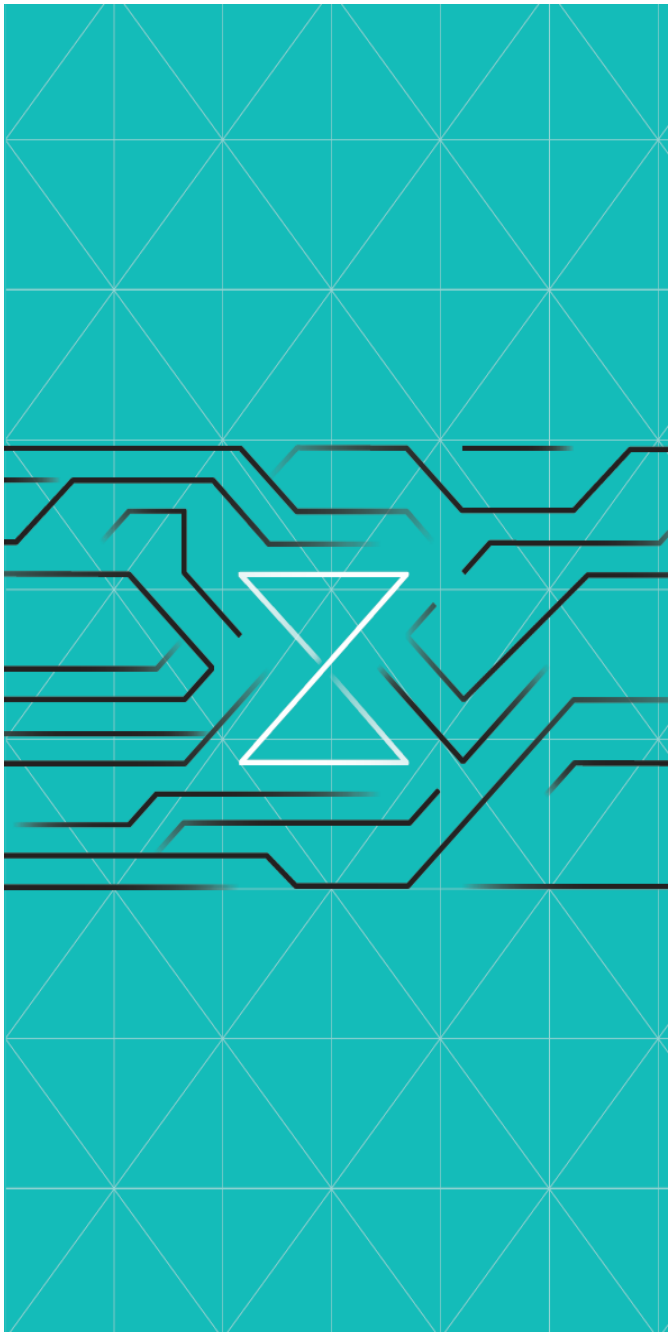


Advanced

250123-1116

PDS2

Thankx for the Memberies

250123-1116

ADVANCED FILTERING AND EDITING OF DATA SETS AND MEMBERS

The Challenge

In z/OS, data is typically organized in structures called Data Sets.

The concept is similar to the way you use files and folders on your personal computer, with a few very important differences, which you will learn more about here.

In this challenge, you will perform some basic operations around Datasets and Members, and when you are finished, you'll run a job to process some of those dataset members.

Before You Begin

Make sure your Visual Studio Code environment is all set up and connected to the z/OS system. Other than that, nothing else is required!

Investment

Steps	Duration
7	30 minutes

PDS2I250123-1116

1 THESE ROCK FILES ROCK!

You are going to experiment with more things to do with PDS library members.

Download the following files associated with this assignment to your computer in a convenient location.

- rocks1.txt
  - rocks2.txt
    - rocks3.txt

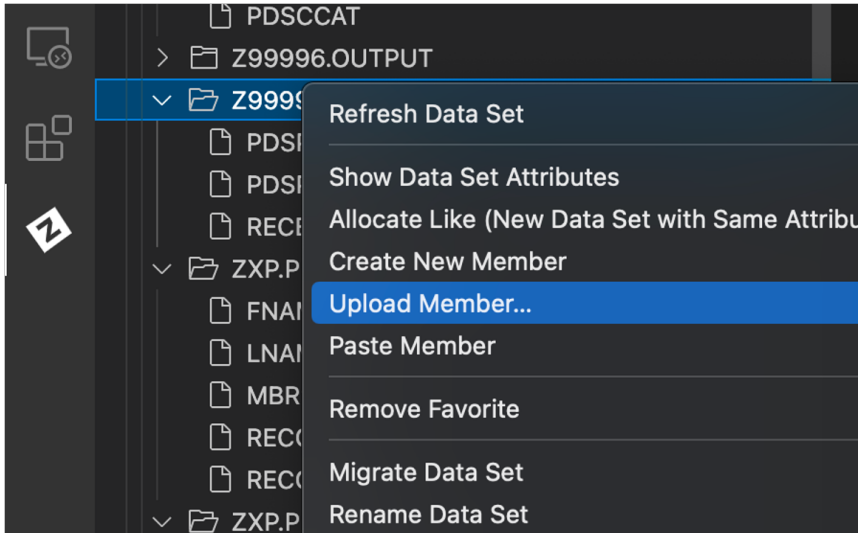
PDS2I250123-1116

2 UPLOAD THE ROCKS

Now upload these files into your dataset collection.

Right-click on the **SOURCE** dataset and select “Upload member...”.

You should be able to select multiple files, so go ahead and select all of the rocks*.txt files. Remember you may need to refresh the dataset view before they show up.



(In the screenshot above, the SOURCE PDS has been added to the list of favorites.)

3 ONTO THE NEXT JOB

Locate **MERGSORT** in the **ZXP.PUBLIC.JCL** dataset and copy/paste that into your **JCL** dataset.

NOTE: Each qualifier (the part between the dots) of a dataset name can be 1-8 characters in length, so we sometimes have to get creative with making names fit into 8 characters.

If you are having trouble copying MERGSORT, check the spelling.

Now open it up and take a look at what is inside.

```
1   //MERGSORT JOB 1,NOTIFY=&SYSUID                                  00010000
2   //SORT     EXEC PGM=SORT                                         00020000
3   //SYSOUT   DD SYSOUT=*                                           00030000
4   //SORTIN   DD DSN=&SYSUID..SOURCE(ROCKS1),DISP=SHR               00040002
5   //         DD DSN=&SYSUID..SOURCE(ROCKS2),DISP=SHR               00050002
6   //         DD DSN=&SYSUID..SOURCE(ROCKS3),DISP=SHR               00060002
7   //SORTOUT  DD DSN=&SYSUID..OUTPUT(ROCKSOUT),DISP=(SHR),          00070000
8   //         LIKE=ZXP.PUBLIC.SOURCE                                00080000
9   //SYSIN    DD *                                                  00090000
10    SORT FIELDS=(1,20,CH,A)                                       00100001
11
```

Figure out what the JCL code is going to do.

A big hint is that when the JES (Job Entry Subsystem) processor sees **&SYSUID.**, it will substitute in your userid, so **&SYSUID..ROCK1** would be interpreted as Zxxxxx.ROCK1.

These variables are known as “Symbolics”, and they come in handy when writing shared JCL scripts where certain values need to change from user to user, or system to system.

4 PREDICT THE OUTPUT

```
//SORTIN   DD DSN=&SYSUID..SOURCE(ROCKS1),DISP=SHR
//         DD DSN=&SYSUID..SOURCE(ROCKS2),DISP=SHR
//         DD DSN=&SYSUID..SOURCE(ROCKS3),DISP=SHR
//SORTOUT  DD DSN=&SYSUID..OUTPUT(ROCKSOUT),DISP=(SHR),
//         LIKE=ZXP.PUBLIC.SOURCE
```

Look at line #8 of the program and notice where it's putting its output.

Can you guess what the dataset member name will be after the Symbolic substitution takes place?

Now submit the MERGSORT job.

5 MAKE YOUR OWN DATASET

We have pre-allocated a number of datasets for you, like OUTPUT, INPUT, JCL and PDS; but you can make your own datasets very easily right through VSCode.

Select the “Create New Data Set” by right-clicking on your connection profile.

Give it a name that starts with your HLQ (your userid), then **PDS2**, followed by any third qualifier 8 characters or less, that isn’t already used. For example, Zxxxxx.PDS2.ANEWPDS or Zxxxxx.PDS2.PUMPKIN.

Select in the next step Partitioned Data Set: Default, and finally select Allocate Data Set.

6 CHECK THE OUTPUT

Refresh the folder view of your **OUTPUT** dataset.

Click on the new **ROCKSOUT** member and observe what the job did with the three input files.

Rename the ROCKSOUT member to **PDS2OUT**.

Nearly done!

7 VALIDATE

All done? Think you got this?

Lastly, delete any **PDS2.*** datasets you created earlier in this challenge.

To finish up this challenge, submit the **CHKAPDS2** validation job from **ZXP.PUBLIC.JCL**.

ADVANCED DATA SET TIPS AND TRICKS	Next up ...
To keep things simple, Zowe hides a lot of the complexity behind the creation of data sets and members. In reality, there are many, many different options being selected by default. You can change these by using the “Zowe-Default-Datasets” options in settings.json, with a full walkthrough of this and a few other nice tricks described in this article: How to Automate z/OS Data Set Allocation Using Zowe & JSON	Make some tacos? Tend to your rock collection? The possibilities are endless, but now is a great time to jump into JCL and learn just how programs run in z/OS.