




# TSO/E REXX

für

-  **Systemprogrammierer**
-  **Operator**
-  **Anwendungsentwickler**

Version 1.1 vom November 2022

Autoren: Wolfram Greis / Rolf Brüning

**European Mainframe Academy GmbH**  
Am Kloostergarten 3  
D 78337 Öhningen  
Tel. +49-7735-938 8668  
ann-chatrine.mueller@mainframe-academy.de

**European Mainframe Academy AG**  
Obergass 23  
CH 8260 Stein am Rhein  
Tel. +41-79-340 64 52  
wolfram.greis@mainframe-academy.de

**Inhaltsverzeichnis**

1	Ziele des Ausbildungsmoduls .....	3
2	Randbedingungen / Voraussetzungen .....	3
3	TSO/E REXX Inhaltsbeschreibung .....	4

## 1 Ziele des Ausbildungsmoduls

Ziel des Moduls ist es, Teilnehmern die in Mainframe-Umgebungen intensiv genutzte Scriptsprache REXX (in Verbindung mit TSO/E) zu vermitteln.

## 2 Randbedingungen / Voraussetzungen

Die Kenntnisse der grundlegenden Logik für prozedurale Programmierung wird vorausgesetzt. Ausserdem der Umgang mit TSO / ISPF / JCL und SDSF.

Den Teilnehmern wird für die Dauer der Ausbildung ein Zugang auf dem aktuellen z/OS System der EMA zur Verfügung gestellt. Dieser Zugang steht auch nach Beendigung der Ausbildung zur Verfügung. Die EMA garantiert diesen Zugang für mindestens drei Monate nach Ausbildungsende.

Der Zugang ist über **VIRTEL** möglich, d.h., dass für den Zugriff auf das EMA-System keine 3270-Emulation installiert werden muss. Der Zugriff ist über einen beliebigen Browser möglich.

Den Teilnehmern wird für die Dauer des Kurses auf der Lernplattform der EMA ein Kursbereich eingerichtet, über den Unterlagen zur Verfügung gestellt werden und in dem ein Forum eingerichtet wird, so dass jederzeit Fragen gestellt und Diskussionen geführt werden können.

Als Begleitmaterial werden Unterlagen und Folien der EMA eingesetzt. Ergänzend werden weitere Ressourcen (z.B. Verweise Redbooks etc.) zur Verfügung gestellt.

### 3 TSO/E REXX Inhaltsbeschreibung

#### Ziele

Die Teilnehmer kennen den Aufbau und die Struktur von REXX Programmen. Sie können REXX eigenständig REXX-Programme für Administrationsaufgaben und andere Zwecke erstellen.

#### Inhalt

##### Überblick

Entwicklung und Ausführung  
Systemumgebung und Libraries  
Komponenten  
Erstellung, Abspeicherung  
Expliziter und impliziter Aufruf  
Variablen  
Ausdrücke  
Keyword Instructions  
Systembibliotheken

##### Prozedur-Steuerung

IF THEN ELSE  
SELECT/WHEN-Konstrukte  
DO-Konstrukte

##### Eingebaute Funktionen

Überblick  
Arithmetische Funktionen  
Vergleichsfunktionen  
Zeichenkettenfunktionen

##### Unterroutinen (Prozeduren)

Interne und externe Unterroutinen  
Erstellung  
Aufruf  
Parameterübergabe  
Selbsterstellte Funktionen  
Fehlerbehandlung  
Returncode (RC)

##### Datenmanipulation

Zusammengesetzte Variablen  
Zerlegen (Parsing)  
Schablonen

##### Data Stack (Datenstapel)

ADDRESS-Instruktionen  
TSO/E externe Funktionen  
Ein-/Ausgabeverarbeitung  
PULL - QUEUE – PUSH  
MAKEBUF – DROPBUF

##### Dateiverarbeitung

Batchverarbeitung  
ALLOC - FREE – EXECIO  
REXX und TSO-Befehle als Batch-Job

##### Ablaufverfolgung, Problemdiagnose

Testhilfen  
TRACE  
Interaktives Debugging