Job Control Language (JCL)

Lesson 2: JCL Syntax

Lesson Objectives

- Syntax for writing JCL
- Rules to be followed

2.1: Syntax of JCL
Description

- JCL statements are coded in 80-byte records.

- Only 72 characters are used to code JCL statements.

- Within 72 characters, JCL statements can be coded in a relatively free-form manner, with just few restrictions.

- Each JCL statement is divided into many fields.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

All JCL statements begin with two slashes (//) in the first two positions (except /*). All positions of a line, from 1 to 71 (included), can be used for coding a JCL statement. Position 72 is used (rarely) for imbedded comment continuation, and positions 73 through 80 are used for numbering purposes.

With the exception of the null (//), delimiter (/*), comment (//*), all other statements follow the same general format:

```
//name     operation parameter1,parameter2    [comment]
```

2.1: Syntax of JCL
Description (contd..)

- Different types of JCL statements
  - JOB
  - EXEC
  - DD
  - DELIMITER
  - Null
  - COMMENTS
  - PROC
  - PEND
  - OUTPUT

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

JOB- Marks the beginning of a job and assign a name to the job.
EXEC- Marks the beginning of a job and names the job step. Identifies the program or procedure to be executed
DD- Identifies an input or output file within a job step and defines all the resources required
DELIMITER(/*)- Indicates the end of data placed in a JCL
NULL(//)-Marks the end of a job
COMMENTS(//*) - Contains comments
PROC - Marks the beginning of a procedure definition
PEND - Indicates the end of a procedure definition
OUTPUT - Supplies options for SYSOUT processing

2.1: Identifier Field
Format

//NAME   Operation   param1,param2 …

- Identifier Field
  - Identifies a record as a JCL statement.
  - For most JCL statements, this field occupies the first two character positions and contains two slashes (//).
  - Two exceptions are:
  - Delimiter statement that contains a slash and the start(/*)
  - Comment statement with identifier field as (//*).

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

Name: Every JCL statement can or must have a name. The name should not exceed 8 characters.

Operation: The operation field follows the name field and specifies the statement's function. Delimiter, comment and null statements do not have an operation field. E.g. JOB, EXEC, DD, PROC, PEND, or OUTPUT. One or more blanks must follow the operation.

Parameter: The parameter field consists of one or more parameters, separated by commas. No imbedded blanks between parameters are permitted Parameters are broadly classified into 2 categories viz. Positional and Keyword.

A positional parameter is identified by its position relative to other parameters in the operand field.

Note: THE COMMENT STATEMENT CAN NOT BE CONTINUED!

2.1: Name Field
Format

- Name Field
  - Associates a name with a JCL statement.
  - Always required on a JOB statement.
  - Must start from column three.
  - Can be up to 8 characters in length.
  - Comprises letters, numbers or national characters (#, @ and $).
  - First character must be a letter or national character.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

2.1: Operation Field

Format

- Operation Field
  - Specifies the statement's function.
  - Can be coded anywhere on the line, but should be separated from the name field by at least one space.
  - Statements that do not have operation field are:
    - /*　------　　　　　　　Delimiter
    - //　------　　　　　　　Null
    - //*　------　　　　　　Comment

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

2.1: Parameter Field
Description

- In JCL parameters are of two types:
  - Positional
    - must be first
    - must be in a specific order
    - absence must be indicated by a comma
    - if the last one is absent, no comma necessary
    - if all are absent, no comma is necessary.
    - E.g..
      - TIME=(5,0) = TIME=(5) = TIME=5
      - TIME=(0,30) = TIME=(,30)
  - Keyword
    - must be after any positional parameters
    - can be any order with respect to one another
    - Identified by an equal sign (=) and variable information
    - DISP=OLD,DISP=(NEW,KEEP,KEEP),DISP=(,KEEP,KEEP)
    - DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

2.1: Parameter Field
Format

- Parameter Field
  - Begins at least one position after the end of operation field and can be extended up to column 71.
  - One or more component parameters are coded, each separated by commas.
    - Parameters supply information that influence statement processing.
  - Comments field begins in the position after the space that marks the end of parameters field.

Rule 1: All positional parameters are coded first in the operand field and in their proper sequence.
E.g.: p1,p2,p3
     p1,p3,p2   illegal because they are not in sequence.

Rule 2: A keyword parameter is identified by a keyword followed by an equal sign (=) and variable information. A keyword parameter follows positional parameter and can be coded in any order.
E.g. : p1,p2,p3,k1=,k2=,k3=
     p1,p2,p2,k3=,k2=,k1=
Both are valid.

Rule 3: The absence of positional parameter is denoted by a comma (,) coded in its place, except when the last or remainder of the positional parameter is not present. The placeholder commas do not need to be coded in this case.
E.g.: p1,,p3,k1=,k2=,k3=   place holder comma required.
    p1,k1=,k2=,k3=   place holder comma not required.

2.1: Syntax of JCL
Example

- Syntax

```
//OUTPUT DD DSN=DA0001T.TRANS,DISP=(,CATLG),
//      SPACE=(TRK,(2,1)),UNIT=SYSDA
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

Rule 4: Both positional parameters and variable information for keyword parameters may be composed of sub parameters. The sub-parameters may be either positional or keyword. Sub-parameters must be coded as a list. The list must be enclosed in parentheses unless only one sub- parameter is coded. When only one sub-parameter is coded the parentheses are optional.

e.g.: p1,(sp1,sp2),p3,k1=(sk1,sk2),k2=,k3=
p1,(sp1),p3,k1=(sk1),k2=,k3=
Or p1,sp1,p3,k1=sk1,k2=,k3=

Comments: are separated from the parameters by a blank. The comment field begins in the position after the space that marks the end of parameters field and ends in column 71. MVS ignores what you code here.

2.2: JCL Rules

Rules

- Positional parameters are coded first in the operand field and in their proper sequence.
  - Absence is denoted by a comma (,) coded in its place.

- A keyword parameter is identified by a keyword followed by an equal sign (=) and variable information.

- Both positional parameters and variable information for keyword parameters may comprise sub-parameters.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

2.2: JCL Rules
Rules for Continuation

- JCL statement can be continued into the next line.
  - Code two slashes at the beginning of the line.
  - Continue in the parameter field.
  - Start anywhere between positions 4 and 16 (4 and 16 included).
- Note:
  - A comma enclosed in an apostrophe cannot be used for continuation.

Rules for Continuation: The JCL statement can be continued in a simple way. The statement must be interrupted at a comma. This means that the last valid character of the line must be a comma followed by at least one blank. Then the statement can be continued into the next line by coding two slashes at the beginning of the line and continuing the parameter field starting anywhere between positions 4 and 16 (4 and 16 included). Note that, the comma that indicates continuation, is not an extraneous character but part of the statement.

2.2: JCL Rules
Rules for Continuation: Example

```
//DD1    DD  DSN=DA0001T.EMPFILE,
//    DISP=(NEW,CATALG,DELETE),
//     UNIT=SYSDA,SPACE=(TRK,(5,1)),
//     DCB=(LRECL=80,RECFM=FB,BLKSIZE=800)
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

Summary

- Every JCL statement must have a name.

- A keyword parameter is identified by a keyword followed by an equal sign (=) and variable information.

- Absence of positional parameter is denoted by a comma (,).

Summary

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

Review Question

- Question 1: All JCL statements begin with:
  - Option 1: //
  - Option 2: **
  - Option 3: --
- Question 2: Which of the parameters are coded first in a JCL statement?
  - Option 1: Positional
  - Option 2: Keyword