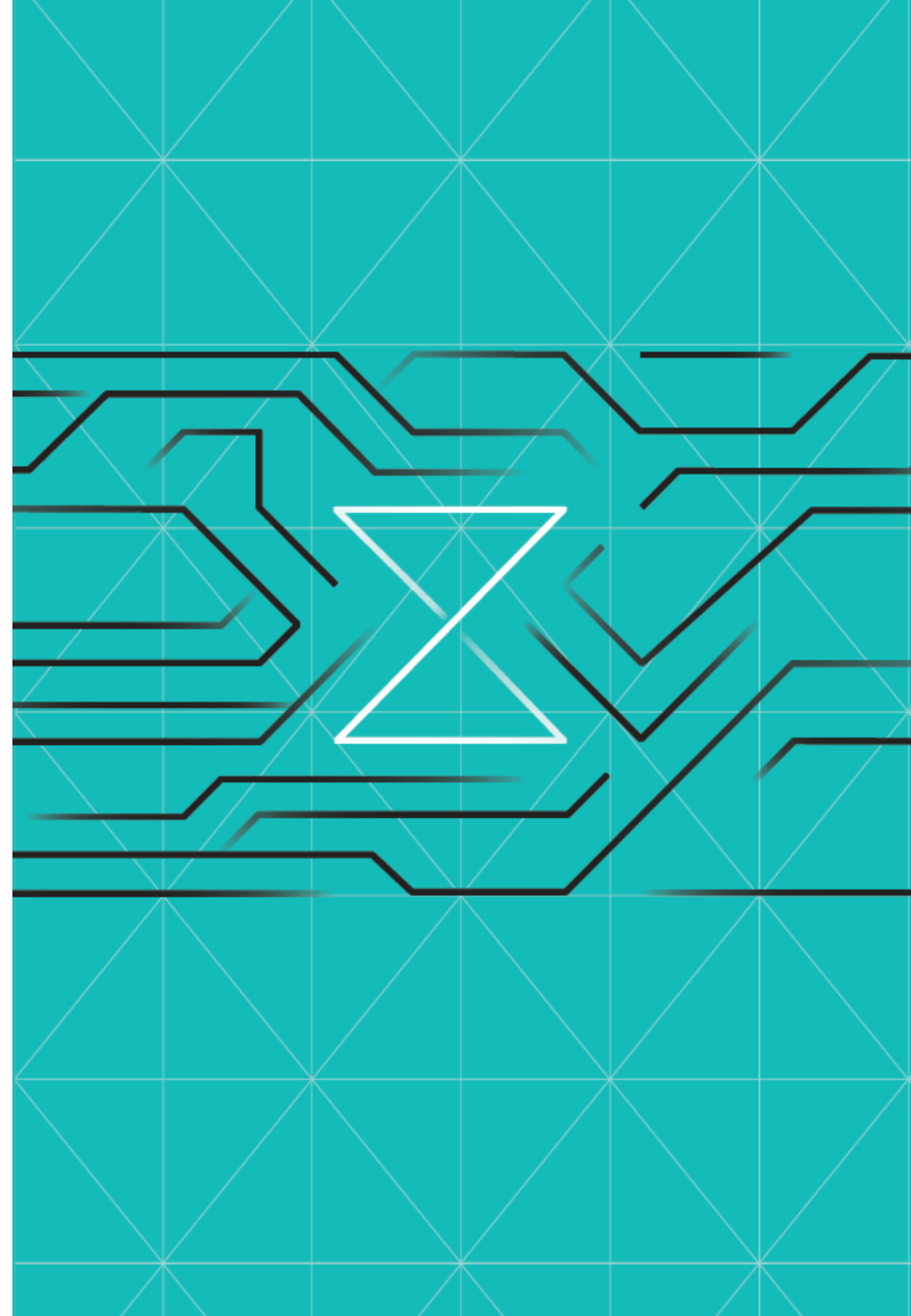


JCL2

ON the Job

- FIX SOME JCL AND GET THAT PROGRAM RUNNING!
- 1 COPY JES2JOB1
- 2 MAKE IT YOUR OWN
- 3 OPEN IT UP!
- 4 RUN THAT CODE
- 5 CHECK THE OUTPUT
- 6 THE JOB DETECTIVE
- 7 MAKE THE NECESSARY UPDATE
- 8 SAVE AND SUBMIT
- 9 CHECK FOR THE CORRECT OUTPUT



FIX SOME JCL AND GET THAT PROGRAM RUNNING!

The Challenge

Let's try to fix a simple problem in some JCL.

The code you'll be investigating is supposed to combine the contents of a few files; but as you'll find out, something is missing.

There are clues in the output that will help, and when you're done, the job will run successfully without any errors.

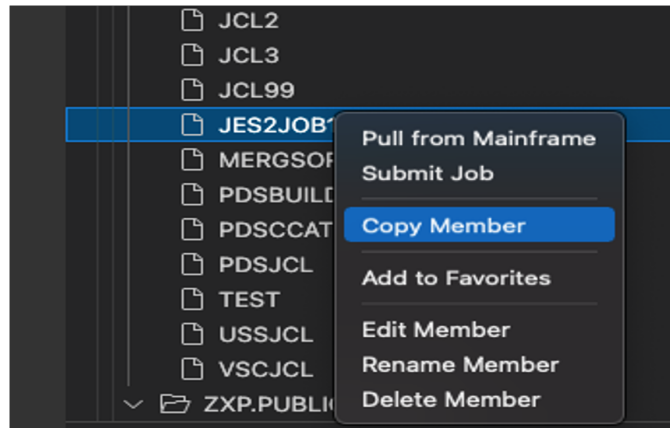
Before You Begin

You should have VSCode fully installed and configured before you begin.

Investment

Steps	Duration
9	45 minutes

1 COPY JES2JOB1



Look in **ZXP.PUBLIC.JCL** for a member called **JES2JOB1**.

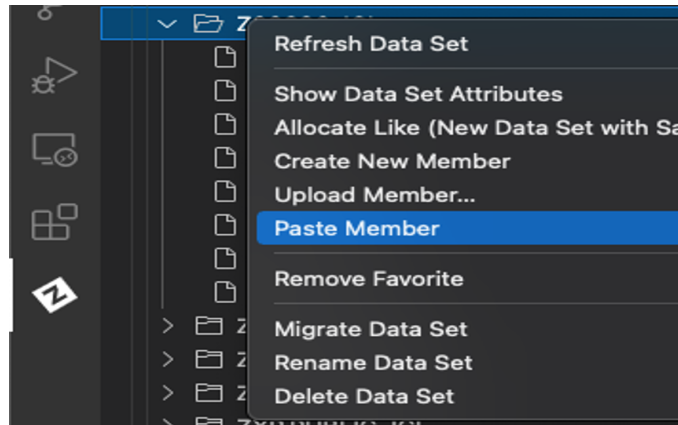
This is the JCL you will be working with for this challenge.

Right-click on it and select "Copy".

2 MAKE IT YOUR OWN

Locate your own ***Zxxxxx.JCL** data set, right-click on it, and select "Paste".

This will, of course, put a new copy of the **JES2JOB1** member into your JCL data set, so you can edit it.



When pasting, you will be prompted to enter a name. Usually, you just want to enter the same name as the source - which is exactly what you should do here.

3 OPEN IT UP!

Click on the **JES2JOB1** member (in your own JCL data set, not the ZXP one) and review the code.

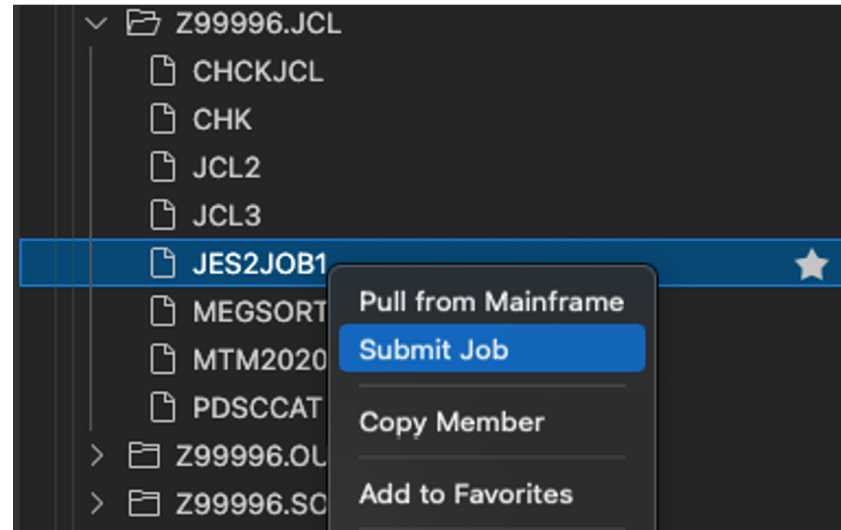
```
1  //JES2JOB1  JOB
2  //COMBINE   EXEC PGM=IRXJCL,PARM='COMBINE'
3  //SYSEXEC   DD DSN=ZXP.PUBLIC.EXEC,DISP=SHR
4  //SYSTSPRT  DD DSN=&SYSUID..OUTPUT(GUITAR),DISP=SHR
5  //SYSTSIN   DD DUMMY
6  //NICKNAME  DD DSN=ZXP.PUBLIC.SOURCE(NICKNAME),DISP=SHR
7  //BRAND     DD DSN=ZXP.PUBLIC.SOURCE(BRAND),DISP=SHR
8  |
```

It looks like it will be running the **IRXJCL** program with a 'COMBINE' parameter.

It will be doing something in your **OUTPUT** data set, and will also be using two members from the **ZXP.PUBLIC.SOURCE** data set.

4 RUN THAT CODE

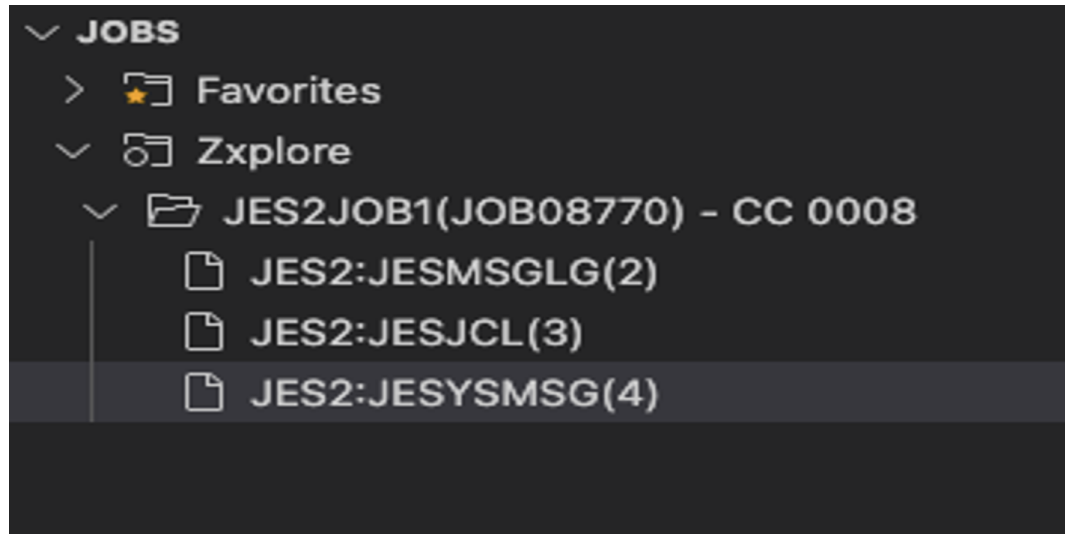
Right-click on on your **JES2JOB1** JCL member and select "Submit Job".



Notice the Job Name that pops up in the lower right corner of your screen. It should start with "JOB", followed by 5 digits.

JCL2 | 230222-1622

5 CHECK THE OUTPUT



In the VSCode left-side navigation area, go to the JOBS section and find the job you just submitted, which matches the job name that popped up in the previous Step.

You may need to reload the JOBS view to find it.

You will notice that it shows a completion code **CC 0008**. That is a “condition code”, and “8” means that there was definitely an error.

(We told you something was missing!)

6 THE JOB DETECTIVE

Look through the output that the failed job created.

```
JES2JOB1.JOB08770.JESYSMSG X
1 ICH70001I Z99996 LAST ACCESS AT 02:01:48 ON MONDAY, AUGUST 23, 2021
2 IEFA111I JES2JOB1 IS USING THE FOLLOWING JOB RELATED SETTINGS:
3 SWA=ABOVE,TIOT SIZE=32K,DSENQSHR=DISALLOW,GDGBIAS=JOB
4 IEF236I ALLOC. FOR JES2JOB1 COMBINE
5 IGD103I SMS ALLOCATED TO DDNAME SYSEXEC
6 IGD103I SMS ALLOCATED TO DDNAME SYSTSPRT
7 IEF237I DMY ALLOCATED TO SYSTIN
8 IGD103I SMS ALLOCATED TO DDNAME NICKNAME
9 IGD103I SMS ALLOCATED TO DDNAME BRAND
10 IRX0555E .....
11 IRX0670E EXECIO error while trying to GET or PUT a record.
12 IEF142I JES2JOB1 COMBINE - STEP WAS EXECUTED - COND CODE 0008
13 IGD104I Z99996.SOURCE RETAINED, DDNAME=SYSEXEC
14 IGD104I Z99996.OUTPUT RETAINED, DDNAME=SYSTSPRT
15 IGD104I Z99996.INPUT RETAINED, DDNAME=NICKNAME
16 IGD104I Z99996.INPUT RETAINED, DDNAME=BRAND
17 IEF373I STEP/COMBINE /START 2021235.0217
18 IEF032I STEP/COMBINE /STOP 2021235.0217
19 CPU: 0 HR 00 MIN 00.01 SEC SRB: 0 HR 00 MIN 00.00 SEC
20 VIRT: 72K SYS: 244K EXT: 464K SYS: 11440K
21 ATB- REAL: 24K SLOTS: 0K
22 VIRT- ALLOC: 12M SHRD: 0M
23 IEF375I JOB/JES2JOB1/START 2021235.0217
24 IEF033I JOB/JES2JOB1/STOP 2021235.0217
25 CPU: 0 HR 00 MIN 00.01 SEC SRB: 0 HR 00 MIN 00.00 SEC
26
```

You may not be able to make sense of much of this, but one line in the **JESYSMSG** section gives a very clear clue about why the program didn't run correctly.

"HOW DO I KNOW WHAT INPUT A PROGRAM NEEDS?"

JCL is often used to run programs on z/OS, and part of running a program is providing that program the files that it needs to run successfully.

The last few lines of the JCL you are working on here are called "DD Statements". **DD** stands for Data Definition, and these statements specify the name of the file being defined, as well as its location (a specific dataset), and how it should be handled (shared, exclusive, existing, new, and so on).

In this case, the program is looking for a third piece of input that was not there in the JCL you were given. You can copy/paste the second DD statement, and then alter it to match what the program is looking for, based on what you discovered in the failing program's output from Step 5.

JCL1230222-1622

7 MAKE THE NECESSARY UPDATE

Edit your copy of JES2JOB1 and try to fix the problem.

There is one missing line of code which you need to add.

You can use the **BRAND** DD Statement as a reference for the line you need to create, but there are two parts of that line that you will need to change to make the job work correctly.

```
1  //JES2JOB1  JOB
2  //COMBINE   EXEC PGM=IRXJCL,PARM='COMBINE '
3  //SYSEXEC   DD DSN=ZXP.PUBLIC.EXEC,DISP=SHR
4  //SYSTSPRT  DD DSN=&SYSUID..OUTPUT(GUITAR),DISP=SHR
5  //SYSTSIN   DD DUMMY
6  //NICKNAME  DD DSN=ZXP.PUBLIC.SOURCE(NICKNAME),DISP=SHR
7  //BRAND     DD DSN=ZXP.PUBLIC.SOURCE(BRAND),DISP=SHR
8                                     ,DISP=SHR
```

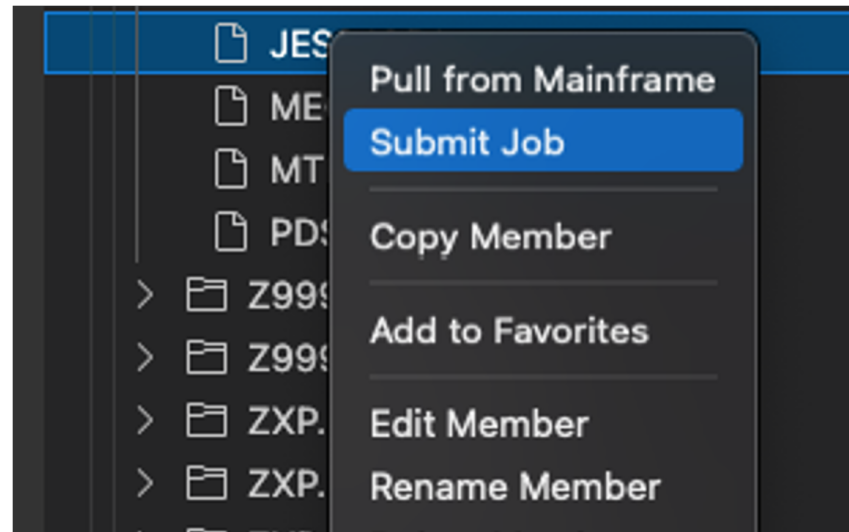
If you want an additional clue, look in the ZXP.PUBLIC.SOURCE dataset to see what other files in there might be helpful for this program.

And remember, spacing matters! Make sure those columns line up.

8 SAVE AND SUBMIT

Make sure to Save a dataset or member whenever you have made changes.

If you don't, and try to submit it again, it will just keep submitting the old version without your updates.



Once you have saved your changes, right-click and select "Submit Job", again noting the number that pops up so you can easily find it in the JOBS section.

9 CHECK FOR THE CORRECT OUTPUT

```
1
2  Legendary Guitars
3
4  Item                Brand                Artist
5  -----
6  Blackie             Fender Strat. (custom)    Eric Clapton
7
8  Cloud               Custom build              Prince
9
10 Lucille             Gibson (almost burned)    B.B. King
11
12 Frankenstrat        Fender/ Gibson Frankencaster  Eddie van Halen
13
14 Monterey Stratocaster  Fender Strat. Flower power  Jimmie Hendrix
15
16 Concorde            Jackson Flying V          Randy Rhoads
17 |
```

When corrected, the job should produce output in your OUTPUT data set called JES2JOB1, similar to what you see above.

Look through the output and hopefully you got a completion code of **** CC 0000****, indicating no more problems.

If not, review Step 7, and make sure you are saving and submitting the correct copy of the JCL.

If you got "CC 0000", open up **ZXP.PUBLIC.JCL** and submit the **CHKJCL2** validation job.

Nice job - let's recap	Next up ...
<p>Congratulations on fixing your first JCL program.</p> <p>A z/OS Sysprog will spend a lot of time hunting down bugs in code.</p> <p>Sometimes the error is just a misunderstanding between what the programmer <i>meant</i> to do, and how it actually got executed by the system. Other times, it's just a simple slip of the finger that turned an M into an N, or put an extra space between two fields.</p> <p>We've all been there, and will go there again!</p>	<p>You've got data sets, you've got jobs. Why not go wild and try some more Unix System Services?</p> <p>You've done USS1, try out USS2 and you'll be ready to handle even more mainframe tasks.</p>