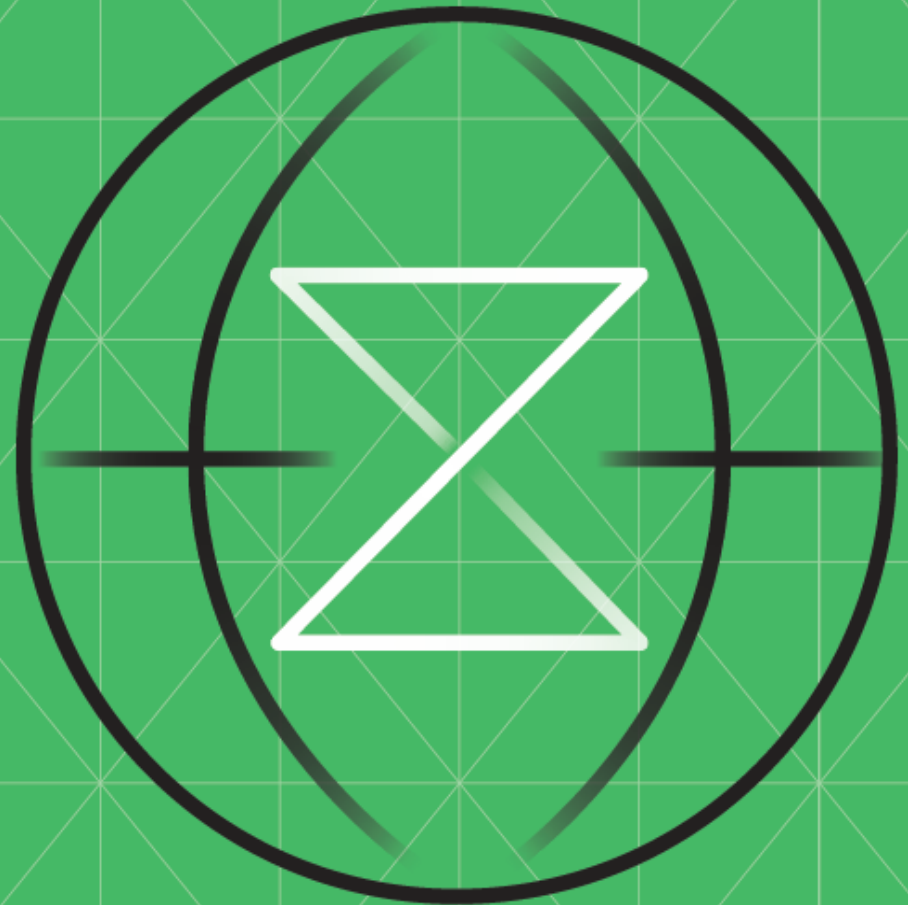


JCL3

Explore and edit JCL, check it in SDSF

- [UNDERSTAND, SUBMIT, FIX, PROGRESS](#)
- [1 UPDATE YOUR DATASETS](#)
- [2 GET FAMILIAR](#)
- [3 WHAT'S THE DIFFERENCE?](#)
- [4 SEE THE OUTPUT GOAL](#)
- [5 SUBMIT THE JCL](#)
- [6 VIEW YOUR JOB SUBMISSION](#)
- [7 SO WHAT WENT WRONG?](#)
- [8 FIND, FIX, REPEAT](#)
- [9 ALL FIXED - SUBMIT](#)



UNDERSTAND, SUBMIT, FIX, PROGRESS

Let's continue with what you can with 3270 terminal interfaces.

The Challenge

The key points for this challenge:

- Understand how to use JCL using TS0/IPSF.
- Submit jobs with errors and use SDSF to determine the errors.
- Edit/fix JCL to successfully submit a job.

Before You Begin

Make sure you have completed TS0 4 (and the previous TS0 challenges) before beginning TS05.

Investment

Steps	Duration
9	90 minutes

1 UPDATE YOUR DATASETS

```
----- ISPF Primary Option Menu -----
Option ==>
0 Settings      Terminal and user parameters
1 View          Display source data or listings
2 Edit          Create or change source data
3 Utilities     Perform utility functions
4 Foreground    Interactive language processing
5 Batch         Submit job for language processing
6 Command       Enter TSO or Workstation commands
7 Dialog Test   Perform dialog testing
8 LM Facility   Library administrator functions
9 IBM Products  IBM program development products
10 SCLM         SW Configuration Library Manager
11 Workplace    ISPF Object/Action Workplace

----- Other Install Products -----
SD SDSF        System Display and Search Facility
IP IPCS        Inter Problem Control Facility
IS ISMF        Inter Storage Management Facility
SM SMP/E       SMP/E and CBIPO Dialogs
HC HCD         HW Configuration Definition Dialog
```

Log on to TSO and ISPF. On the Option line, enter `tso submit 'ZXP.PUBLIC.JCL(chalcopy)'`.

This will ensure that you have the correct datasets and members under your userid.

Check to make sure you now have the **WSLJCL** member in your **JCL** dataset.

Hint: You will have to go in and out of a few panels to get there.

2 GET FAMILIAR

Menu	Functions	Confirm	Utilities	Help
VIEW	ZXP.PUBLIC.DATA			
Command	===>			
Name	Prompt	Size	Created	
CAPACITY		12	2022/03/06	
CITY		12	2022/03/06	
COACH		12	2022/03/06	
STADIUM		12	2022/03/06	
STATE		12	2022/03/06	
TEAM		12	2022/03/06	
VSAMCARS		1000	2022/03/14	
End				

Now that you have the files you need, get yourself familiar with the data you will be working with.

The topic for the challenge: *The USA Women's Soccer League*

Get to the member list of **ZXP.PUBLIC.DATA**, and type **B** to browse next to each member.

Take a look at the six different files - you will be using the information inside of them.

3 WHAT'S THE DIFFERENCE?

Menu Functions Confirm Utilities Help					
VIEW	ZXP.PUBLIC.EXEC				Re
Command ===>					
	Name	Prompt	Size	Created	
	COMBINE		22	2021/08/20	2021/12
	REXXDBC		77	2021/12/23	2021/12
	REXX2		1500	2021/08/27	2021/08
	RXPDSDIR		58	2021/09/03	2021/09
	RXSQLSMP		53	2022/01/05	2022/01
	RXSYSLOG		18	2021/12/03	2021/12
	WSL		48	2022/03/06	2022/03
	WSL\$		49	2022/03/06	2022/03
	End				

Time to produce some output with these files. You will be doing this using JCL and running a job.

Before you run the job, take a look at what the output will look like first using **WSL\$** in **ZXP.PUBLIC.EXEC**

The WSL\$ and WSL REXX execs are essentially the same. The difference being that WSL\$ has already allocated the physical dataset name to the logic file name.

Your task: take the WSL REXX and run a job using JCL to produce the same output as WSL\$.

4 SEE THE OUTPUT GOAL

```
USA Women Soccer League
Team      Angel City FC
City      Los Angeles
State     California
Stadium   Banc of California Stadium
Capacity  22000
Coach     Freya Coombe
*****

Team      Chicago Red Stars
City      Bridgeview
State     Illinois
Stadium   SeatGeek Stadium
Capacity  20000
Coach     N/A
*****

Team      Houston Dash
City      Houston
State     Texas
Stadium   PNC Stadium
Capacity  7000
Coach     James Clarkson
```

Run WSL\$ to understand what your job submission SHOULD look like.

Type **ex** next to WSL\$ execute the program and view output.

You will be using JCL to run your WSL exec.

The trick? The JCL is not be perfect yet, and you will need to determine why.

Submit the WSCJCL job, which uses WSL REXX logic, to see what errors are wrong.

5 SUBMIT THE JCL

```
File Edit Edit_Settings Menu Utilities Compilers
VIEW      Z99994.JCL(WSLJCL) - 01.00
Command ==> sub
***** ***** Top of Data *****
==MSG> -Warning- The UNDO command is not available until
==MSG> your edit profile using the command RECO
000001 //WSLJCL      JOB
000002 //COMBINE    EXEC PGM=IRXJCL,PARM='WSL'
000003 //SYSEXEC     DD DSN=ZXP.PUBLIC.EXEC,DISP=SHR
000004 //SYSTSPRT     DD SYSOUT=*
000005 //SYSTSIN     DD DUMMY
000006 //IDIREPT     DD DUMMY
000007 //TAAM         DD DSN=ZXP.PUBLIC.DATA(team),DISP=SHR
000008 //CITY         DD DSN=ZXP.PUBLIC.DATA(CITY),DISP=SHR
000009 //STATE        DD DSN=ZXP.PUBLIC.DATA(STATE),DISP=SHR
000010 //STADIUM       DD DSN=ZXP.PUBLIC.DATA(STADIUM),DISP=SHR
000011 //CAPACITY     DD DSN=ZXP.PUBLIC.DATA(CAPACITY),DISP=SHR
000012 //COACH        DD DSN=ZXP.PUBLIC.DATA(COACH),DISP=SHR
***** ***** Bottom of Data *****
```

Find the JCL that you need to submit.

Go to your JCL dataset, find **WSLJCL** and type **s** for Select.

Run the JCL by typing `submit` in the command line.

At bottom of the screen, you will see 'JOB WSLJCL SUBMITTED' with a job name associated.

6 VIEW YOUR JOB SUBMISSION

```
Display Filter View Print Options Search Help
-----
SDSF STATUS DISPLAY ALL CLASSES
COMMAND INPUT ==>
PREFIX=* DEST=(ALL) OWNER=Z99994 SYSNAME=
NP  JOBNAME  JobID   Owner   Prty Queue   C   Pos  S
    Z99994   TSU02638 Z99994   15 EXECUTION  A   514  S
    CHALCOPY JOB02425 Z99994    1 PRINT      A   521
    CHALCOPY JOB02433 Z99994    1 PRINT      A   644
    Z99994   TSU02625 Z99994    1 PRINT      A   657
    WSLJCL   JOB02641 Z99994    1 PRINT      A   657
```

Check the submission. You have done this before in VSCode and in TS04 you do it with SDSF.

Type `=sd;st` on the command line. This is a short cut to getting to SDSF and then checking the status of jobs.

Optionally, you can F3 back to the main menu and go to SDSF that way.

CAN I EDIT MY JCL WITHOUT ANY PENALTY OF MESSING IT UP?

If you are looking at a job submission in SDSF and want to quickly edit the job right there, you can do so by typing **SJ** next to the job in the DA, ST, and O views.

This allows you to edit the job in SDSF without any permanent changes in your JCL file.

That being said, this is only temporary so any changes you make in SDSF will not stick.

You will still have to go back into your JCL member and save the edits there.

Note that you can also make changes to the source JCL in the ISPF editor, submit the JCL and then **Cancel** to leave the source unchanged. You *cannot* do this with the Zowe extension in VSCode.

7 SO WHAT WENT WRONG?

```
Display Filter View Print Options Search Help
-----
SDSF STATUS DISPLAY ALL CLASSES L
COMMAND INPUT ==>
PREFIX=* DEST=(ALL) OWNER=Z99994 SYSNAME=
NP  JOBNAME  JobID   Owner   Prty Queue   C   Pos
    Z99994   TSU04467 Z99994   15 EXECUTION
    ?_ WSLJCL   JOB04470 Z99994   1 PRINT     A   628
```

Check the job name **WSLJCL** and see if it was successful.

View the different joblog files using **?** as a command ("action character") next to the job.

You can view the entire joblog as one big file by typing **s** as the action character.

Use F11 to shift to the right and view if the job you submitted for **WSLJCL** was successful or not.

What did you find?

Hint: in the list views (DA, ST, O) keep scrolling to the right (F11) until you find a column that says, 'Max-RC'. You may scroll for a bit. Enter right 40 on the command line to scroll faster.

8 FIND, FIX, REPEAT

```
Display Filter View Print Options Search Help
-----
SDSF JOB DATA SET DISPLAY - JOB WSLJCL (JOB04470) LI
COMMAND INPUT ==>
PREFIX=* DEST=(ALL) OWNER=Z99994 SYSNAME=
NP DDNAME StepName ProcStep DSID Owner C Dest
   JESMSG LG JES2          2 Z99994 X LOCAL
   JESJCL  JES2          3 Z99994 X LOCAL
   s_ JESYSMSG JES2        4 Z99994 X LOCAL
```

Scroll through the various files of the joblog to understand the error in the JCL and what went wrong.

Once you have identified the error, go back to the JCL file in your JCL dataset, type **e** for edit the member, and make the necessary edits to fix the JCL.

Submit the job again and repeat the cycle until you have successfully completed the job.

Hint: there may be more than one error (what do you think?), so keep repeating until the job is successful.

9 ALL FIXED - SUBMIT

```
Display Filter View Print Options Search Help
-----
SDSF OUTPUT DISPLAY WSLJCL JOB04486 DSID 101 LINE 0
COMMAND INPUT ==> _
***** TOP OF DATA *****
USA Women Soccer League
Team      Angel City FC
City      Los Angeles
State     California
Stadium   Banc of California Stadium
Capacity  22000
Coach     Freya Coombe
*****
Team      Chicago Red Stars
City      Bridgeview
State     Illinois
Stadium   SeatGeek Stadium
Capacity  20000
Coach     N/A
```

You will know when the job is successful when you see a **SYSTSPRT** member in the joblog, and the output is the same as running the WSL\$ exec on its own.

Once again, you might have more than one error to fix, so it might take a few cycles to get the job to “run clean”.

Once the job is successful, get credit for this challenge: submit '**ZXP.PUBLIC.JCL(CHKTS05)**' using whichever method suits you.

JCL3/230901-1113

Nice job - let's recap	Next up ...
<p>You have successfully navigated ISPF while editing a job and checking its submissions until it was correct.</p> <p>You have done this before in previous JCL challenges but this time you did it on TSO instead of VS Code.</p> <p>This is something you will be doing a lot in TSO, and it is important to know what to look for in submission errors and how to quickly get back to your JCL to fix the errors.</p> <p>Overall, you have continued to practice your TSO/ISPF navigation skills and will continue to become more familiar with its functionality.</p>	<p>Go check out the other Extended and Advanced modules for your continued learning journey on IBM Z Xplore.</p>