

# COBOL Bootcamp



**18 days**

Basic COBOL z/OS training for programmers in the z/OS environment for students with a higher education background and some programming experience

# Introduction

This course will give students basic skills in COBOL, TSO/ISPF, JCL and SQL/DB2 to participate in development and maintenance projects of COBOL based systems in the z/OS IBM Mainframe environment.

Theory and practice are mixed in a way that you benefit fully of the topics explained.

The instructors have experience from full-time training with IBM and IBMs training providers for many years.

# Agenda

- Day 1-2**      z/OS TSO/ISPF Fundamentals
- Day 3-7**      COBOL Fundamentals I
- Day 8-11**    z/OS JCL & Utilities Fundamentals
- Day 12-15**   COBOL Fundamentals II
- Additional Programming Exercises
- Day 15**        Final Quiz I
- Day 16-18**    SQL Programming Fundamentals
- Day 18**        Final Quiz II

## Courseware.

- z/OS TSO/ISPF Fundamentals
- z/OS COBOL Fundamentals
- z/OS JCL & Utilities Fundamentals
- SQL Programming Fundamentals

Students will also be given the “Mainframe Cobol” book from Mike Murach & Associates Inc, ISBN 1-890774-24-3

<http://murach.com/books/mainframe.htm>

## Other reference materials

### *IBM z/OS Internet Library*

- IBM Manuals

<http://www-03.ibm.com/systems/z/os/zos/bkserv/>

### *IBM Redbooks*

- Introduction to the New Mainframe: z/OS Basics

<http://www.redbooks.ibm.com/abstracts/sg246366.html>

# z/OS TSO/ISPF Fundamentals

## Audience

Beginning COBOL Programmers with no experience of the z/OS environment, but have a common understanding of operating systems

## Objectives

- To log on to TSO and start ISPF
- To navigate through ISPF/PDF dialogs and to use basic ISPF functions and the ISPF Editor
- To use ISPF/PDF to allocate datasets and edit datasets via the ISPF Editor primary and line commands
- To use ISPF to manipulate; copy, rename, delete, list, sort, datasets
- To describe TSO/E commands

## Contents

- **z/OS overview**  
z/Architecture overview, z/OS Overview, The concept of Virtual Storage, Job Entry Subsystem, Catalog Management, System Managed Storage: SMS, Establishing a TSO Session, ISPF
- **An introduction to ISPF**  
What is ISPF?, Dialog Manager, Panel Hierarchy, General structure of ISPF Panels, Navigating in ISPF, Jump Function and Command Stacking, Action Bar Menus, Pop-Up Window, Settings, View a Dataset, Edit Dataset, SDSF  
Lab Exercises
- **Working with ISPF**  
Allocating Datasets; Logical and Physical Records, Record Formats and Blocking, Dataset Types, VSAM Dataset Types, Dataset Naming Rules, Pritioned Dataset Naming Rules, Dataset Naming Conventions, Entering Dataset and Library Name, Entering Member Names, ISPF Primary Options Meny, Utility Selection Panel, Allocate New Dataset Panel, Dataset Information  
Lab Exercises
- **Editing Datasets – The ISPF Editor**  
Edit Entry Panel, Edit Member Selection Panel, The ISPF Editor and Profile, Scrolling, More Scrolling, Scrolling Left and Right, Split Screen, Swap, Swap List, Swap Next, Swapbar., Save and Cancel, Find Command, Bound and Exclude Commands, Change Command, Create Command, Replace Command, Copy Command, Sort Command

## Lab Exercises

The Prefix area, Prefix Area Commands; Copy, Move, Repeat Single line and multiple lines, Insert and Delete, single and multiple lines, Cols Command, Cut and Paste, Additional Commands

## Lab Exercises

- **Copying and Moving Datasets**

The Move/Copy Utility – From and To Datasets, Copy/Move Member – From and to Datasets, Member Selection Panel

- **Renaming a Dataset or Member**

Specifying Old and New Dataset, Specifying Old and New Member-name

- **Deleting a Dataset or Member**

Specifying the Dataset to be deleted, The Confirm Pop-Up Window, Specifying a Member to be deleted, The Confirm Pop-Up Window, The Member Deleted Message

## Lab Exercises

- **Working with Dataset Lists**

Dataset List Utility Panel, DSLIST Display Panel, Dataset List Actions Pop-Up Window

## Lab exercises

- **Working with TSO/ E**

TSO/E Commands; General Layout, General Notation, Help Command, Allocate, Delete, Listds, Free, Transmit, Receive

## Lab Exercises

- **Working with JCL**

JES2 Processing Phases, Spool, Jobqueues, Sysout

Introduction to JCL

Submit, SDSF

## Lab Exercises

# COBOL Fundamentals

## Audience

Beginning COBOL Programmers with little or no experience of programming in the COBOL language but a programming fundamentals background.

## Objectives

After completing the course you should:

- Understand how to take advantage of COBOL features
- Define and initialize data contained within the program's Working-Storage Section or external to the program.
- Use COBOL statements to perform arithmetic logic operations.
- Create definitions for both sequential and random file processing
- Define and manipulate data within tables.
- Use COBOL statements to invoke a subprogram, including passing the data between the two programs.
- Understand time and date functions intrinsic functions
- Utilize string related function and verbs

## Contents

- **COBOL overview**

What is COBOL, sample business problem, sample flowchart, sample pseudo-code, COBOL program organization, COBOL language structure, COBOL syntax diagrams, COBOL reference format

- **The organization of a COBOL program**

Identification Division, Environment Division, Data Division, Data Item descriptions, Level Numbers, Level Indicator, Special Level Numbers, Examples, Data-Names, Data-Names Qualification, Filler, Picture Clause, Picture Clause Symbols, Usage Clause, Internal Representation of data, Value Clause, Figurative Constants, Classes of Data, Editing, Redefines, Samples.

Lab exercises.

- **Procedure Division**

Procedure Division and Moves; Paragraphs, Terminator Statements, Other Terminators, Scope Terminators, Move statement, Corresponding phrase, Numeric-Edited Moves,, Reference Modification, Set to True statement, Initialize statement

Lab Exercises

Calculation Statements; Add statement- different formats, Add Corresponding, On Size Error, Intermediate Results, Numeric Data, Subtract statement, Multiply statement, Divide statement, Compute statement.

Lab Exercises

Conditional Logic

Perform statement formats, Inline Perform, Out-Of-Line Perform, Relational expressions, If statement, Compound Conditionals, Class Condition, Compound Conditions, Continue statement, Evaluate statement

Lab Exercises

Date, Time and Other issues; Accepts statement – different formats, Display statement

Lab Exercises

- **Files in COBOL**

Environment Division – Input-Output Section; File Control – Sequential, File Control – Indexed, File Status

Data Division – File Section; File Definitions, Move Mode vs Locate Mode

- **Procedure Division – File Processing statements**

Open, Close, Read Sequential Access, Read Random Access, Read Dynamic Access, Start statement, Write statement, Printer Control, Write for VSAM indexed, Rewrite statement, Delete statement

Lab Exercises

- **Table Handling**

Data Division – Table Handling; Occurs clause – fixed length tables, variable length tables, Indexing, Array organization

Procedure Division – Table Handling; Subscripting, Search statement – serial, Set Index statement, Search statement – Binary

Lab Exercises

- **Library Services**

Cope statement, nested Copy, Copy Replacing, Introduction to Language Environment – LE

- **Sort/Merge**

COBOL Program Organization – Sort/Merge, Sort statement Using, Giving, Merge statement, Sort Procedure, Release statement, Return statement, Sort-Return, Sort Control Dataset

Lab Exercises

- **Call and Linkage**

Procedure Division – Call statement, Call by Content/Reference, Data Division – Linkage Section, Pointer, Procedure Division and Linkage, Set Address, Special Registers

Lab Exercises

- **Run Unit**

Run Unit, Nested Programs, Common Programs, Global Data, External Data

Lab Exercises

- **COBOL Intrinsic Functions**

COBOL Program Organization – Procedure Division, Intrinsic Function, Intrinsic Functions Syntax, Intrinsic Functions – Argument and Values, Date Formats, COBOL Intrinsic Functions; Current-Date, Integer-Of-Date, Date-Of-Integer, Integer-Of-Day, Day-Of-Integer, Date-To-YYYYMMDD, Day-To-YYYYMMDD, Nesting Functions, Length, Lower-Case, Upper-Case, Reverse, Arithmetical, Business and Mathematical Functions, Numval-C

Lab Exercises

- **Advanced COBOL Statements**

COBOL Program Organization – Procedure Division; Inspect Tallying, Inspect Replacing, Inspect Converting, String, Unstring

Lab Exercises

- **Compiler and Run Time Options**

COBOL Compiler and Run Time Options, LE Run-Time Option



# z/OS JCL & Utilities Fundamentals

## Audience

Beginning COBOL Programmers with experience from the TSO/ISPF environment in the z/OS Operating System, who want to be able to manage their execution environment of Batch COBOL Programs.

## Objectives

- Code basic JCL statements using proper syntax and coding rules to:
  - Create new datasets, referencing existing datasets
  - Condition-code testing using IF/THEN/ELSE/ENDIF constructs
  - Create and use Generation Data Groups, GDG
  - Code Instream and Cataloged Procedures using symbolic parameters and overrides
  - Use selected Utility Programs as IEBGENER, IEBCOPY and SORT

## Contents

- **Introduction to JCL**  
Operating Systems, Programs and Data, Job Control Language, JES Responsibilities, JOB statement, EXEC statement, JCL Errors, Return-Codes
- **JOB, EXEC and DD Statements**  
JCL statement format, Parameters  
The JOB statement; JOB statement syntax, CLASS, MSGCLASS, MSGLEVEL, NOTIFY, TYPRUN and REGION parameters  
The EXEC statement syntax, Program execution, TIME parameter  
The DD-statement; Why Data Definitions, Accessing a Dataset, Dataset Concatenation, Data in the input-stream, SYSOUT processing, Comment statement  
Lab Exercises
- **DD Parameters, a second look**  
DD statement syntax, Permanent Dataset naming, UNIT, VOLUME, Volume Table of Contents, VTOC, DSCB, Space specifications, Release unused space, Disposition – Syntax and Defaults, Disposition parameter; NEW, OLD, KEEP, CATLG, DELETE, MOD,PASS, Backward Reference, Temporary Datasets, Special DD-names, JOBLIB, STEPLIB  
Lab Exercises

- **Introduction to Utilities and Conditional Execution**

Classes of Utilities, Utility Selection, Utility Control Statement Format, IEBGENER, IEBPTPCH, IEHLIS, IDCAMS, Conditional execution of JOB steps, Return Code Setting, COND parameter on JOB, COND parameter on EXEC, Abnormal Termination, IF/THEN/ELSE/ENDIF JCL constructs

Lab Exercises

- **Data Management, Organization and Format**

Record Formats, Data Control Block parameter, Fixed Length Record Format, Variable Length Record Format, Undefined Length Record Format, System Determined Blocksize, Dataset Organizations; Sequential, Partitioned, Direct, Storage Management Subsystem, SMS, AVGREC, Data Class, Partitioned Organization – PDSE, VSAM Datasets, RECORG, KEYOFF

Lab Exercises

- **Generation Data Groups**

Generation Data Group – The Need, IDCAMS Define GDG, LIMIT, SCRATCH, NOEMPTY etc., Rollin – Rolloff, GDG in Multistep Job

Lab Exercises

- **Procedures**

Instream Procedure, Cataloging a Procedure, Procedure Modifications, Modifying EXEC-, DD-statements, symbolic parameters

Lab Exercises

- **More about Utilities**

IEBGENER Copy, IEBCOPY Copy, IEBCOPY Listing, IEBUPDTE

Lab Exercises

- **More on Procedures**

Procedures and Include Groups, Accessing the PROC – JCLLIB statements, The INCLUDE statement, The SET statement, Overriding Symbolic Parameters

Lab Exercises

- **Sort/Merge**

Sort Process, EBCDIC Collating Sequence, Control Statement Format, SORT Statement Format, Sort on two fields, Sort JCL Statements, Merge Process, Merge Control Statement, Merge JCL Statements

Lab Exercises

# SQL Programming Fundamentals

## Audience

Beginning COBOL Programmers with some experience of programming using the COBOL language.

## Objectives

After completing the course you should:

- Understand the relational model
- Have a basic understanding of the Data Definition Language, DDL, Database, Tablespace, Table and Index
- Understand the Data Manipulation Language: SELECT, INSERT, UPDATE and DELETE
- Use SPUFI for sending queries to DB2 and analyse the result
- Use DCL-gen to generate the COBOL structure for a given table.
- Understand how host-variables are used in a COBOL-program
- Use SQLCA in a COBOL-program to check the outcome of a SQL statement
- Understand the concept of a Cursor to fetch multiple rows
- Test and debug a simple COBOL-SQL program

## Contents

- **The Relational Model**  
Background and theory
- **SQL-DDL Data Definition Language**  
Create Database, Tablespace, Table and Index  
Lab exercises.
- **SQL-DML Data Manipulation Language Basics**  
Select, Order By, Union, Union All, Operators, Between, Like, Is Not, Inner Join  
Lab Exercises
- **SQL-DML Functions**  
Column- and scalar- functions, Group By, Having, Concatenation, Subselect, Handling Nulls, Inner Join  
Lab Exercises

- **SQL-DML Updates**

  - Delete, Insert and Update

  - Inner Join

  - Lab Exercises

- **SQL and the COBOL Program**

  - The COBOL Compiler built in SQL Pre-Compiler, SQLCA, DCLGEN and Host Variables, Handling single row results, Cursor Basics, EXEC SQL DECLARE CURSOR, Using a Cursor, Fetching rows from a cursor

  - Lab Exercises

- **Common SQL-codes**