JCL (Job Control Language)

PREPARED BY MVSQUEST TEAM

Date	Author	Version
03-Nov-2012	MVSQuest Team	Draft Version
13-Nov-2012	MVSQuest Team	Version 1.0

FOR ANY CORRECTIONS/ADDITIONS, SEND MAIL TO MVSQUEST@YAHOO.COM

If the questions shared by you found interesting by the team, the same will be added in the next version with your name suffix.

We recommend reading our book 'Mainframe Refresher Part-1' than reading FAQs. This book covers all the concepts in a quick and sufficient style. There can be 'n' number of questions on the same concept. Hence understanding the concept would serve the purpose rather than reading FAQ that has wide scope depending on every interviewer's expertise.

JOB CONTROL LANGUAGE

What is JCL?
JCL is Job Control Language. It defines the work to be executed by the OS. It is used to execute the programs and it itself is not a programming language.

What are the primary statements used in JCL?
JOB statement indicates start of the job definition. This should be the first statement in any JCL. EXEC statement defines the program to be executed. DD statements defines the input and output files of the program. There could be only one JOB statement. There can be 1 to 255 EXEC statements in one JOB and there can be 0 to 3273 DD statements under every EXEC statement.
The below is the list of commonly used statements.

```
 JOB              : Identifies a job and supplies accounting info
 EXEC             : Identifies a job step by indicating the name of the program to be executed.
 DD               : Identifies a data set to be allocated for the job step
 Delimiter(/*)    : Marks the end of an in-stream dataset
 Null(//)         : Marks the end of a job
 Comments(//*)    : Provides Comments
 PROC             : Marks the beginning of a procedure
 PEND             : Marks the end of a procedure
 OUTPUT           : Supplies options for SYSOUT processing.
```

What indicates the start and end of the JOB?
JOB statement indicates start of the job and null statement (line that has only // in first two columns) indicates end of the job. Any EXEC or DD coded after null statement will not be sent for execution.

Brief the coding sheet of JCL

```
//NAME   OPERATION  OPERAND  COMMENT
```

NAME field can be 1-8 chars length and start in third column following // and first character should be alphabet and rest can be alpha numeric or national character.
OPERATION field indicates type of statement like JOB/EXEC/DD/PROC etc. It follows NAME field with a leading space.
OPERANDS are parameters for the operation and they follow OPERATION field with leading space.
COMMENT field is optional and it follows OPERANDS by a leading space.

How to continue a statement when the parameters exceed more than 72 columns?
The operand parameters are separated by comma. End the line to be continued before 72 with a parameter ending with comma and continue with next parameter on next line anywhere between 4 and 16. The first two columns of the continued line should be //.

What are the two types of parameters in JCL statements?
They are Positional and Keyword parameters. Positional parameters are characterized by their position in relation to other parameters. The Key-word parameters are independent of their position and identified with key-word followed by '='. If a statement has positional parameters, they should appear first before any keyword parameters as they get meaning only by their position. Even a keyword parameter can have positional parameters inside it like DISP parameter. Absence of preceding positional parameter should be specified to the system by a comma so that following positional parameter if any, can get its meaning rightly.

Can we say any parameter that is of format XXXX=YYY as keyword parameter?
Generally YES. All keyword parameters have '='. Exception to this rule is PGM= and PROC=. Though they have '=' they should immediately appear after the NAME parameter and hence positional. Remember by definition keyword parameters can be coded anywhere within the definition of the statement.

What is *COMMENT FIELD* and **COMMENT LINE**?

Comment field is specified operand field to write comments for the line and comment line(s) are used to write comments for Job/Steps. It starts with //*.

//SHRUTHI JOB (1234),'MS',CLASS=A
//* THIS IS A COMMENT LINE
//STEP01 EXEC PGM=IEFBR14   *THIS DELETES DATASETS*

JOB Statement and its Parameters

What are the parameters generally coded on JOB statement and how many positional parameters are there in JOB statement?

Following are some of the parameters that are used on JOB statements.

Account, Name, Class, Region, COND, MSGCLASS, MSGLEVEL, TYPRUN, PRTY and Region etc. Out of these Account, Name parameters are the two positional parameters and are mandatory and remaining are key word parameters.

What is the significance of CLASS parameter?

Every CLASS (A-Z, 0-9) has defined with its own priority and limitations. The definition is totally installation specific. Production class that has high priority in our installation is defined for 'P' and test jobs should not be submitted with this class. Not just priority – classes usually associated with CPU time and number of tapes it can use at maximum and many more.

How JES prioritises the jobs submitted at the same time?

The job of CLASS that is defined with top priority is given first priority. If all jobs are submitted with same class, the PRTY parameter if any coded on JOB decided priority. If that is also not there, then the first job reached JES will get priority (the earlier job number assigned).

What is the significance of MSGCLASS parameter?

It is a keyword parameter which specifies the output class to which system messages for your job are to be routed. Output class is an alphabetic (A thru Z) or numeric (0 thru 9) character.

Explain JOB statement.

Jobname, (acctnum,acctname) (positional parms),'Name' (positional parm), CLASS= , PRTY= , MSGCLASS= , MSGLEVEL= , TYPRUN= , NOTIFY= . CLASS and MSGCLASS can have one of the 36 values, A-Z and 0-9. Class depends on the amount of CPU time required and use of tapes by the job. MSGCLASS specifies the job scheduler message output class. TYPRUN can be SCAN or HOLD. PRTY is used to specify the priority of the job, smaller number indicates higher priority (allowed values are 0-15 for JES2 and 0-14 for JES3). MSGLEVEL have two positional parameters stmts, mesgs; Stmts of '0' indicates print only Job statements, '2'-print all JCL and JES statements and Messages, '3'-only JCL & JES statements print; Mesgs of '0' only JCL messages print; if the job ABENDs, then JES messages too, '1'- JCL and JES messages print. NOTIFY is used to notify the user about the completion of Job.

What is the significance of TYPRUN parameter in JCL?
How do you check syntax of a JCL without running it?

If TYPRUN=SCAN is coded, it checks job for syntactical errors.
If TYPRUN=HOLD is coded, holds the Job without running after successful syntax verification. This job can be run only after operator releases the Job.

How the operator/programmer can release the job that is submitted with TYPRUN=HOLD?

By typing 'A' against the job in the spool

What is the significance of Region Parameter on Job Card?

To override the installation default Region size for executing your job, Region parameter will be used. REGION=0M parameter again allocates maximum space that is available in the system to execute the job. If the installation does not have IEALIMIT or IEFUSI that sets the maximum limit, then REGION=0 may cause storage issues and should be avoided.

Which memory is allocated for the Job when you request REGION=100K?

By default, only virtual memory is allocated. If you need central storage memory, you need to code ADDRSPC=REAL in the job card.

What is the parameter to be passed in the job card for the unlimited time, irrespective of the job class?

TIME=1440

What does the TIME parameter signify ? What does TIME=1440 mean ?

TIME parameter can be used to overcome S322 abends for programs that genuinely need more CPU time. TIME=1440 means no CPU time limit is to be applied to this step.

What is MSGLEVEL parameter?

It is a keyword parameter that has two positional parameters in it. First parameter requests the details to be in the JESJCL and the second parameter requests the details to be in JESYSMSG of JOB Log. MSGLEVEL=(1,1) provides the maximum information in JOB Log.
MSGLEVEL=(statements, messages)
Statements:
   0 - only the job statement is to be written;
   1 - all input control statements, cataloged procedure statements and the internal representation of procedure statement parameters after symbolic parameters substitution are to be written;
   2- only input job control statements are to be written;
Messages:
   0 - No allocation/termination messages are to be printed unless the job terminates abnormally;
   1 - All allocation/termination messages are to be printed.

How do you Restart a step in JCL?

Using the RESTART=step name command in the Job statement you can restart the job from a particular step.

How to send job end notification to multiple user-Ids?

This is not possible directly. NOTIFY=&SYSUID sends notification to the user who submitted job. You cannot pass more than one ID in NOTIFY clause of JOB statement. Alternatively, you can have IKJEFT01 STEP at the end of the job that can issue SEND message to multiple users. But the limitation is this could say only job completion but not of RC/Abend etc.

Though I have coded NOTIFY=&SYSUID, I am not getting job notifications. Why?

1. Either your job card have USER and PASSWORD of other ID. In such case, the USERID specified in USER is a submitter and the message will go to him only. By Changing NOTIFY=&SYSUID to NOTIFY=SHRUTHI (Assuming SHRUTHI is your ID), you can sort out this.
2. You could be out of INTERCOM. To enable intercom, issue the command TSO PROFILE INTERCOM from ISPF command line.

EXEC Statement and its Parameters

Explain EXEC statement with the various parameters used.

The PGM parameter specifies the Program to be executed. ACCT parameter is used to specify the accounting number of the Step, if it is different from the Job accounting number. PARM is used to send values to the program, when it is executed. REGION specifies the amount of storage a Job step can use. DPRTY is used to specify priority to the step. COND specifies the condition for executing subsequent Job step. TIME sets a CPU time limit for a Job step. Ex. //step1 exec pgm=prog1,parm='aa',region=64k

How do you pass data to the program as the job is being executed?

1. By using 'PARM' parameter in the EXEC statement
2. Through instream data using SYSIN DD *
3. Files or any other common mechanism like database etc

I would like to use 120 characters of data from JCL to Program. What are the options?

As only maximum of 100 bytes can be sent through PARM, the options for this requirement would be either thru SYSIN DD * or files.

What are parameters you can specify on Job statement as well as on exec stmt ?

TIME, COND, REGION and ADDRSPC

What is the significance of TIME parameter?

Time parameter is used to specify the amount of CPU time that a job or job step is permitted to use before it terminates based on where it is coded JOB or STEP respectively.

What would happen if TIME is specified on both JOB as well as EXEC statements?

If the TIME specified on STEP is less than what is specified in JOB, STEP time will override what is coded on Job. If TIME specified in STEP is greater than what is specified in JOB, JOB time will override STEP time.

What would happen if REGION is specified on both JOB as well as EXEC statements?

REGION value coded in JOB overrides the one coded on STEP.

What would happen if COND is coded on both JOB as well as EXEC statements?

Both will be effective. When any of the COND coded in Job is true, rest of the job steps in the job is flushed. When one of the COND coded in STEP is true, only that particular step is flushed.

Conditions

What are the options in JCL if I need to execute the second step based on first step return code/abend?

1. Specify COND parameter in STEP02 and code conditions that check against STEP01. STEP02 will be executed if the COND code is FALSE.
2. Code IF THEN ENDIF for STEP02. If the condition coded is true, STEP02 will be executed.

Though 'IF' clause is easy as it works like in other languages, you would see only COND in most of the production JCLs as 'IF' is relatively new feature.

How to code COND parameter?

It is a keyword parameter which can be specified on the JOB or EXEC statements. COND consists of 3 sub parameters - code(0 thru 4095),logical operator and stepname of which to be compared.
STEPNAME cannot be coded while COND is coded in JOB whereas it is optional in EXEC. Without STEPNAME, the check will happen against all the already executed steps.

How do you specify a COND parameter for a job step so that the step will never be executed?

Code some condition that would always true like COND=(0,LE) or COND=(4095,GE).
As no step can return negative return code or any code greater than 4095, the above conditions are always true and so the steps that have these condition code will never be executed.

What does COND=ONLY and COND=EVEN mean?

ONLY means that this job step will be executed only if any of the previous steps had abnormally terminated. EVEN means that this jobs step will be executed even if a previous step abnormally terminated.

How many conditions can be specified in COND and what will be if only two out of 5 conditions coded are true?

Maximum 8 conditions can be coded and if any one of the conditions is TRUE, the STEP will be flushed

What is the difference between the COND parameter specified on the EXEC statement and the one specified on the JOB statement?

COND parameter specified on EXEC statement determines whether step should be bypassed or executed. COND parameter defined on the JOB statement will determine whether Job should be terminated at a certain point or continued. When COND parameter is defined on the JOB statement, the system evaluates condition specified in the COND parameter before beginning of any job step and compares the code specified in the COND parameter with the return code of all previous jobsteps. If that condition is true, the rest of the job steps are bypassed.

How to bypass the first step using COND?

Code COND=ONLY for the first EXEC.

STEP03 should be executed only if STEP02 is executed and that is independent of return code of the STEP02. STEP04 should be executed if STEP03 is abended with S0C7 abend only. How do you code COND for this?

This is directly possible in COND. COND can do checks against return code only and no options like whether executed or not. Similarly COND=ONLY will ensure a STEP to run in case of abends but you cannot check against particular abend codes. IF solves these problem as below.

1. //CHK IF STEP02.RUN=TRUE THEN
2. //CHK IF STEP03.ABENDCC=S0C7 THEN

In case of RESTART, how the condition code checks against prior steps are evaluated?

Any condition code check coded against the STEP that is not executed in the RESTART run is considered as DUMMY check and ignored. This is not true for the job runs under Scheduler product.

DD Statement and its Parameters

Explain the DD statements

DD statement is used to defined the input and output for the program executed. Commonly coded parameters in DD statement are DSN, DISP, SPACE, DCB, UNIT, VOL and LABEL.

What is importance of DSN parameter?

It names the dataset to be accessed or created. It can be 1-44 characters of length. This has many qualifiers delimited with period. Ex: PROD.MVSQUEST.CHENNAI.EMPLOYEE.

What will happen if DSN is not specified?

If the dataset is output, then automatically a temporary dataset will be created and will not be available after step execution. If the dataset is input, then you get error/abend.

How many positional parameters are there to a DISP statement and what are their uses?

DISP has there are three parameters (DISP=(Parm1,Parm2,Parm3)).
Parameter 1: This specifies current data set disposition – Valid Values are NEW,OLD,SHR or MOD
Parameter 2: This instructs what the system to do with the dataset if the step normally completes –
Valid Values are CATLG, UNCATLG, DELETE, PASS or KEEP
Parameter 3: This instructs what the system to do with the dataset if the step abnormally ends –
Valid Values are CATLG, UNCATLG, DELETE or KEEP

What is the usage difference between DISP=SHR and DISP=OLD?

Both are specified when the referred dataset already exists. When DISP=SHR is coded, the dataset can be concurrently used by other jobs whereas DISP=OLD ensures exclusiveness. DISP=SHR is generally used for input and DISP=OLD is for rewrite or append data cases.

When DISP=MOD is used?

1. The data needs to be appended in the bottom of the already existing dataset.
2. The dataset to be created and data to be written when dataset does not exist.
3. DISP=(MOD,DELETE,DELETE) is a very usual parameter that is used in the first IFEBR14 step of every job where all the output datasets created in the JOB are coded with this DISP. This will delete the already existing datasets if any and if none exists, it creates and deletes it.

The disposition in the JCL is MOD and the program opens the file in OUTPUT mode. What happens ? The disposition in the JCL is SHR and the pgm opens the file in EXTEND mode. What happens?

Records will be written to end of file (append) when a WRITE is done in both cases.

What is the usage of DISP=PASS?

PASS passes the data set on to subsequent job steps, and each step can use the data set once. It is a positional sub parameter of the DISP which could only be specified under normal termination action. Pass saves time because the system retains the data set location and volume information for the specific job. DISP=(NEW,PASS,DELETE) is commonly used DISP for all the temporary datasets.

How do you mention Temporary datasets in JCL ? Do the temporary dataset exists after the completion of job ?

Using && prefix. The dataset does not exist after completion of job.

What will happen to the dataset if no DISP is coded?

The default DISP is NEW,DELETE,DELETE. So if the dataset specified already exists, STEP will end with JCLERROR stating dataset already exists. If dataset does not exist, then after step's completion, the dataset is deleted.

What does a disposition of (NEW,CATLG,KEEP) mean?

That this is a new dataset and needs to be allocated, to CATLG the dataset if the step is successful and to KEEP but not CATLG the dataset if the step abends. Thus if the step abends, the dataset would not be catalogued and we would need to supply the vol. ser the next time we refer to it.

How do you access a file that had a disposition of KEEP?

Need to supply volume serial no. VOL=SER=xxxx.

What is NOTCAT 2

This is an MVS message indicating that a duplicate catalog entry exists. E.g., if you already have a dataset with dsn = 'xxxx.yyyy' and u try to create one with disp=new,catlg, you would get this error. The program open and write would go through and at the end of the step the system would try to put it in the system catalog. At this point since an entry already exists the catlg would fail and give this message. You can fix the problem by deleting/uncataloging the first dataset and going to the volume where the new dataset exists (this info is in msglog of job) and cataloging.

When UNCATLG option is used in real time?

When you don't want the dataset to be used without knowing volume serial number, then those dataset are created with UNCATLG as second parameter. But this is no more an option used in real time as all datasets are mandatorily cataloged by SMS (Storage Management System). Even if you code UNCATLG, the dataset will be cataloged in today's world.

What is the difference between KEEP and CATLG?. When KEEP is used?

In earlier days – a dataset can be cataloged or uncataloged. If you would like to leave cataloged dataset in cataloged status and uncataloged status in uncataloged status, KEEP is used. As UNCATLG is no more really different option than CATLG today, there is no difference between CATLG and KEEP as well.

What is the purpose of SPACE parameter?

It is a keyword parameter which should allocated on the DD statement for the output data sets stored on the disk. It consists of the following sub-parameters:
SPACE=(BLKS/CYL/TRK,(primary, secondary, index),RLSE,CONTIG)

RLSE releases all unused space when the data set is closed. Space is released only if the data set is not empty and if the data set is closed after being opened.

CONTIG requests the primary space be allocated only on contiguous tracks and cylinders, that is all tracks on a cylinder are contiguous, and if more than one cylinder is needed, the cylinders are also contiguous. Always code CONTIG if track overflow is used.

What is the significance of UNIT parameter ?

UNIT parameter is used along with a DD statement. It conveys information input or output devices that will be used by the job like TAPE, DASD or temporary DASD units.

What is the use of VOL parameter for dataset?

The volume parameter is used to identify the volume serial number of the device on which a given dataset is to be written or on which a dataset resides.

What is the significance of BUFNO parameter?

The BUFNO parameter is used to specify the number of buffers to be allocated to virtual storage for the dataset. A large number buffers can improve I/O performance.

What is the use of DUMMY parameter ?

The dummy parameter is coded on DD statement to test the execution of a program without actually reading or writing to a dataset.

What are the parameters that have to be coded on a DD statement in order to retrieve a cataloged data set?

The minimum parameters needed are DSN and DISP.

How does the system get information about the block size?

DCB info comes from :-
1) FILE CONTROL and FD Section of the Program
2) The label - like a tape
3) From the VTOC - for Dasd
4) From the JCL - DCB=BLKSIZE=nnn.

What is a Label?

LABEL is a keyword parameter which can be specified on DD statement and consists of the following subparameters:
LABEL=(Relative File #, Type of Label Processing)
The LABEL parameter tells the type of label, the relative file number, and whether the data set is to be protected for input or output.

When should be NL be specified as a type of label processing?

NL should be specified when a program needs to process unlabeled tapes NL can also be specified when the pgm wants to create unlabeled tape because the system's default action, in cases when parameter is not specified, will create IBM standard label. Nonlabeled tapes are often used for sending tapes to another installation. That way you don't have to worry about the tape label corresponding to the standards at the other installation or about accidentally matching the volume serial number of an existing tape at the installation.

How do you describe the data which is a part of the input job stream?

You should use either DD * or DD Data.
//SYSIN DD *
Data
/*

What is the difference between * and Data?

DD * and DD DATA describe the input data which follows the above mention cards. If the input data contains record switch // in col 1 and 2 then DD Data should be used.

What is the significance of OUTLIM parameter?

The OUTLIM parameter is coded on the DD statement and is used to limit the number of lines that are output to the destination specified in the sysout.

What is primary allocation for a dataset?

The space allocated when the dataset is first created.

What is the difference between primary and secondary allocations for a dataset?

Secondary allocation is done when more space is required than what has already been allocated.

How many extents are possible for a sequential file ? For a VSAM file ?

16 extents on a volume for a sequential file and 123 for a VSAM file.

If the primary is 10 tracks and secondary is 20 tracks, then can we say the file can surely get 10+20*15=310 tracks maximum if required?.

Nop. One extent is not one secondary or primary. It means the contiguous memory allocation with respect to primary/secondary. So if many 20 tracks of contiguous memory available in the volume allocated, then only 310 is possible. If only 10 tracks on contiguous memory available, then 2 extents is required to satisfy one secondary itself.

What would be the solution for SB37 abend?

1. Increase the Space allocation as B37 usually means allocated memory is exhausted and program tries to write more.
2. One another reason for B37 is system could not able to get the requested secondary within 5 contiguous memory (extents) collection. If this is the case, then you need to reduce space allocation than increase.

We end up in applying solution 1 only most of the times in real time.

What is the difference between creating a new sequential flat file and PDS with respect to SPACE parameter?

SPACE=(TRK,(n,m)) for a sequential file, SPACE=(TRK,(n,m,p)) for a PDS where n is primary allocation, m is secondary allocation. The p designates how many directory blocks to allocate for a PDS.

What is the need for directory block?

Directory block is used to save the information about the members. In one directory block (that is of 256 bytes), details of 5-6 members can be stored. If the PDS is created with 1 directory block and you try to save the 7th member, you will get SE37 abend stating directory is full.

How much is memory space involved, when we code BLOCKSIZE, TRK & CYL

One block constitutes 32KB of formatted memory/ 42KB of Unformatted memory, 6 blocks makes one Track & 15 Tracks makes one cylinder.

How the datasets are concatenated and the rules associated with that?

Datasets are concatenated by writing a DD statement for the first dataset and then adding a DD statement without a DDNAME for each dataset to be concatenated in the order they are to be read. The following is an example of three datasets concatenated:

```
//EMPFILE    DD      DSN=MVSQUEST.CHN.DATA,DISP=SHR
//           DD      DSN=MVSQUEST.BLR.DATA,DISP=SHR
//           DD      DSN=MVSQUEST.MUM.DATA,DISP=SHR
```

1) Datasets must be of the same type (disk or tape) and format.
2) All datasets must have the same logical record length.
3) Block size can be different but if so, the dataset with the largest block-size must be listed first.

What is STEPLIB, JOBLIB? What is it used for?

Specifies that the private library (or libraries) specified should be searched before the default system libraries in order to locate a program to be executed. The JOBLIB statement is placed after the JOB statement and is effective for all job steps. It cannot be placed in a cataloged procedure. The STEPLIB statement is placed after the EXEC statement and is effective for that job step only. Unlike the JOBLIB statement, the STEPLIB can be placed in a cataloged procedure

What is order of searching of the libraries in a JCL?

First any private libraries as specified in the STEPLIB or JOBLIB, then the installation specific system libraries (like SYS1.LINKLIB) defined in link list.

What happens if both JOBLIB & STEPLIB is specified ?

The search will happen only in STEPLIB. JOBLIB is ignored.

Procedures-Cataloged/Instream

What is meant by Procedure ? What are different types of procedures?

The Procedure (PROC) is used to keep set of frequently used JCL statements in it and invoke from JCL as and when required. This it provides advantages of reusability, integrity and reducing redundancy. There are two types of procedures - Instream procedure and Catalogued procedure.

Explain Instream Procedures

The procedure that is included in the input stream of job is called Instream procedure. They start with PROC statement before the first EXEC statement and end with PEND statement. The visibility of instream procedure is only within the JCL it is coded. Maximum of 15 instream procedures can be coded in a JOB and one procedure can be invoked any number of times in the job.

Explain Cataloged Procedures

Catalogued procedures are written stored in a separate PDS. The source of the PROC PDS is specified in the JCLLIB statement of the JCL where the PROC is invoked. It does not need PEND statement as it is not part of job stream.

Name some of the JCL statements that are not allowed in procs

 Some of the JCL statements which are not allowed in procedures are:

   JOB, Delimiter(/*), or Null statements
   JOBLIB or JOBCAT DD statements
   DD * or DATA statements
   Any JES2 or JES3 control statements

How to code instream data in Procedure?

Instream data is not allowed in procedures as procedures meant for generic purpose. To have instream data, code generic SYSIN DD DUMMY inside the procedure and do dataset override from the JCL that invokes the proc and pass the instream data.

```
//STEP1.SYSIN  DD *
..
/*
```

What are the two options available in PROCs so that it can be written relatively generic?
   1. Using Symbolic parameters
   2. Overrides

Explain the usage of Symbolic Parameters.

The variable portion of the PROC (whether DSN name or Space parameter values or anything else) can be specified with Symbolic parameters and the value for the same can be supplied from the JCL that invokes the PROC so that the PROC can be generic to an extent.

How you code the Symbolic Parameters and how the value for the same can be supplied?.

In the PROC (EMP), they are 1-7 characters word prefixed with & (say &BRANCH). Default value for it can be supplied along with PROC definition following PROC keyword((like //EMP **PROC** BRANCH=TRICHY). The default values can be overridden while invoking the proc from the JCL (like //STEP1 **EXEC** EMP BRANCH=TRICHY) or assign the value with SET statement before invocation of the PROC (like // **SET** BRANCH=TRICHY).

What is the dataset name in DD1 and what is purpose of VAR1=?. How to concatenate symbolic parameters?

```
//EMP PROC VAR1=TRICHY
//S1  EXEC PGM=PGM1
//DD1  DD DSN=&VAR1.&VAR2.&VAR3..&VAR4,DISP=SHR

//T1 EXEC EMP VAR1=,VAR2=MVS,VAR3=QUEST,VAR4=DATA
```

Dataset name is MVSQUEST.DATA and VAR1= is used to nullify the default value defined in the PROC definition. Symbolic parameters can be concatenated with or without delimiter of '.'. If you would like to see '.' In between two symbolic parameter values, you need to give double dots between them(like &VAR3..&VAR4).

How do you restart the job from a step(START) in the proc (BEAUTY) that is invoked in the JOB step CRIME?

In job card, specify RESTART=procstep.stepname. where procstep = name of the JCL step that invoked the procedure and stepname = name of the procedure step from where we need to restart. In this case, RESTART=CRIME.START

A PROC has five steps. Step 3 has a condition code. How can you override/nullify this condition code?

Provide the override on the EXEC stmt in the JCL as follows:
   //STEP001 EXEC procname, COND.stepname = value

All parameters on an EXEC stmt in the proc such as COND, PARM have to be overridden like this.

How do you override a specific DDNAME/SYSIN in PROC from a JCL?

//<stepname.dd> DSN=...

Explain the SET command.

A value of SET by PROC statement is changed, if a SET statement appears within the procedure. SET statement can be placed anywhere following the Job statement. SET is conditional and is not affected by conditional execution of the IF-THEN-ELSE-ENDIF. SET statement can replace setting symbolic parameter values on both EXEC and PROC statement, they also allow to create symbolic values in JCL and assign them without having to create a cataloged or in-stream procedures.

Value assigned to a symbolic variable inside a PROC statement is available for the following JCL statements in the JOB that invoked the PROC?

Yes

Explain Nesting Procedure.

In-stream procedures can be nested to a maximum of 15 levels (i.e. one proc invoking other and so on). We cannot make backward reference between nested procedure. Up to only one over ridding statement is possible. All step-names should be unique So that we can override them correctly.

Explain Include statement.

It is used to copy in JCL stored as a member of a PDS using the JCLLIB statement to name the PDS and placing INCLUDE statements in the JCL, where we want the members to be copied.
// INCLUDE MEMBER=member-name . PDS must be named with JCLLIB statement. PDS must be cataloged and have DCB attributes of LRECL=80 and RECFM=F or FB. It can be place anywhere after JOB statement. It is placed in Cataloged and In-stream procedures. Include group can be nested up to 15 levels. Include cannot Contain JOB, PROC/PEND, JCLLIB, DD *, JES2 & JES3 statements.

While I submit the job, the PROCs are expanded. How do I know that from the PROCs are picked up.

Search in the PDS in the order given in the JCLLIB following JOB card as below.

// JCLLIB ORDER=('SHRUTHI.PROC','PARAM.PROC')

If you don't find them there, then it should have been expanded from system proc libraries. Simplest way to check JES MESSAGE of the job submitted. You could see the information of from where the PROC is expanded.

ERRORS/RC/ABENDs

How do you set a return code 16 from the COBOL program?

RETURN-CODE is a special in COBOL program that can be used to pass the code back to calling program or OS. The below code before STOP RUN sets return code of program executing step to 16.

MOVE 16 TO RETURN-CODE

Is there any utility programs that does return code setup?

Yes. Using IDCAMS SET MAXCC command, you can set the return code using

What are the two types of JCL ERROR?

1. The errors before execution of JOB - due to JCL syntax issues.
2. The errors before execution of STEP – Duplicate dataset issues, dataset unavailable etc.

What are the two types of ABENDs?

1. User Abends – That are issued by the programs for the 'program known' exceptions.
2. System Abends – That are issued by the system as it could not execute the statement.

How a COBOL program issue a user abend say U0999 with a sample?

Assuming CITY in the input file is not found in DB2 Table, the program may not want to proceed further and instead it could abend with the issue in SYSOUT so that production support person can correct the CITY in file or add CITY in table and proceed. Usually installation specific abend routines are called for abending the program. Below is a sample of IBM provided abend routine call.

MOVE 999 TO WS-ABEND-CODE
CALL 'ILBOABN0' USING WS-ABEND-CODE

What are the common system abends?

SB37,SD37,SE37	-	Space Abends
S822,S80A,S804	-	Region Abends
S122,S222	-	Job Cancelled
S322	-	Time Out (CPU Time Limit exceeded)
S522	-	Time Out (I/O wait time exceeded)
SOC7	-	Data Exception (Non numeric data in Numeric Field)
SOC4	-	Protection Exception (Out of array access, Write a file without open etc)
S013	-	Open issue (Usually member specified is not found in PDS)

How to get the dump in case of abnormal termination of the STEP?

Add the DD SYSUDUMP or SYSABEND in the STEP. While SYSUDUMP produces formatted dump of user areas, SYSABEND produces formatted dump of user as well as system areas like LSQA and IOS control blocks for the failed tasks. There is another DD SYSMDUMP can also be coded which produce user as well as system areas in unformatted machine readable format. If more than one DD is coded for dump, the last one coded would be effective.

GDGs
(Example of GDG Base – SHRUTHI.EMP.DATA)

What is a Generation Data Group (GDG)?
Generation Data Group is a group of chronologically or functionally related datasets. GDGs are processed periodically, often by adding a new generation, retaining previous generations, and sometimes discarding the oldest generation.

How is a GDG base created?
A GDG base is created in the system catalog and keeps track of the generation numbers used for datasets in the group. IDCAMS utility is used to define the GDG base.

How GDG dataset looks and How do you refer in JOBS?
It looks like GDGDBASE.GnnnnVxx. Example – SHRUTHU.EMP.DATA.G0001V00
'nnnn' is generation number which will be incremented every time you create a new gen. 'xx' is level number. SHRUTHI.EMP.DATA(0) refers the current generation i.e the generation number that has highest value. SHRUTHI.EMP.DATA(-1) refers previous generation. To create a new generation, code SHRUTHI.EMP.DATA(+1).

How do you refer the generation created in STEP01 in STEP02?
SHRUTHI.EMP.DATA(+1). References are always relative and the generations are updated only at the end of the job and NOT end of the step. Hence creation and reference both should be of (+1).

What are the limitation of GDG?
GDGs can be sequential, direct or Partitioned but cannot be VSAM
GDGs are 'must' cataloged.

What are the main advantages of GDG?
1. System takes care of creating new generations and referring the intended generations using the relative references. No need for manual deletes/changes between restarts.
2. By referring the GDG Base, all datasets of GDG base is automatically concatenated provided all generations on the same volume.

What will happen if the GDG is defined with limit of 10 and your job step tries to create 11th generation?
Depends on whether the option EMPTY coded or not during GDG creation, all the prior 10 generations will be deleted from catalog or only the oldest generation will be dropped from catalog respectively.

Is it possible to get the deleted generation of GDG?
Depends on whether SCRATCH option coded or not during GDG creation, the deleted generation can be recovered using HRECOVER.

What is model dataset label(Model DSCB)?
A model dataset label is a pattern for the dataset label created for any dataset named as a part of the GDG group. The system needs an existing dataset to serve as a model to supply the DCB parameters for the generation data group one wishes to create. The model dataset label must be cataloged. The model DSCB name is placed on the DCB parameter on the DD statement that creates the generation data group.

Which Utility program is used to create GDG base ?

IDCAMS, Define statement is used to create GDG base

Do all versions of the GDG have to be of the same record length ?

No, the DCB of the model dataset can be overridden when you allocate new versions.

How are different versions of GDG named ?

Base-file-name.GnnnnV00 where nnnn= generation number (upto 255). nnnn will be 0000 for the 1st generation.

Suppose 3 generations of a GDG exist. How would you reference the 1 st generation in the JCL?

Use GDG name(-2).

Suppose a generation of GDG gets created in a particular step of a proc. How would you refer the current generation in a subsequent step? What would be the disposition of this generation now?

Relative generation numbers are updated only at the end of the job, not at the end of a step. To allocate a new generation, we would be using (+1) with a DISP of (NEW,CATLG,DELETE). To refer to this in a subsequent step in the same job, we would again use (+1) but with a DISP of SHR or OLD.

Can we browse or edit the GDG dataset if it is a tape entry?

No.

In the below JOB, if SLEEP1 need 4 seconds CPU and SLEEP2 needs 10 seconds CPU, what would happen to the JOB on execution?

SLEEP2 will abend with S322 as the maximum time left out for JOB execution after SLEEP1 is just 6 seconds (Job Limit Time – Already executed STEPs Consumption i.e 10 -4 = 6 in this case)

TIME coded on JOB overrides the TIME coded on STEP in this case.

```
//SUVA123B JOB (XXXX,I000000,1234),SHRUTHI,CLASS=T,
//    MSGCLASS=X,NOTIFY=SHRUTHI,TIME=(,10)
//*
//SLEEP1    EXEC PGM=IKJEFT01,TIME=(,10)
//SYSTSPRT   DD SYSOUT=*
//SYSTSIN    DD *
  EX 'SHRUTHI.DASH5.UTILS(SLEEP)' '9'
/*
//SLEEP2    EXEC PGM=IKJEFT01,TIME=(,10)
//SYSTSPRT   DD SYSOUT=*
//SYSTSIN    DD *
  EX 'SHRUTHI.DASH5.UTILS(SLEEP)' '9'
/*
```

What steps will be executed/flushed in the below job and why?

```
//SUVA123B JOB (XXXX,I000000,1234),SHRUTHI,CLASS=T,
//    MSGCLASS=X,NOTIFY=SHRUTHI,COND=(4,LT)
//*
//STEP01    EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN     DD *
 SET MAXCC=4
/*
//STEP02    EXEC PGM=IEFBR14,COND=((4,LT,STEP01))
//*
//STEP03    EXEC PGM=IEFBR14,COND=((4,LT,STEP01),(0,EQ,STEP02))
//*
//STEP04    EXEC IDCAMS,COND=((4,LT,STEP01))
//SYSPRINT DD SYSOUT=*
//SYSIN     DD *
 SET MAXCC=8
/*
//STEP05    EXEC PGM=IEFBR14,COND=EVEN
```

1. STEP01 will be executed being the first step.
2. STEP02 will be executed as the COND coded in STEP as well as JOB is FALSE.
3. STEP03 will be bypassed as one of the conditions coded in the STEP is true (0,EQ,STEP02).
4. STEP04 will be executed as the COND coded in STEP as well as JOB is FALSE. After the execution of this, JOB condition becomes TRUE and hence any step coded below will be ignored.
5. STEP05 will be bypassed as JOB has a TRUE condition. Though EVEN coded, job is already stopped as JOB COND satisfied at end of STEP04.

Assuming there is no load module in the name MUTHU exists, what are steps will be executed/flushed in the below job and why?

```
//SUVA123B JOB (XXXX,I000000,1234),SHRUTHI,CLASS=T,
//    MSGCLASS=X,NOTIFY=&SYSUID,COND=(4,LT),RESTART=STEP02
//*
//STEP01    EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
 SET MAXCC=3
/*
//CHK IF STEP01.RUN=FALSE THEN
//STEP02    EXEC PGM=IEFBR14
// ENDIF
//CHK IF STEP01.RC = 3 THEN
//STEP03    EXEC PGM=IEFBR14
// ENDIF
//STEP04    EXEC PGM=IEFBR14,COND=(3,EQ,STEP01)
//*
//STEP05    EXEC PGM=MUTHU
//CHK IF STEP05.ABENDCC=S0C7 THEN
//STEP06    EXEC PGM=IEFBR14
```

1. STEP01 will be flushed as this is RESTART.
2. STEP02 will be executed as the STEP01 did not run in this execution (STEP01.RUN=FALSE is true). If this is coded as STEP01.RUN=TRUE or any other checks like STEP01.RC = 0 etc then STEP02 could have been skipped.
3. STEP03 will be skipped as STEP01.RC check cannot be done as the STEP01 didn't execute.
4. STEP04 will execute as any conditions coded in COND against the STEPS didn't execute will be considered as dummy checks.
5. STEP05 will abend with S806 (Load Module Not Found)
6. STEP06 will be flushed as the abend is not S0C7.

Spool Output

```
22.18.35 JOB18041  IEF403I SUVA123B - STARTED - TIME=22.18.35
22.18.35 JOB18041  NDC101I SUVA123B STEP02              00
22.18.35 JOB18041  NDC101I SUVA123B STEP03           FLUSH
22.18.35 JOB18041  NDC101I SUVA123B STEP04              00
 -  -  -  -  -  -  -  -  -  -  -  -  -  - 20 Line(s) not Dis
22.18.35 JOB18041  NDC101I SUVA123B STEP05            *S806
22.18.35 JOB18041  NDC101I SUVA123B STEP06           FLUSH
22.18.35 JOB18041  IEF404I SUVA123B - ENDED - TIME=22.18.35
```

The ID is disconnected while in use. What would you do?

If installation allows, you could ask your friend to kill your user id job from the spool. If access is not there, then ask your friend submit the below kind of job that will purge your ID by yourself. The other option is 'just' wait for the ID to be released by system itself after installation defined idle time.

```
//SHRUTHIT JOB (TEST,T),T,MSGCLASS=X,CLASS=P,COND=(0,NE),
//         NOTIFY=SHRUTHI,USER=SHRUTHI,PASSWORD=SUVA123$
//*********************************************************************
//* BY MVSQUEST – TO PURGE THE ID
//*********************************************************************
//*
//BATCH   EXEC PGM=SDSF
//ISFOUT   DD SYSOUT=*
//ISFIN    DD *
OWNER SHRUTHI
ST SHRUTHI
FIND 'SHRUTHI'
++P
ST
/*
```

The above job is batch mode of accessing SDSF. USER= and PASSWORD= is the userid and password of the ID to be killed.

What is the return code of the below JCL STEP if the connection to mvsquest.ftp is timed out?

```
//BEFORE    EXEC PGM=FTP
//*
//SYSUDUMP  DD  SYSOUT=*
//OUTPUT    DD  SYSOUT=*
//SYSPRINT  DD  SYSOUT=*
//*
//INPUT     DD  *
mvsquest.ftp
shruthi
suva123$
cd /opt/pgp/data/nkluser/korea/Inbound/
BINRRY
GET emp.zip 'PROD.EMP.CHENNAI'
/*
//
```

The RC will be zero only. To get failures, you need to code PARM in EXEC statement as below. With the below PARM, any failures in FTP results return code of 12 for the STEP. You could change 12 to any number in the PARM.

```
//BEFORE    EXEC PGM=FTP,PARM='(EXIT=12'
```

What is the DSN name of DD01 and DD02?

JCL:

```
//SUVA123B JOB (XXXX,I000000,1234),SHRUTHI,CLASS=T,
//     MSGCLASS=X,NOTIFY=&SYSUID,TYPRUN=SCAN
//*
//PROCLIB JCLLIB ORDER=('KHK029U.DASH5.UTILS')
//*   BY MVSQUEST FOR SET SCOPE UNDERSTANDING
//  SET PREFIX=SHRUTHI
//STEP01    EXEC PROC1 PREFIX=SHRUTHI
//*
//STEP02    EXEC PGM=IEFBR14
//DD02    DD DSN=&PREFIX..EMP.CHENNAI,
//           DISP=(OLD,DELETE,DELETE),
//           SPACE=(TRK,(1,10),RLSE)
```

PROC:

```
//PROC1 PROC PREFIX=PROC
//PROC01 EXEC PGM=IEFBR14
//DD01      DD DSN=&PREFIX..EMP.CHENNAI,
//             DISP=(OLD,DELETE,DELETE),
//             SPACE=(TRK,(1,10),RLSE)
// SET PREFIX=MVSQUEST
```

1. DD01 is PROC.EMP.CHENNAI. PREFIX has default value of PROC inside the PROC. The value of PREFIX is not overridden while invoking the PROC in STEP01. SET Values will be active only if there is no default values inside the PROC.

2. DD02 is SHRUTHI.EMP.CHENNAI. Any PREFIX (symbolic parameter) value defined in the PROC has its scope only within the PROC and its nested PROCs.

Disclaimer:

Though this document is prepared by MVSQuest Team (mvsquest@yahoo.com), many FAQ documents available in net and shared by our friends are referred to prepare this. This FAQ does not cover IBM Utilities related FAQs including SORT. We will be adding them as well as more code related FAQs in version 2.0