

THE COMPLETE JOB CONTROL LANGUAGE COURSE

Programs used



www.MainframeNation.com

List of programs

1. Create a PS dataset
2. Create a PDS dataset
3. Create a PDS dataset member
4. Delete a PS dataset
5. Delete a PDS dataset
6. Delete a PDS dataset member
7. Create a GDG
8. Create a GDG Generation
9. Create a KSDS
10. Create an ESDS
11. Create a RRDS
12. Create a LDS
13. LISTCAT a VSAM
14. REPRO - KSDS to PS
15. REPRO - PS to KSDS
16. REPRO - PS to ESDS
17. REPRO - PS to RRDS
18. REPRO - RRDS to RRDS
19. REPRO - ESDS to ESDS
20. Compare PS datasets
21. Compare PDS datasets
22. Copy one PDS to another
23. Copy one PDS to another using SYSUT
24. Merge Datasets
25. SELECTive Copy
26. Rename member while copying
27. Replace member while copying
28. Compress a PDS
29. Exclude members while copying
30. Rename and Replace while copying
31. Copy whole job into another dataset
32. Copy multiple job into another dataset
33. Copy steps using INCLUDE and EXCLUDE
34. Copy steps using POSITION member



35. List a PDS
36. List a VTOC
37. SCRATCH, UNCATALOG a dataset
38. UNCATALOG a dataset
39. SORT a field
40. SORT multiple fields
41. SORT multiple fields using FORMAT keyword
42. Copy a dataset using SORT
43. SORT with INCLUDE
44. SORT INCLUDE with multiple conditions
45. SORT with OMIT condition
46. SORT with OUTREC
47. OUTREC using FINDREP
48. OUTREC only specific columns
49. Split file based on a condition
50. Multiple copies using OUTFIL
51. Split a dataset into equal parts



Programs used

1. Create a PS dataset

```
//CREATEPS JOB CLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),MSGCLASS=A 00010000
//***** 00011000
//***** JCL TO CREATE A PS DATASET 00012000
//***** 00013000
//STEP010 EXEC PGM=IEFBR14 00020000
//DD1 DD DSN=ABHI.COURSE.PS, 00030000
// DISP=(MOD,CATLG,DELETE), 00040000
// SPACE=(TRK,(10,100),RLSE), 00050000
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=800,DSORG=PS) 00060000
//DD2 DD DSN=ABHI.COURSE.PS2, 00061000
// DISP=(MOD,CATLG,DELETE), 00062000
// SPACE=(TRK,(10,100),RLSE), 00063000
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=800,DSORG=PS) 00064000
/* 00070000
```

2. Create a PDS dataset

```
//CREATPDS JOB CLASS=A,PRTY=15,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID 00010000
//***** 00020000
//***** JCL TO CREATE A PDS DATASET 00030000
//***** 00040000
//STEP010 EXEC PGM=IEFBR14 00050000
//DD1 DD DSN=ABHI.COURSE.PDS, 00060000
// DISP=(NEW,CATLG,DELETE), 00070000
// SPACE=(CYL,(10,100,4),RLSE), 00080000
// DCB=(LRECL=80,BLKSIZE=800,RECFM=FB,DSORG=PO) 00090000
// 00100001
//STEP020 EXEC PGM=IEFBR14 00110001
//DD1 DD DSN=ABHI.COURSE.PDS2, 00120001
// DISP=(NEW,CATLG,DELETE), 00130001
// SPACE=(CYL,(10,100,4),RLSE), 00140001
// DCB=(LRECL=80,BLKSIZE=800,RECFM=FB,DSORG=PO) 00150001
/* 00160001
```

3. Create a PDS dataset member

```
//CREATMEM JOB CLASS=A,PRTY=15,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID 00020001
//***** 00030001
//***** JCL TO CREATE A PDS DATASET MEMBER 00040001
//***** 00050001
//STEP010 EXEC PGM=IEBGENER 00060001
//SYSUT1 DD * 00070001
//SYSUT2 DD DSN=ABHI.COURSE.PDS(MEMBER1), 00070101
// DISP=SHR 00080001
//SYSIN DD DUMMY 00090001
//SYSPRINT DD SYSOUT=* 00100001
/* 00110001
```



4. Delete a PS dataset

```
//DELPS JOB CLASS=A,PRTY=15,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID 00020001
//***** 00030001
//***** JCL TO DELETE A PS DATASET 00040001
//***** 00050001
//STEP010 EXEC PGM=IEFBR14 00060001
//DD1 DD DSN=ABHI.COURSE.PS, 00070001
// DISP=(OLD,DELETE,DELETE) 00080001
/* 00090001
```

5. Delete a PDS dataset

```
//DELPS JOB CLASS=A,PRTY=15,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID 00020000
//***** 00030000
//***** JCL TO DELETE A PDS DATASET 00040001
//***** 00050000
//STEP010 EXEC PGM=IEFBR14 00060000
//DD1 DD DSN=ABHI.COURSE.PDS, 00070001
// DISP=(OLD,DELETE,DELETE) 00080000
/* 00090000
```

6. Delete a PDS dataset member

```
//DELMEM JOB CLASS=A,PRTY=15,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID 00020001
//***** 00030000
//***** JCL TO DELETE A PDS DATASET MEMBER 00040001
//***** 00050000
//STEP010 EXEC PGM=IDCAMS 00060001
//DD1 DD DSN=ABHI.COURSE.PDS, 00070000
// DISP=SHR 00080001
//SYSPRINT DD SYSOUT=* 00081001
//SYSIN DD * 00082001
DELETE 'ABHI.COURSE.PDS(MEMBER1)' FILE(DD1) 00083001
/* 00090000
```

7. Create a GDG

```
//CREGDG JOB CLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),MSGCLASS=A, 00020000
// PRTY=15 00030000
//***** 00050000
//***** JCL TO CREATE A GDG 00060000
//***** 00070000
//STEP010 EXEC PGM=IDCAMS 00110000
//SYSPRINT DD SYSOUT=* 00120000
//SYSIN DD * 00130000
DEFINE GDG(NAME(BANK.CUSTOMER.MONTHLY) - 00140000
LIMIT(12) - 00150000
NOEMPTY - 00160000
SCRATCH 00170000
/* 00180000
```



8. Create a GDG Generation

```
//CREGEN JOB CLASS=A, NOTIFY=&SYSUID, MSGLEVEL=(1,1), MSGCLASS=A, 00010000
//          PRTY=15 00020000
//***** 00030000
//***** JCL TO CREATE A GDG GENERATION 00040000
//***** 00050000
//STEP010 EXEC PGM=IEFBR14 00060000
//DD1 DD DSN=ABHI.COURSE.GDG(+1), 00070003
//          DISP=(NEW,CATLG,DELETE), 00071000
//          SPACE=(TRK,(100,100),RLSE), 00072000
//          DCB=(LRECL=80,RECFM=FB,BLKSIZE=800,DSORG=PS) 00073000
//SYSPRINT DD SYSOUT=* 00080000
/* 00130000
```

9. Create a KSDS

```
//CREKSDS JOB CLASS=A, MSGLEVEL=(1,1), MSGCLASS=A, NOTIFY=&SYSUID, 00010007
//          PRTY=15 00020007
//* 00040007
//* >>>> CREATE A KSDS <<<< 00050007
//* 00060007
//STEP010 EXEC PGM=IDCAMS 00070007
//SYSPRINT DD SYSOUT=* 00080007
//SYSIN DD * 00090007
        DEFINE CLUSTER ( - 00100007
                NAME (ABHI.COURSE.KSDS5) - 00110012
                INDEXED - 00120007
                KEYS (5 0) - 00130007
                RECSZ (200 200) - 00140009
                FREESPACE (10 20) - 00150007
                TRACKS (50 30) - 00160007
                CISZ (4096) - 00170007
                VOLUME (ZASYS1) - 00180007
/* 00190007
```

10. Create an ESDS

```
//CREESDS JOB CLASS=A, MSGLEVEL=(1,1), MSGCLASS=A, NOTIFY=&SYSUID, 00010003
//          PRTY=15 00020003
//* 00030003
//* >>>> CREATE AN ESDS <<<< 00040003
//* 00050003
//STEP010 EXEC PGM=IDCAMS 00060003
//SYSPRINT DD SYSOUT=* 00070003
//SYSIN DD * 00080003
        DEFINE CLUSTER ( - 00090003
                NAME (ABHI.COURSE.ESDS) - 00100007
                NONINDEXED - 00110003
                RECSZ (200 200) - 00120005
                FREESPACE (10 10) - 00130003
                CISZ (8192) - 00140003
                TRACKS (20 20) - 00150003
                VOLUME (ZASYS1) - 00160003
/* 00170003
```



11. Create a RRDS

```
//CRERRDS JOB CLASS=A,MSGLEVEL=(1,1),MSGCLASS=A,NOTIFY=&SYSUID, 00010006
//          PRTY=15 00020006
//* 00030006
//* >>>> CREATE A RRDS <<<< 00040006
//* 00050006
//STEP010 EXEC PGM=IDCAMS 00060006
//SYSPRINT DD SYSOUT=* 00070006
//SYSIN DD * 00080006
        DEFINE CLUSTER ( - 00090006
                NAME (ABHI.COURSE.RRDS) - 00100011
                NUMBERED - 00110006
                RECSZ(200 200) - 00120007
                CISZ(4096) - 00130007
                TRACKS(20 20) - 00140006
                VOLUME(ZASYS1) - 00150006
                FREESPACE(10 10)) 00160006
/* 00170006
```

12. Create a LDS

```
//CRELDS JOB CLASS=A,MSGLEVEL=(1,1),MSGCLASS=A,NOTIFY=&SYSUID, 00010005
//          PRTY=15 00020005
//* 00030005
//* >>>> CREATE A LDS <<<< 00040005
//* 00050005
//STEP010 EXEC PGM=IDCAMS 00060005
//SYSPRINT DD SYSOUT=* 00070005
//SYSIN DD * 00080005
        DEFINE CLUSTER ( - 00090005
                NAME (ABHI.COURSE.LDS) - 00100006
                LINEAR - 00110005
                CISZ(4096) - 00120005
                TRACKS(20 20) - 00130005
                VOLUME(ZASYS1) 00140005
                ) 00150005
/*
```

13. LISTCAT a VSAM

```
//LISTCAT JOB CLASS=A,MSGLEVEL=(1,1),MSGCLASS=A,NOTIFY=&SYSUID, 00010000
//          PRTY=15 00020000
//* 00030000
//* >>> LISTCAT A VSAM <<< 00040000
//* 00050000
//STEP010 EXEC PGM=IDCAMS 00060000
//SYSPRINT DD SYSOUT=* 00070000
//SYSIN DD * 00080000
        LISTCAT ENTRY (ABHI.COURSE.RRDS) - 00090000
                ALL 00100000
/* 00110000
```



14. REPRO - KSDS to PS

```
//REPKSKS JOB CLASS=A, NOTIFY=&SYSUID, MSGLEVEL=(1,1), MSGCLASS=A, 00010000
//          PRTY=15 00020000
//* 00030003
//* >>>> JCL TO COPY FROM KSDS TO PS <<<< 00040003
//* 00050000
//STEP010 EXEC PGM=IDCAMS 00060000
//SYSPRINT DD SYSOUT=* 00070000
//* 00080000
//KSDSFILE DD DISP=SHR, DSN=ABHI.COURSE.KSDS 00090003
//PSFILE DD DISP=SHR, DSN=ABHI.COURSE.PS 00100003
//* 00110000
//SYSIN DD * 00120000
        REPRO INFILE(KSDSFILE) - 00130000
            OUTFILE(PSFILE) 00140003
/* 00150000
```

15. REPRO - PS to KSDS

```
//REPPSKS JOB CLASS=A, MSGLEVEL=(1,1), MSGCLASS=A, NOTIFY=&SYSUID, 00010007
//          PRTY=15 00020007
//* 00030007
//* >>>> REPRO FROM PS TO KSDS <<<< 00040007
//* 00050007
//STEP010 EXEC PGM=IDCAMS 00060007
//SYSPRINT DD SYSOUT=* 00070007
//SYSOUT DD SYSOUT=* 00080007
//* 00090007
//PSFILE DD DISP=SHR, DSN=ABHI.COURSE.PS 00100007
//KSDSFILE DD DISP=SHR, DSN=ABHI.COURSE.KSDS 00110007
//* 00120007
//SYSIN DD * 00130007
        REPRO INFILE(PSFILE) - 00140007
            OUTFILE(KSDSFILE) 00150007
/* 00160007
```

16. REPRO - PS to ESDS

```
//REPPSES JOB CLASS=A, NOTIFY=&SYSUID, MSGLEVEL=(1,1), MSGCLASS=A, 00020001
//          PRTY=15 00030000
//* 00050001
//* >>>> JCL TO COPY FROM PS TO ESDS <<<< 00060001
//* 00070001
//STEP010 EXEC PGM=IDCAMS 00110000
//SYSPRINT DD SYSOUT=* 00120000
//* 00130002
//PSFILE DD DISP=SHR, DSN=ABHI.COURSE.PS 00140002
//ESDSFILE DD DISP=SHR, DSN=ABHI.COURSE.ESDS 00150002
//* 00160002
//SYSIN DD * 00170002
        REPRO INFILE(PSFILE) - 00180003
            OUTFILE(ESDSFILE) 00190003
/* 00200002
```



17. REPRO - PS to RRDS

```
//REPPSRR JOB CLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),MSGCLASS=A, 00010001
//          PRTY=15 00020001
//* 00030001
//* >>>> JCL TO COPY FROM PS TO RRDS <<<< 00040001
//* 00050001
//STEP010 EXEC PGM=IDCAMS 00060001
//SYSPRINT DD SYSOUT=* 00070001
//* 00081002
//PSFILE DD DISP=SHR,DSN=ABHI.COURSE.PS 00082003
//RRDSFILE DD DISP=SHR,DSN=ABHI.COURSE.RRDS 00083004
//* 00084002
//SYSIN DD * 00085002
        REPRO INFILE(PSFILE) - 00086005
            OUTFILE(RRDSFILE) 00087005
/* 00088002
```

18. REPRO - RRDS to RRDS

```
//REPRRRR JOB CLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),MSGCLASS=A, 00010000
//          PRTY=15 00020000
//* 00030000
//* >>>> JCL TO COPY FROM RRDS TO RRDS <<<< 00040000
//* 00050000
//STEP010 EXEC PGM=IDCAMS 00060000
//SYSPRINT DD SYSOUT=* 00070000
//* 00080000
//RRFILE DD DISP=SHR,DSN=ABHI.COURSE.RRDS 00090001
//RRFILE1 DD DISP=SHR,DSN=ABHI.COURSE.RRDS1 00100001
//* 00110000
//SYSIN DD * 00120000
        REPRO INFILE(RRFILE) - 00130000
            OUTFILE(RRFILE1) 00140000
/* 00150000
```

19. REPRO - ESDS to ESDS

```
//REPESES JOB CLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),MSGCLASS=A, 00010000
//          PRTY=15 00020000
//* 00030000
//* >>>> JCL TO COPY FROM ESDS TO ESDS <<<< 00040000
//* 00050000
//STEP010 EXEC PGM=IDCAMS 00060000
//SYSPRINT DD SYSOUT=* 00070000
//* 00080000
//ESFILE DD DISP=SHR,DSN=ABHI.COURSE.ESDS 00090000
//ESFILE1 DD DISP=SHR,DSN=ABHI.COURSE.ESDS1 00100000
//* 00110000
//SYSIN DD * 00120000
        REPRO INFILE(ESFILE) - 00130000
            OUTFILE(ESFILE1) 00140000
```



20. Compare PS datasets

```
//IEBCOMP JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000108
//* 00000208
//* >> COMPARE TWO PS DATASETS << 00000308
//* 00000408
//STEP1 EXEC PGM=IEBCOMPR 00000508
//SYSUT1 DD DSN=ABHI.COURSE.PS1,DISP=SHR <--- DATASETS TO COMPARE 00000608
//SYSUT2 DD DSN=ABHI.COURSE.PS2,DISP=SHR <--- DATASETS TO COMPARE 00000708
//SYSPRINT DD SYSOUT=* 00000808
//SYSOUT DD SYSOUT=* 00000908
//SYSIN DD * 00001008
        COMPARE TYPORG=PS <----- FILE IS PS 00001108
/* 00001208
```

21. Compare PDS datasets

```
//IEBCOMP2 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000105
//* 00000205
//* >> COMPARE TWO PDS DATASETS << 00000305
//* 00000405
//STEP1 EXEC PGM=IEBCOMPR 00000505
//SYSUT1 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <--- DATASETS TO COMPARE 00000605
//SYSUT2 DD DSN=ABHI.COURSE.PDS4,DISP=SHR <--- DATASETS TO COMPARE 00000705
//SYSPRINT DD SYSOUT=* 00000805
//SYSOUT DD SYSOUT=* 00000905
//SYSIN DD * 00001005
        COMPARE TYPORG=PO <----- FILE IS PDS 00001105
/* 00001205
```

22. Copy one PDS to another

```
//IEBCOPY JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000109
//* 00000208
//* >> COPY ONE PDS TO ANOTHER << 00000308
//* 00000408
//STEP1 EXEC PGM=IEBCOPY 00000508
//DD1 DD DSN=ABHI.COURSE.PDS2,DISP=SHR <--- INPUT DATASET 00000610
//DD2 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <--- OUTPUT DATASET 00000710
//SYSPRINT DD SYSOUT=* 00000808
//SYSOUT DD SYSOUT=* 00000908
//SYSIN DD * 00001008
        COPY OUTDD=DD2 <--- DD NAME OF OUTPUT DATASET 00001110
           INDD=DD1 <--- DD NAME OF INPUT DATASET 00001210
/* 00001308
```



23. Copy one PDS to another using SYSUT

```
//IEBCOPY1 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000102
//* 00000201
//* >> COPY ONE PDS TO ANOTHER USING SYSUT << 00000301
//* 00000401
//STEP1 EXEC PGM=IEBCOPY 00000501
//SYSUT1 DD DSN=ABHI.COURSE.PDS2,DISP=SHR <--- INPUT DATASET 00000603
//SYSUT2 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <--- OUTPUT DATASET 00000703
//SYSPRINT DD SYSOUT=* 00000801
//SYSOUT DD SYSOUT=* 00000901
//SYSIN DD DUMMY 00001001
/* 00001101
```

24. Merge Datasets

```
//IEBCOPY2 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000102
//* 00000202
//* >> MERGE DATASETS << 00000302
//* 00000402
//STEP1 EXEC PGM=IEBCOPY 00000502
//DD1 DD DSN=ABHI.COURSE.PDS2,DISP=SHR <---- INPUT DATASET 00000602
//DD2 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <---- INPUT DATASET 00000702
//DD3 DD DSN=ABHI.COURSE.PDS4,DISP=SHR <---- OUTPUT DATASET 00000802
//SYSPRINT DD SYSOUT=* 00000902
//SYSOUT DD SYSOUT=* 00001002
//SYSIN DD * 00001102
COPY OUTDD=DD3 <--- DD NAME OF OUTPUT DATASET 00001202
INDD=DD1 <--- DD NAME OF INPUT DATASET 00001302
INDD=DD2 <--- DD NAME OF INPUT DATASET 00001402
/* 00001502
```

25. SELECTive Copy

```
//IEBCOPY3 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000104
//* 00000204
//* >> SELECTIVE COPY << 00000304
//* 00000404
//STEP1 EXEC PGM=IEBCOPY 00000504
//DD1 DD DSN=ABHI.COURSE.PDS2,DISP=SHR <--- INPUT DATASET 00000604
//DD2 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <--- INPUT DATASET 00000704
//DD3 DD DSN=ABHI.COURSE.PDS4,DISP=SHR <--- OUTPUT DATASET 00000804
//SYSPRINT DD SYSOUT=* 00000904
//SYSOUT DD SYSOUT=* 00001004
//SYSIN DD * 00001104
COPY OUTDD=DD3 <--- DD NAME OF OUTPUT DATASET 00001204
INDD=DD1 <--- DD NAME OF INPUT DATASET 00001304
INDD=DD2 <--- DD NAME OF INPUT DATASET 00001404
SELECT MEMBER=(MEMBER2,MEMBER4,MEMBER5) <-- MEMBERS TO BE COPIED 00001504
/* 00001604
```



26. Rename member while copying

```
//IEBCOPY4 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000103
//* 00000203
//* >> RENAME MEMBER WHILE COPYING << 00000303
//* 00000403
//STEP1 EXEC PGM=IEBCOPY 00000503
//DD2 DD DSN=ABHI.COURSE.PDS2,DISP=SHR <-- INPUT DATASET 00000603
//DD3 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <-- INPUT DATASET 00000703
//DD4 DD DSN=ABHI.COURSE.PDS4,DISP=SHR <-- OUTPUT DATASET 00000803
//SYSPRINT DD SYSOUT=* 00000903
//SYSOUT DD SYSOUT=* 00001003
//SYSIN DD * 00001103
COPY OUTDD=DD4 <--- DD NAME OF OUTPUT DATASET 00001203
INDD=DD2 <--- DD NAME OF INPUT DATASET 00001303
INDD=DD3 <--- DD NAME OF INPUT DATASET 00001403
SELECT MEMBER=( (MEMBER1,NEW1),MEMBER2) <-- MEMBER TO RENAME 00001503
/* 00001603
```

27. Replace member while copying

```
//IEBCOPY5 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000106
//* 00000206
//* >> REPLACE MEMBER WHILE COPYING << 00000306
//* 00000406
//STEP1 EXEC PGM=IEBCOPY 00000506
//DD2 DD DSN=ABHI.COURSE.PDS2,DISP=SHR <-- INPUT DATASET 00000606
//DD3 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <-- INPUT DATASET 00000706
//DD4 DD DSN=ABHI.COURSE.PDS4,DISP=SHR <-- OUTPUT DATASET 00000806
//SYSPRINT DD SYSOUT=* 00000906
//SYSOUT DD SYSOUT=* 00001006
//SYSIN DD * 00001106
COPY OUTDD=DD4 <--- DD NAME OF OUTPUT DATASET 00001206
INDD=DD2 <--- DD NAME OF INPUT DATASET 00001306
INDD=DD3 <--- DD NAME OF INPUT DATASET 00001406
SELECT MEMBER=( (MEMBER2,,R) ) <-- MEMBER TO REPLACE 00001506
/* 00001606
```

28. Compress a PDS

```
//IEBCOPY6 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000103
//* 00000203
//* >> COMPRESS A PDS << 00000303
//* 00000403
//STEP1 EXEC PGM=IEBCOPY 00000503
//DD1 DD DSN=ABHI.COURSE.PDS5,DISP=SHR <-- DATASET TO COMPRESS 00000603
//SYSPRINT DD SYSOUT=* 00000703
//SYSOUT DD SYSOUT=* 00000803
//SYSIN DD * 00000903
COPY OUTDD=DD1 <--- SAME DDNAME IN INDD AND OUTDD 00001003
INDD=DD1 00001103
/* 00001203
```



29. Exclude members while copying

```
//IEBCOPY7 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000102
/* 00000202
/* >> EXCLUDE MEMBERS WHILE COPYING << 00000302
/* 00000402
//STEP1 EXEC PGM=IEBCOPY 00000502
//DD1 DD DSN=ABHI.COURSE.PDS2,DISP=SHR <-- INPUT DATASET 00000602
//DD2 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <-- INPUT DATASET 00000702
//DD3 DD DSN=ABHI.COURSE.PDS4,DISP=SHR <-- OUTPUT DATASET 00000802
//SYSPRINT DD SYSOUT=* 00000902
//SYSOUT DD SYSOUT=* 00001002
//SYSIN DD * 00001102
COPY OUTDD=DD3 <-- DDNAME OF OUTPUT DATASET 00001202
INDD=DD1 <-- DDNAME OF INPUT DATASET 00001302
INDD=DD2 <-- DDNAME OF INPUT DATASET 00001402
EXCLUDE MEMBER=(MEMBER4,MEMBER1) <--- MEMBERS TO IGNORE 00001502
/* 00001602
```

30. Rename and Replace while copying

```
//IEBCOPY8 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000100
/* 00000200
/* >> RENAME AND REPLACE MEMBER WHILE COPYING << 00000300
/* 00000400
//STEP1 EXEC PGM=IEBCOPY 00000500
//DD2 DD DSN=ABHI.COURSE.PDS2,DISP=SHR <-- INPUT DATASET 00000600
//DD3 DD DSN=ABHI.COURSE.PDS3,DISP=SHR <-- INPUT DATASET 00000700
//DD4 DD DSN=ABHI.COURSE.PDS4,DISP=SHR <-- OUTPUT DATASET 00000800
//SYSPRINT DD SYSOUT=* 00000900
//SYSOUT DD SYSOUT=* 00001000
//SYSIN DD * 00001100
COPY OUTDD=DD4 <--- DD NAME OF OUTPUT DATASET 00001200
INDD=DD2 <--- DD NAME OF INPUT DATASET 00001300
INDD=DD3 <--- DD NAME OF INPUT DATASET 00001400
SELECT MEMBER=(MEMBER1,MEMBER2,R) <-- MEMBER TO RENAME/REPLACE 00001500
/* 00001600
```

31. Copy whole job into another dataset

```
//EDIT1 JOB CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01 00000100
/* 00000200
/* >> copy whole job into another dataset << 00000300
/* 00000400
//STEP1 EXEC PGM=IEBEDIT 00000500
//SYSUT1 DD DSN=ABHI.COURSE.PS1,DISP=SHR 00000600
//SYSUT2 DD DSN=ABHI.COURSE.PS2,DISP=SHR 00000700
//SYSPRINT DD SYSOUT=* 00000800
//SYSOUT DD SYSOUT=* 00000900
//SYSIN DD * 00001000
EDIT START=COPY1 00001100
/* 00001200
```



32. Copy multiple job into another dataset

```
//EDIT1      JOB   CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01      00000100
//*                                                  00000200
//*          >> copy multiple job into another dataset <<          00000300
//*                                                  00000400
//STEP1      EXEC  PGM=IEBEDIT                                          00000500
//SYSUT1     DD   DSN=ABHI.COURSE.PS1,DISP=SHR                        00000600
//SYSUT2     DD   DSN=ABHI.COURSE.PS3,DISP=SHR                        00000700
//SYSPRINT   DD   SYSOUT=*                                             00000800
//SYSOUT     DD   SYSOUT=*                                             00000900
//SYSIN      DD   *                                                  00001000
              EDIT  START=COPY1                                        00001100
              EDIT  START=COPY2                                        00001200
/*                                                  00001300
```

33. Copy steps using INCLUDE and EXCLUDE

```
//EDIT1      JOB   CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01      00000100
//*                                                  00000200
//*          >> copy steps using include and exclude <<          00000300
//*                                                  00000400
//STEP1      EXEC  PGM=IEBEDIT                                          00000500
//SYSUT1     DD   DSN=ABHI.COURSE.PS1,DISP=SHR                        00000600
//SYSUT2     DD   DSN=ABHI.COURSE.PS3,DISP=SHR                        00000700
//SYSPRINT   DD   SYSOUT=*                                             00000800
//SYSOUT     DD   SYSOUT=*                                             00000900
//SYSIN      DD   *                                                  00001000
              EDIT  START=COPY1,TYPE=EXCLUDE,STEPNAME=STEPSA        00001102
              EDIT  START=COPY2,TYPE=INCLUDE,STEPNAME=STEPB        00001202
/*                                                  00001300
```

34. Copy steps using POSITION member

```
//EDIT3      JOB   CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01      00000100
//*                                                  00000200
//*          >> copy steps using position parameter <<          00000300
//*                                                  00000400
//STEP1      EXEC  PGM=IEBEDIT                                          00000500
//SYSUT1     DD   DSN=ABHI.COURSE.PS1,DISP=SHR                        00000600
//SYSUT2     DD   DSN=ABHI.COURSE.PS3,DISP=SHR                        00000700
//SYSPRINT   DD   SYSOUT=*                                             00000800
//SYSOUT     DD   SYSOUT=*                                             00000900
//SYSIN      DD   *                                                  00001000
              EDIT  START=COPY1,TYPE=POSITION,STEPNAME=STEPB        00001100
/*                                                  00001200
```



35. List a PDS

```
//LIST      JOB   CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01      00000100
//*                                                00000200
//*          >> LIST a PDS <<                                       00000300
//*                                                00000400
//STEP1     EXEC  PGM=IEHLIST                                           00000500
//DD1       DD   UNIT=DISK,VOLUME=SER=PUB000,DISP=SHR                   00000600
//SYSPRINT  DD   SYSOUT=*                                               00000700
//SYSOUT    DD   SYSOUT=*                                               00000800
//SYSIN     DD   *                                                       00000900
           LISTPDS DSNAME=ABHI.COURSE.PDS1,VOLUME=DISK=PUB000         00001000
/*                                                00001100
```

36. List a VTOC

```
//LIST1     JOB   CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01      00000100
//*                                                00000200
//*          >> LIST a VTOC <<                                       00000300
//*                                                00000400
//STEP1     EXEC  PGM=IEHLIST                                           00000500
//DD1       DD   UNIT=DISK,VOLUME=SER=PUB000,DISP=SHR                   00000600
//SYSPRINT  DD   SYSOUT=*                                               00000700
//SYSOUT    DD   SYSOUT=*                                               00000800
//SYSIN     DD   *                                                       00000900
           LISTVTOC FORMAT,VOLUME=DISK=PUB000                           00001000
/*                                                00001100
```

37. SCRATCH, UNCATALOG a dataset

```
//PROG      JOB   CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01      00000101
//*                                                00000200
//*          >> scratch, uncatlg a Datast <<                               00000300
//*                                                00000400
//STEP1     EXEC  PGM=IEHPROGM                                           00000500
//DD1       DD   UNIT=DISK,VOLUME=SER=PUB001,DISP=SHR                   00000600
//SYSPRINT  DD   SYSOUT=*                                               00000700
//SYSOUT    DD   SYSOUT=*                                               00000800
//SYSIN     DD   *                                                       00000900
           SCRATCH DSNAME=ABHI.COURSE.PS0,VOL=DISK=PUB001             00001000
           UNCATLG DSNAME=ABHI.COURSE.PS0                               00001100
/*                                                00001200
```

38. UNCATALOG a dataset

```
//PROG1     JOB   CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=HERC01      00000101
//*                                                00000200
//*          >> uncatlg a Datast <<                                       00000301
//*                                                00000400
//STEP1     EXEC  PGM=IEHPROGM                                           00000500
//DD1       DD   UNIT=DISK,VOLUME=SER=PUB000,DISP=SHR                   00000601
//SYSPRINT  DD   SYSOUT=*                                               00000700
//SYSOUT    DD   SYSOUT=*                                               00000800
//SYSIN     DD   *                                                       00000900
           UNCATLG DSNAME=ABHI.COURSE.PS1                               00001001
/*                                                00001100
```

39. SORT a field

```
//SORT1      JOB  CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//          PRTY=15      00020000
//*      00030013
//*          >>>>>  SORT BY RELEASE YEAR  <<<<<<<<<      00040013
//*      00050013
//STEP1      EXEC  PGM=SORT      00060013
//SORTIN     DD   DSN=ABHI.COURSE.IMDB,DISP=SHR      00070013
//SORTOUT    DD   DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00080013
//SYSPRINT   DD   SYSOUT=*      00090013
//SYSOUT     DD   SYSOUT=*      00100013
//SYSIN      DD   *      00110013
              SORT  FIELDS=(1,88,CH,A)      00120013
/*      00130013
```

40. SORT multiple fields

```
//SORT2      JOB  CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//          PRTY=15      00020000
//*      00030002
//*          >>>>>>  SORT MULTIPLE FIELDS <<<<<<      00040002
//*      00050002
//STEP1      EXEC  PGM=SORT      00060002
//SORTIN     DD   DSN=ABHI.COURSE.IMDB,DISP=SHR      00070002
//SORTOUT    DD   DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00080002
//SYSPRINT   DD   SYSOUT=*      00090002
//SYSOUT     DD   SYSOUT=*      00100002
//SYSIN      DD   *      00110002
              SORT  FIELDS=(89,14,CH,D,      00120002
                        155,4,CH,D)      00130002
/*      00140002
```

41. SORT multiple fields using FORMAT keyword

```
//SORT3      JOB  CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//          PRTY=15      00020000
//STEP1      EXEC  PGM=SORT      00030000
//SORTIN     DD   DSN=ABHI.COURSE.IMDB,DISP=SHR      00040002
//SORTOUT    DD   DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00050002
//SYSPRINT   DD   SYSOUT=*      00060002
//SYSOUT     DD   SYSOUT=*      00070002
//SYSIN      DD   *      00080002
              SORT  FORMAT=CH,FIELDS=(149,4,D,155,4,D)      00090002
/*      00100000
```



42. Copy a dataset using SORT

```
//SORT4      JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//              PRTY=15      00020000
//*      00030003
//*              >>>>>>> COPY A DATASET <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<< 00040003
//*      00050003
//STEP1      EXEC PGM=SORT      00060000
//SORTIN     DD DSN=ABHI.COURSE.IMDB,DISP=SHR      00070003
//SORTOUT    DD DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00080003
//SYSPRINT   DD SYSOUT=*      00090003
//SYSOUT     DD SYSOUT=*      00100003
//SYSIN      DD *      00110003
              OPTION COPY      00120003
/*      00130000
```

43. SORT with INCLUDE

```
//SORT6      JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//              PRTY=15      00020000
//*      00030004
//*              >>>>>>> SORT WITH INCLUDE <<<<<<<< 00040004
//*      00050004
//STEP1      EXEC PGM=SORT      00060000
//SORTIN     DD DSN=ABHI.COURSE.IMDB,DISP=SHR      00070004
//SORTOUT    DD DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00080004
//SYSPRINT   DD SYSOUT=*      00090004
//SYSOUT     DD SYSOUT=*      00100004
//SYSIN      DD *      00110004
              SORT FIELDS=(149,4,CH,D)      00120004
              INCLUDE COND=(89,12,CH,NE,C'Documentary')      00130004
/*      00140000
```

44. SORT INCLUDE with multiple conditions

```
//SORT7      JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//              PRTY=15      00020000
//*      00030002
//*              >>>>>>> SORT INCLUDE WITH MULTIPLE COND <<<<<<< 00040002
//*      00050002
//STEP1      EXEC PGM=SORT      00060002
//SORTIN     DD DSN=ABHI.COURSE.IMDB,DISP=SHR      00070002
//SORTOUT    DD DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00080002
//SYSPRINT   DD SYSOUT=*      00090002
//SYSOUT     DD SYSOUT=*      00100002
//SYSIN      DD *      00110002
              SORT FIELDS=(1,88,CH,A)      00120002
              INCLUDE COND=(149,4,CH,EQ,C'2018',OR,      00130002
                            155,4,CH,GT,C'7.0')      00140002
/*      00150002
```



45. SORT with OMIT condition

```
//SORT8      JOB  CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//          PRTY=15      00020000
//*      00030002
//*      >>>>>> SORT WITH OMIT CONDITION <<<<<<<<      00040002
//*      00050002
//STEP1     EXEC  PGM=SORT      00060000
//SORTIN    DD   DSN=ABHI.COURSE.IMDB,DISP=SHR      00070002
//SORTOUT   DD   DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00080002
//SYSPRINT  DD   SYSOUT=*      00090002
//SYSOUT    DD   SYSOUT=*      00100002
//SYSIN     DD   *      00110002
            SORT  FIELDS=(1,88,CH,A)      00120002
            OMIT  COND=(149,4,CH,LT,C'2010',AND,      00130002
                        155,4,CH,LT,C'7.0')      00140002
/*      00150000
```

46. SORT with OUTREC

```
//SORT12     JOB  CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//          PRTY=15      00020000
//STEP1     EXEC  PGM=SORT      00030000
//SORTIN    DD   DSN=ABHI.COURSE.IMDB,DISP=SHR      00040001
//SORTOUT   DD   DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00050001
//SYSPRINT  DD   SYSOUT=*      00060001
//SYSOUT    DD   SYSOUT=*      00070001
//SYSIN     DD   *      00080001
            SORT  FIELDS=(149,4,BI,D)      00090001
            OUTREC FIELDS=(160,8,149,4)      00100001
/*      00110000
```

47. OUTREC using FINDREP

```
//SORT22     JOB  CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1),      00010000
//          PRTY=15      00020000
//*      00030002
//*      >>>>>>>> OUTREC using FINDREP <<<<<<<<<<<<<<<<<<<<<<<<<<<<      00040002
//*      00050002
//STEP1     EXEC  PGM=SORT      00060000
//SORTIN    DD   DSN=ABHI.COURSE.IMDB,DISP=SHR      00070002
//SORTOUT   DD   DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR      00080002
//SYSPRINT  DD   SYSOUT=*      00090002
//SYSOUT    DD   SYSOUT=*      00100002
//SYSIN     DD   *      00110002
            OPTION COPY      00120002
            OUTREC FINDREP=(IN=C'Feature Film',OUT=C'Movie')      00130002
/*      00140000
```



48. OUTREC only specific columns

```
//SORT12 JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1), 00010000
//          PRTY=15 00020000
//* 00030005
//* >>>>> OUTREC ONLY SPECIFIC COLUMNS <<<<<< 00040005
//* 00050005
//STEP1 EXEC PGM=SORT 00060000
//SORTIN DD DSN=ABHI.COURSE.IMDB,DISP=SHR 00070005
//SORTOUT DD DSN=ABHI.COURSE.IMDB.OUTPUT,DISP=SHR 00080005
//SYSPRINT DD SYSOUT=* 00090005
//SYSOUT DD SYSOUT=* 00100005
//SYSIN DD * 00110005
        SORT FIELDS=(149,4,BI,D) 00120005
        OUTREC BUILD=(9:149,4,30:155,4) 00130005
/* 00140000
```

49. SPLIT file based on condition

```
//SORT25 JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1), 00010000
//          PRTY=15 00020000
//* 00030004
//* >>>>>>> SPLIT FILE BASED ON CONDITION <<<<<<<< 00040004
//* 00050004
//STEP1 EXEC PGM=SORT 00060000
//SORTIN DD DSN=ABHI.COURSE.IMDB,DISP=SHR 00070004
//SORTOF01 DD DSN=ABHI.COURSE.IMDB.Y2017,DISP=SHR 00080004
//SORTOF02 DD DSN=ABHI.COURSE.IMDB.Y2018,DISP=SHR 00090004
//SORTOF03 DD DSN=ABHI.COURSE.IMDB.Y2019,DISP=SHR 00100004
//SYSPRINT DD SYSOUT=* 00110004
//SYSOUT DD SYSOUT=* 00120004
//SYSIN DD * 00130004
        OPTION COPY 00140004
        OUTFIL FILES=01,INCLUDE=(149,4,CH,EQ,C'2017') 00150004
        OUTFIL FILES=02,INCLUDE=(149,4,CH,EQ,C'2018') 00160004
        OUTFIL FILES=03,INCLUDE=(149,4,CH,EQ,C'2019') 00170004
/* 00180000
```

50. Multiple copies using OUTFIL

```
//SORT26 JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1), 00010000
//          PRTY=15 00020000
//* 00030005
//* >>>>>>> MULTIPLE COPIES USING OUTFIL <<<<<<<< 00040005
//* 00050005
//STEP1 EXEC PGM=SORT 00060000
//SORTIN DD DSN=ABHI.COURSE.IMDB,DISP=SHR 00070005
//SORTOF01 DD DSN=ABHI.COURSE.IMDB.Y2017,DISP=SHR 00080005
//SORTOF02 DD DSN=ABHI.COURSE.IMDB.Y2018,DISP=SHR 00090005
//SORTOF03 DD DSN=ABHI.COURSE.IMDB.Y2019,DISP=SHR 00100005
//SYSPRINT DD SYSOUT=* 00110005
//SYSOUT DD SYSOUT=* 00120005
//SYSIN DD * 00130005
        OPTION COPY 00140005
        OUTFIL FILES=(01,02,03) 00150005
/* 00160000
```



51. SPLIT a dataset into equal parts

```
//SORT27 JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID,MSGLEVEL=(1,1), 00010000
//          PRTY=15 00020000
//* 00030002
//* >>>>>>>>> SPLIT A DATASET INTO EQUAL PARTS <<<<<<<<<< 00040002
//* 00050002
//STEP1 EXEC PGM=SORT 00060000
//SORTIN DD DSN=ABHI.COURSE.IMDB,DISP=SHR 00070002
//SORTOF01 DD DSN=ABHI.COURSE.IMDB.Y2017,DISP=SHR 00080002
//SORTOF02 DD DSN=ABHI.COURSE.IMDB.Y2018,DISP=SHR 00090002
//*ORTOF03 DD DSN=ABHI.COURSE.IMDB.Y2019,DISP=SHR 00100002
//SYSPRINT DD SYSOUT=* 00110002
//SYSOUT DD SYSOUT=* 00120002
//SYSIN DD * 00130002
        OPTION COPY 00140002
        OUTFIL FILES=(01,02),SPLIT 00150002
/* 00160000
```

 Mainframe Nation



About Me



My name is Abhishek Rathi. I am an Electronics & Communications Engineer and a Mainframe Developer.

My Mainframe experience involves COBOL, CICS, JCL, TSO, ISPF, VSAM, DB2 SQL.

I published my first Udemy course on Jan 2017. There were no courses about Mainframes on Udemy at that time.

I have since released 5 courses on Mainframes. Over the time the content of all these 5 courses has been consolidated into 2 courses.

The first course titled TSO/ISPF is meant for those who have no knowledge about Mainframes and want to have a head start in this field. It covers TSO/ISPF and other Mainframe Technologies like JCL, VSAM, CICS, COBOL. The second course covers JCL in detail.


All my courses are hands-on practical training.

You can get in touch with me online through my website or my YouTube account:



My Other Courses

Currently I have 3 Courses on Mainframes.



BESTSELLER


The Complete Mainframe Professional Course :...

Abhishek Rathi

★★★★★ 4.4 (823)

~~\$194.99~~ **\$11.99**

[ENROLL](#)



BESTSELLER

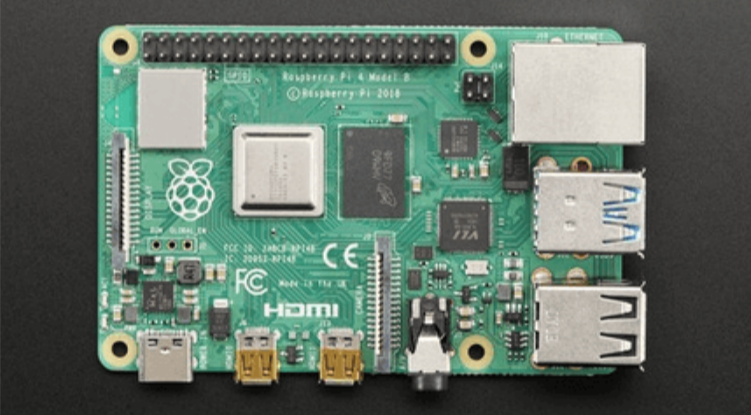
The Complete JCL (Job Control Language) Course

Abhishek Rathi

★★★★★ 4.4 (248)

~~\$194.99~~ **\$11.99**

[ENROLL](#)



Mainframe Pi: Turn your Raspberry Pi into a...

Abhishek Rathi

★★★★★ 4.4 (11)

Free

[ENROLL](#)



"Excellent course. Recommended!"

- Ailton José Nogueira
Udemy student

"Abhishek, This course gives a very clear understanding for JCL basics. Amazing content. Appreciate!"

- Adam Vegh
Udemy student

"Till now I find the course very detailed and absorbing. I work on mainframes and use JCL day-in and day-out. These videos definitely did a good job of defining the basics of JCL."

- Samintra Majumdar
Udemy student

"I know nothing about mainframes and here I am, learning in this course. I recommend this course for everyone that has no prior knowledge about mainframes, or whoever would like to enrich his knowledge. Enjoy!"

- Mario Azzi
Udemy student