

Introduction to CGIs on z/OS

Course Summary

Description

CGI stands for Common Gateway Interface, and it is the most widely used programming technology for creating dynamic web pages at the server. Students who complete this course will be able to design CGI scripts and programs to run on z/OS used as a web server.

Objectives

After taking this course, students will be able to:

- Define what a CGI is
- Describe the context a CGI runs in
- Describe how a CGI is invoked from a web page
- Describe the differences between GET and POST CGI processing
- Describe what the necessary environment variables are and how to access them
- Describe how to emit responses to the server from a CGI
- Specify how to emit non-EBCDIC data from a CGI
- Suggest how to approach debugging and testing
- Submit batch jobs from a CGI
- Plan for / design a CGI interface that uses cookies to keep track of the process of a dialog
- Describe the processing concerns and issues facing the design of a CGI that accesses DB2 data

Topics

- Review of basic Web concepts
- Overview of CGI processing
- Review of Forms and Controls
- Building responses to client requests
- GET request processing
- POST request processing
- Error handling
- Testing
- Debugging CGIs
- Emitting non-EBCDIC output
- DB2 Access
- The RRSF Interface
- Submitting batch jobs
- nph processing
- The Design of Web-based dialogs

Audience

This course is designed for programmers, experienced with working in an OS/390 or z/OS environment, who will be designing and coding CGI applications that are to be run on z/OS.

Prerequisites

At the very least, the student entering this course should have some background in z/OS UNIX and HTML or XHTML, such as might be obtained from courses PT5642 ("Introduction to z/OS UNIX") and PT5644 ("You and z/OS and the World Wide Web").

Duration

One day

Introduction to CGIs on z/OS

Course Outline

- I. Introduction**
 - A. The Web - Basic Concepts Review
 - B. CGI Overview
 - C. HTTP Servers on z/OS
- II. HTML 5 - Selected Topics**
 - A. Review from Introduction to z/OS UNIX
 - B. Additional tags: <H1> - <H6>,
, <DIV>
 - C. Useful attributes
 - D. Events and event handlers
 - E. Introduction to JavaScript
- III. Forms and INPUT Controls**
 - A. Introduction to Forms and Controls
 - B. The FORM Element
 - C. The INPUT Element
 - D. Successful controls
 - E. Submit processing
 - F. Accessing Programs Without Forms
- IV. Building a Response**
 - A. Responses
 - B. Response Headers
 - C. Writing to stdout
 - D. Sample CGIs: C, COBOL, Assembler, PL/I, REXX, Shell Script
- V. GET Processing**
 - A. The GET environment
 - B. Parsing the QUERY_STRING
- VI. POST Processing**
 - A. The POST environment
 - B. Determine the size of data (CONTENT_LENGTH)
 - C. Determine the boundary string (CONTENT_TYPE)
 - D. Determining the server being used (SERVER_SOFTWARE)
 - E. Accessing the Input Data
 - F. Analyzing the Data
- VII. Error Handling, Testing, and Debugging CGIs**
 - A. Error Handling
 - B. Testing
 - C. Debugging
- VIII. Other Topics**
 - A. Emitting non-EBCDIC Output
 - B. DB2 Access
 - C. The RRSAP interface
 - D. Submitting batch jobs
 - E. nph Processing
 - F. Creating the Illusion of a conversation (Cookies)
- IX. The Design of Web-based dialogs**
 - A. The Server
 - B. Pages
 - C. Links
 - D. Scripts
 - E. Forms
 - F. CGIs
 - G. State Tools
 - H. User IDs
- X. Conclusion**