up environment for zos
23 /usr/lpp/cbclib/xlclang/bin/xlclang found, using CGO
24 Done!
25 $ pwd # collapsed multi-line command
26 /u/ibmuser/builds/egvgArRC/0/ibmdev/cics-catalog-manager
27 $ $DBB_HOME/bin/groovyz \ # collapsed multi-line command
28 ** Build start at 20221005.065205.052
29 ** Repository client created for https://workshop.dev:9444/dbb
30 ** Build output located at builds/egvgArRC/0/ibmdev/cics-catalog-manager/builds/101/build.20221005.065205.052
31 ** Build result created for BuildGroup:cics-catalog-manager-5-incorrect-item-cost BuildLabel:build.20221005.065205.052 at https://workshop.dev:9444/dbb/rest/buildResult/543
32 ** --fullBuild option selected. Building all programs for application cics-catalog-manager
33 ** Writing build list file to builds/egvgArRC/0/ibmdev/cics-catalog-manager/builds/101/build.20221005.065205.052/buildList.txt
34 ** Scanning source code.
35 ** Invoking build scripts according to build order: BMS.groovy,Cobol.groovy,Assembler.groovy,P
LI.groovy,LinkEdit.groovy
36 ** Building files mapped to Cobol.groovy script
37 *** Building file cics-catalog-manager/COBOL/DFH0XPOW.cbl
38 *** Building file cics-catalog-manager/COBOL/DFH0XODE.cbl
39 *** Building file cics-catalog-manager/COBOL/DFH0XECC.cbl
40 *** Building file cics-catalog-manager/COBOL/DFH0XVDS.cbl
41 *** Building file cics-catalog-manager/COBOL/DFH0XCMN.cbl
42 *** Building file cics-catalog-manager/COBOL/DFH0XGUI.cbl
43 *** Building file cics-catalog-manager/COBOL/DFH0XCUI.cbl
44 *** Building file cics-catalog-manager/COBOL/SAM2.cbl
45 *** Building file cics-catalog-manager/COBOL/DFH0XSSM.cbl
46 *** Building file cics-catalog-manager/COBOL/DFH0XST2.cbl
47 *** Building file cics-catalog-manager/COBOL/DFH0XWOD.cbl
48 *** Building file cics-catalog-manager/COBOL/DFH0XWOD.cbl
49 *** Building file cics-catalog-manager/COBOL/SAM1.cbl

```

12. You can also log on to the DBB web app to exam all the build results from all developers. Right-Click on this link in the DBB log and choose to Open link in a new tab.

<https://workshop.dev:9444/dbb>

Alternatively open a new browser tab and click the IBM DBB link from the bookmark bar.

13. When prompted use ibmdev and Passw0rd to log in and now you can inspect the build collections and detail build result.

The screenshot shows a web-based application window titled "IBM DBB". The main content area is titled "IBM Dependency Based Build". It displays a table of collections:

Collection	Collection Owner	Latest Build	Build Owner
cics-catalog-manager-5-incorrect-item-cost	ibmdev	10/5/2022	ibmdev
cics-catalog-manager-5-incorrect-item-cost-outputs	ibmdev	10/5/2022	ibmdev

At the bottom of the table, there are pagination controls: "Items per page: 15", "1–2 of 2 items", "1 of 1 page", and navigation arrows.

14. Expand the collection name with your branch name and click on one of the **Build Results**, then click **View Build Report**, you can review the details of the program with dependencies and their compile return code etc.

#### Main Content

##### Build Report

###### Toolkit Version:

Version: 1.1.3  
Build: 151  
Date: 28-Feb-2022 17:26:26

###### Build Summary

Number of files being built: 18

	File	Commands	RC	Data Sets	Outputs	Deploy Type	Logs
1	cics-catalog-manager/COBOL/DFH0XPOW.cbl <a href="#">Show Dependencies</a>	IGYCRCTL	0	IBMUSER.EXMPCAT.COBOL(DFH0XPOW)			DFH0XPOW.cobol.log
		IEWBLINK	0		IBMUSER EXMPCAT LOAD(DFH0XPOW)	LOAD	
2	cics-catalog-manager/COBOL/DFH0XODE.cbl <a href="#">Show Dependencies</a>	IGYCRCTL	0	IBMUSER.EXMPCAT.COBOL(DFH0XODE)			DFH0XODE.cobol.log
		IEWBLINK	0		IBMUSER EXMPCAT LOAD(DFH0XODE)	LOAD	
3	cics-catalog-manager/COBOL/DFH0XECC.cbl <a href="#">Show Dependencies</a>	IGYCRCTL	0	IBMUSER.EXMPCAT.COBOL(DFH0XECC)			DFH0XECC.cobol.log
		IEWBLINK	0		IBMUSER EXMPCAT LOAD(DFH0XECC)	LOAD	
4	cics-catalog-manager/COBOL/DFH0XVDS.cbl <a href="#">Show Dependencies</a>	IGYCRCTL	0	IBMUSER.EXMPCAT.COBOL(DFH0XVDS)			DFH0XVDS.cobol.log
		IEWBLINK	0		IBMUSER EXMPCAT LOAD(DFH0XVDS)	LOAD	
5	cics-catalog-manager/COBOL/DFH0XCMN.cbl <a href="#">Show Dependencies</a>	IGYCRCTL	0	IBMUSER.EXMPCAT.COBOL(DFH0XCMN)			DFH0XCMN.cobol.log
		IEWBLINK	0		IBMUSER EXMPCAT LOAD(DFH0XCMN)	LOAD	
6	cics-catalog-manager/COBOL/DFH0XGUI.cbl <a href="#">Show Dependencies</a>	IGYCRCTL	0	IBMUSER.EXMPCAT.COBOL(DFH0XGUI)			DFH0XGUI.cobol.log
		IEWBLINK	0		IBMUSER EXMPCAT LOAD(DFH0XGUI)	LOAD	

### Note on DBB Collection

In order to use the build dependency information collected by the DependencyScanner for dependency resolution and impact analysis, all scanned source files (both programs and dependency files) will need their resulting logical files stored in the DBB repository database as part of a dependency Collection.

A collection is a repository container for logical files. The scope of a collection is determined by the user but generally a collection contains all the logical files of a Git branch. This way the logical files in a collection can use the scanned file's relative path from the sourceDir as a unique identifier in the collection.

Collections themselves can have any name but it is recommended to use the name of the Git branch of the source files being scanned Collection names must be unique in the DBB repository database. And an error will occur when trying to create a collection that already exists. A good practice is to first check if the collection with that name already exists before creating it.

- 15.** Back to the **GitLab** page, click **Back** in the browser to the pipeline detail page and click on the **Build Report** job. On the right, you can see this job has some artifacts attached to it, click the **Browse** button you can download the compiler output of this build. This step is optional but used to keep a copy of the compiler output on GitLab for demonstration purposes.

IBM Devs > cics-catalog-manager > Jobs > #348

passed Job Build Report triggered 42 minutes ago by IBM Devs

1 Running with gitlab-runner 15.2.1 (32fc1585)  
2 on default-runner Bf55yBPz  
3 Preparing the "shell" executor 00:00  
4 Using Shell executor...  
5 Preparing environment 00:01  
6 Preparing environment 00:01  
7 Running on d-4782-k...  
8 Getting source from Git repository 00:01  
9 Getting source from Git repository 00:01  
10 Fetching changes with git depth set to 20...  
11 Reinitialized existing Git repository in /home/gitlab-runner/builds/Bf55yBPz/0/ibmdev/cics-catalog-manager/.git/  
12 Checking out 2b7945a0 as 5-incorrect-item-cost...  
13 Removing build\_93/  
14 Skipping Git submodules setup  
15 Executing "step\_script" stage of the job script 00:30  
16 Executing "step\_script" stage of the job script  
17 \$ mkdir build\_5CT\_PIPELINE\_ID # collapsed multi-line command  
18 /home/gitlab-runner/builds/Bf55yBPz/0/ibmdev/cics-catalog-manager/build\_101  
19 Successfully downloaded USS file '/u/ibmuser/builds/reports-101.zip' to local file 'reports-101.zip'  
20 created: META-INF/  
21 inflated: META-INF/MANIFEST.MF  
22 inflated: META-INF/REINDEXES-1.1.1

**Build Report**

**Duration:** 33 seconds  
**Finished:** 42 minutes ago  
**Queued:** 2 seconds  
**Timeout:** 1h (from project)

**Runner:** #1 (Bf55yBPz) default-runner

---

**Job artifacts**  
These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.

[Keep](#) [Download](#) [Browse](#)

---

Commit 2b7945a0 in 16 removed the incorrect item cost value code

---

Pipeline #101 for 5-incorrect-item-cost

[Report](#)

---

→ [Build Report](#)

16. Let's navigate back to examine the third job where we do the **CICS Deploy**, this is an example that uses the zowe CLI CICS plugin to dynamically issue **NEWCOPY** command to the CICS programs so that the changes can be refreshed in CICS.

```

2fc1585 version=15.2.1
  19 Downloading artifacts from coordinator... ok      id=348 responseStatus=200 OK token=MaQz5em
    p
  ↴ 21 Executing "step_script" stage of the job script   00:08
    22 $ echo "Deploying to CICS"
    23 Deploying to CICS
    24 $ zowe cics refresh program DFH0XCMN --region-name CICSTS56
    25 The program 'DFH0XCMN' was refreshed successfully.
    26 $ zowe cics refresh program DFH0XVDS --region-name CICSTS56
    27 The program 'DFH0XVDS' was refreshed successfully.
  ↴ 29 Cleaning up project directory and file based variables   00:00
    31 Job succeeded

```

In a real-world scenario, DBB can package the load modules and other resources into a TAR file, then upload to a centralized location like Artifactory or GitLab Package Registry as the CI (continuous integration) pipeline, then there can be another closed loop manages the CD part (continuous delivery) that drives an automated delivery lifecycle for the versions and release management.

- 17. Go back to the previous pipeline screen and look at the **Integration test** job, this is another simple example that invokes **curl** command to test the program by calling an API from z/OS Connect. And you can see the item cost issue has now been fixed.

```

30 {
31   "totalItems": 15,
32   "items": [
33     {
34       "summary": {
35         "stock": "There are 131 items in stock. [KEY=QmFsbC5QZW5zLkJsYWNrLjI0cGsuLi4uLi4uLi4uL
i4uLi4uLi4uLg=]",
36         "orders": "0 on order at unit price of A$2.9. Total order value: $0"
37       },
38       "information": {
39         "itemReference": 10,
40         "description": "Ball Pens Black 24pk",
41         "cost": "2.9",
42         "department": 10,
43         "stock": 131,
44         "onOrder": 0
45       }
46     },

```

#### Note:

We used zAppBuild in our lab. It is a highly customizable build solution for building z/OS applications using Apache Groovy build scripts and IBM Dependency Based Build (DBB) APIs.

The zAppBuild repository is intended to be cloned to a single location on Unix Systems Services (USS) and used to build all of your z/OS applications. This is done by simply copying the supplied application-conf folder (located in the samples folder) to the application source repository you want to build and then verify/update the contained default configuration property values to ensure they meet the build requirements of your application.

The zAppBuild sample provides the following language build scripts by default:

Assembler / BMS / Cobol / LinkEdit / PLI / DBDgen / PSBgen / MFS and ZunitConfig

All language scripts both compile and optionally link-edit programs. The language build scripts are intended to be useful out of the box but depending on the complexity of your applications' build requirements, may require modifications to meet your development team's needs. By following the examples used in the existing language build scripts of keeping all application specific references out of the build scripts and instead using configuration properties with strong default values, the zAppBuild sample can continue to be a generic build solution for all of your specific applications.

The build scope of zAppBuild is an application that is loosely defined as one or more Git repositories containing all the z/OS source files required to build the application. There are no specific rules as to the structure of the repositories except that one repository must contain the high-level application-conf folder provided by zAppBuild which contains all of the configuration properties for building the application programs.

zAppBuild supports a number of build scenarios:

Single Program - Build a single program in the application.

List of Programs - Build a list of programs provided by a text file.

Full Build - Build all programs (or buildable files) of an application.

Impact Build - Build only programs impacted by source files that have changed since the last successful build.

Impact Build with baseline reference - Build only programs impacted by source files that have changed by diffing to a previous configuration reference.

Topic Branch Build - Detects when building a topic branch for the first time and will automatically clone the dependency data collections from the main build branch in order to avoid having to rescan the entire application.

Merge Build - Build only changed programs which will be merged back into the mainBuildBranch by using a triple-dot git diff.

Scan Source - Skip the actual building and only scan source files to store dependency data in collection (migration scenario).

Scan Source + Outputs - Skip the actual building and only scan source files and existing load modules to dependency data in source and output collection (migration scenario with static linkage scenarios).

For more detail refer to <https://github.com/IBM/dbb-zappbuild>

## Complete the merge request to submit work to the master branch

Since everything looks good and our code is ready to progress to the next stage which is to be merge with the master branch.

18. Click on the **Merge Request** in the GitLab project on the left, and you can find the current draft merge request.

19. Click on the **draft merge request** it shows current status is **Open**, the **pipeline** for the branch full build has passed successfully, **approval** is optional in this case, and you can click on **Mark as ready** button to prepare the merge.

20. Click **Merge** button to make it official.

21. Congratulations, you've just merged your work into the master branch. Now your working branch is deleted, also the original GitLab **Issue** is closed automatically too.
22. Now it triggers another build on the master branch so that your code will be built again with any other changes other people might have committed to master while you were working on the fix. Click the project name button to go back to the project home page. **cics-catalog-manager** And you can see the pipeline is running. For this lab, you will expect the same result.

The screenshot shows the GitLab interface for the project 'cics-catalog-manager'. At the top, there's a header with 'IBM Devs > cics-catalog-manager'. Below the header, the project name 'cics-catalog-manager' is displayed with a 'git' icon. To the right are buttons for notifications (Bell), stars (Star 0), forks (Fork 0), and a 'Clone' button. Underneath the project name, it says 'Project ID: 10' and shows a file icon. Below this, it lists '24 Commits', '2 Branches', '0 Tags', and '5.8 MB Project Storage'. The description 'DevOps CI/CD Pipeline Lab - CICS Catalog Manager' follows. A progress bar indicates the status of the pipeline. Below the progress bar, there's a dropdown menu set to 'master', a search bar with 'cics-catalog-manager / +', and buttons for 'Find file', 'Web IDE', and 'Clone'. At the bottom, a commit message is shown: 'Merge branch '5-incorrect-item-cost' into 'master''. It includes an author icon (IBM Devs), a timestamp ('authored 2 minutes ago'), and a commit hash ('3070448c').

## Part 8: Summary

**Congratulations**, you have completed all the tasks in this lab.

In this lab you performed the following steps:

- Exploring the GitLab project and review the issue and create a new branch to work on the fix.
- Create a clone of the new branch into your IDE
- Debugging the application at statement level
- Make the code change
- Run a user build to build the program in your own workspace
- Early integration test of the application before deployment
- Commit the code and push to the remote Git repo to close the merge request