

**[Exercise 1]**

1. 下面的 JCL 语句是否正确? 如果有错误的话,请指出错误的原因.

- a. // JOB MSGCLASS=J
- b. //DD1 DSN=DATA DD
- c. //TEST\*1 JOB MSGLEVEL=(1,1)
- d. //STEPSEVEN EXEC PGM=ACCPAY
- e. //STEP1 EXEC PGM=ACCREC
- f. //DD3 DDDSN=DATA
- g. /OUTPUT EXEC PGM=CHECKS
- h. //DDS DD DSN=DATA,UNIT=234
- i. // JOB1 JOB MSGLEVEL=(1,1)

2. 根据给出的信息,写出相应的 JOB STATEMENTS

a)

JOB 名 -> **ABC1**  
会计信息 -> **D415-3**  
Programmer -> **YAMADA**  
印刷全部的 Message  
印刷全部的 JCL 语句  
MESSAGE CLASS -> **A**

b)

JOB 名 -> **ABC2**  
会计信息 -> **459B-7F**  
Programmer -> **YAMADA**  
不印刷 Message  
只印刷 JOB STATEMENTS

c)

JOB 名 -> **ABC3**  
Programmer -> **YAMADA**  
不印刷 Message  
只印刷当前 JCL 内的 STATEMENTS  
MESSAGE CLASS -> **Q**

**[Exercise 2]****1. 根据给出的信息,写出 EXEC 语句**

a.

STEP NAME: STEPA

PROGRAM : HOKOKU

PARM : 10-25-92

b.

STEP NAME: STEPB

PROGRAM : KYUYO

PARM : TSUKI

**2. 从右边选出左边的 DATASET 的种类**

	1) &&JIKAN	a. 简单的 DATASET 名
	2) JIKAN	b. 带修饰的 DATASET 名
	3) JIKAN(YAMADA)	c. 临时的 DATASET 名
	4) 省略 DATASET 名	d. PDS 的 DATASET 名
	5) JIKAN.KEIRI.KAIKEI	e. 无效的 DATASET 名
	6) SHUKKINBO	

**3. 写出要在 DASD 上做成下面要求的名为 SHUKKIN 的 DATASET 的语句**

- DD 名为 KYUYODD
- 程序异常结束时不保存 DATASET, JOB 结束时保存 DATASET
- DATASET 建立在 SYSDA 的装置上
- 使用的 VOLUME 名为 KYUYO
- 最少需要 20TRACK 的空间,必要时每次增加 1TRACK

**[Exercise 3]**

请参照给定的参照说明完成下面的 JCL 里划线的部分.

1. //COMPILE EXEC PGM=PLI
2. //COMPOUT DD UNIT=SYSDA,VOL=SER=DISK12,DISP=(NEW,PASS),  
// SPACE=(80,240)
3. //LKED EXEC PGM=LINKEDIT
4. //LKEDIN DD DISP=OLD,DSN=\_\_\_\_\_
5. //SYSLMOD DD DISP=(NEW,PASS),SPACE=(1024,(200,20,1)),  
// DSN=&&GOSET(GO),VOL=\_\_\_\_\_
6. //GO EXEC PGM=\_\_\_\_\_
7. //MYDATA DD DSN=MYDATA,DISP=(NEW,CATLG),  
// VOL=SER=USER10,  
// SPACE=(800,50),DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
8. //TEMP DD UNIT=SYSDA,SPACE=(800,200),DCB=\_\_\_\_\_

Reference:

Line NO.	Parameter Type	Reference Line
4	Data set name	2
5	Volume information	2
6	Program name	5
8	DCB attribute	7

**[Exercise 4]****Using of Utilities****LAB1:****STEP1:**

Create a JCL names as JCLLAB1 in your own PDS – SX01I.EV\*\*\*\*.JCL (should be created in PLI exercise)

JCL step1, using utility **IDCAMS** to delete the data set SX01I.EV\*\*\*\*.JCLLAB1.QSAM

JCL step2, Using utility **IEBGENER** to create a data set.

create and catalog a QSAM data set names **SX01I.EV\*\*\*\*.JCLLAB1.QSAM**, allow for 5 tracks primary and 2 tracks secondary on a SYSDA volume, record format using FB, and length is 80, using the default block size, and send some data to the output data set using in-stream input.

**STEP2:** Add the MSGLEVEL parameter to JOB statement and changes the value of it then submit it to check the different from **SDSF**

**MSGLEVEL=(0,0) -> ?**

**MSGLEVEL=(0,1) -> ?**

**MSGLEVEL=(1,0) -> ?**

**MSGLEVEL=(1,1) -> ?**

**MSGLEVEL=(2,0) -> ?**

**MSGLEVEL=(2,1) -> ?**

**STEP3:** Using MSGLEVEL= (1, 1), Change the DISP parameter of the DD statement and check the output from 3.4

DISP= (, CATLG, DELETE) ->

DISP= (NEW, PASS) ->

DISP= (MOD, CATLG) ->

DISP= (OLD, CATLG) ->

---

**LAB2:**

Create a JCL names as JCLLAB2 in your own PDS – SX01I.EV\*\*\*\*.JCL (should be created in PLI exercise)

JCL step1, invoke the utility **IDCAMS** to delete the data set:

SX01I.EV\*\*\*\*.JCLLAB2.PDS

SX01I.EV\*\*\*\*.JCLLAB2.STEP4

SX01I.EV\*\*\*\*.JCLLAB2.STEP5

SX01I.EV\*\*\*\*.JCLLAB2.STEP6

JCL step2, invoke the utility **IEBGENER** to copy the data set you created in LAB1 to a member named LAB2 in SX01I.EV\*\*\*\*.JCLLAB2.PDS. The data set attributes is record length 80, block size 3200, directory 10.

JCL step3, invoke the utility **IEBGENER** to copy member JCLLAB21 and JCLLAB22 from SX01I.JCLEDU.JCLM to SX01I.EV\*\*\*\*.JCLLAB2.PDS, and copy member JCLLAB2B from SX01I.JCLEDU.JCLM to SX01I.EV\*\*\*\*.JCLLAB2.PDS and rename it to JCLLAB3.

JCL step4, invoke the utility **IEHLIST** to list the entries of SX01I.EV\*\*\*\*.JCLLAB2.PDS, and the output is SX01I.EV\*\*\*\*.JCLLAB2.STEP4, using record length 133.

JCL step5, invoke the utility **ICEMAN** to sort SX01I.EV\*\*\*\*.JCLLAB2.PDS (JCLLAB21) by the first 5 bytes Ascending and the next 5 bytes descending, the sorted output is SX01I.EV\*\*\*\*.JCLLAB2.STEP5, using the same record length of input data set.

JCL step6, invoke the utility **ICEMAN** to merge SX01I.EV\*\*\*\*.JCLLAB2.PDS (JCLLAB22) and SX01I.EV\*\*\*\*.JCLLAB2.STEP5 to SX01I.EV\*\*\*\*.JCLLAB2.STEP6, using the same record length of input data set.

---

### **LAB3 (Optional)**

If you have time, please try the other utilities in the class materials.

**[Exercise 5]****Modifying of Procedure**

1. Create a JCL named as JCLLAB51 in SX01I.EV\*\*\*\*.JCL, using the external procedure SX01I.JCLEDU.JCLPROC (IBMZCPL) to compile your program SX01I.EV\*\*\*\*.PLI (SAMPLE1). Your load module must kept on a new data set SX01I.EV\*\*\*\*.LMODNEW, the new data set should be cataloged, and the record format is U, the directory is 2, the UNIT is SYSDA.

2. Create a JCL named as JCLLAB52, the requirement is following:

- Create an in-stream procedure using **IEBGENER**.

```
//TESTPOC PROC
//IEBGENER EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DUMMY
//SYSUT2 DD SYSOUT=*
//SYSIN DD DUMMY
// PEND
```

- Add a step using the proc you created to copy an in-stream data to a new data set – SX01I.EV\*\*\*\*.JCLLAB52.OUT1, the record length is 80, the record format is FB,

```
JONES   FRED   53AF   87   5701 NINE MILE ROAD
ANDERSONDON    78AF   34   320 WESTHERIMAR, #219
```

- Add a step using the proc you created to copy SX01I.EV\*\*\*\*.JCLLAB52.OUT1 to SX01I.EV\*\*\*\*.JCLLAB2.PDS (JCLLAB52), the data set SX01I.EV\*\*\*\*.JCLLAB2.PDS should be created when you finished the previous exercise.
- Add step using the proc you created to printing selected portions of data set - SX01I.EV\*\*\*\*.JCLLAB52.OUT1, the SYSIN is following:

```
GENERATE MAXFLDS=3
RECORD FIELD=(10,20,,1),FIELD=(10,1,,15),FIELD=(6,5,,30)
```

**[Exercise 6]****Controlling job flow according to Condition**

- Copy the JCL SX01I.EV\*\*\*\*.JCL (JCLLAB2) to SX01I.EV\*\*\*\*.JCL (JCLLAB61)

Add the following condition to the JOB using COND parameter.

1. Step3 only execute when Step2 is success
2. Whatever the result of the previous step, execute Step4
3. If the return code of Step3 is less than 8, execute Step5
4. Only if the return code of Step3 and Step5 all equal to 0, execute Step6

After modifying the JOB, execute it. Then under SDSF to check the result

Comment the Step1 out then resubmit the JOB and check the result

- Copy the JCL SX01I.EV\*\*\*\*.JCL (JCLLAB2) to SX01I.EV\*\*\*\*.JCL (JCLLAB62)

Add the following condition to the JOB using IF/THEN/ELSE.

1. Step3 only execute when Step2 is success
2. Whatever the result of the previous step, execute Step4
3. If the return code of Step3 is less than 8, execute Step5
4. Only if the return code of Step3 and Step5 all equal to 0, execute Step6

After modifying the JOB, execute it. Then under SDSF to check the result

Comment the Step1 out then resubmit the JOB and check the result