

TACL Programming I for NonStop Servers U8636S

HPE course number	U8636S
Course length	3 days
Delivery mode	ILT, VILT
View schedule, local pricing, and register	View now
View related courses	View now

This course teaches you the basics of writing support utilities in the Tandem Advanced Command Language (TACL). Through hands-on exercises, you will gain valuable experience in using the fundamental features of TACL programming. After completing this course, you will be able to write TACL macros and routines, use the TACL debugging facility, and write TACL code to interact with and control NonStop Server processes using the Inline facility.

Why HPE Education Services?

- IDC MarketScape leader 5 years running for IT education and training*
- Recognized by IDC for leading with global coverage, unmatched technical expertise, and targeted education consulting services*
- Key partnerships with industry leaders OpenStack®, VMware®, Linux®, Microsoft®, ITIL, PMI, CSA, and SUSE
- Complete continuum of training delivery options—self-paced eLearning, custom education consulting, traditional classroom, video on-demand instruction, live virtual instructor-led with hands-on lab, dedicated onsite training
- Simplified purchase option with HPE Training Credits

Audience

- System programmers

Prerequisites

- Concepts and Facilities or familiarity with NonStop utility

Course objectives

At the conclusion of this course you should be able to:

- Write system management utility programs by learning to use the fundamental features of the TACL language

Detailed course outline

Module 0 Overview	<ul style="list-style-type: none"> • Course objective 	<ul style="list-style-type: none"> • Schedule
Module 1 Overview of TACL Features	<ul style="list-style-type: none"> • TACL as a command interpreter 	<ul style="list-style-type: none"> • Productivity tools and information aids
Module 2 TACL Variables	<ul style="list-style-type: none"> • Obtaining information about variables using either commands or built-ins • Using commands or built-in functions to create, initialize, modify, and eliminate variables • The use of “frames” and variable management 	<ul style="list-style-type: none"> • Lab exercise (30 minutes): Learn and understand how to logon and use defined function keys • Practice using TACL interactively
Module 3 Editing Variables	<ul style="list-style-type: none"> • Performing variable file I/O • Performing global editing of a variable • Performing line editing of a variable • Performing character editing of a variable 	<ul style="list-style-type: none"> • Locating the position of a string within a variable • Extracting lines and characters from a variable • Lab exercise (1 hour): Exercising variable editing
Module 4 Writing Functions: Macros	<ul style="list-style-type: none"> • Macro functionality and syntax • Parameter substitution 	<ul style="list-style-type: none"> • Writing macros • Using nested macros • Lab exercise (30 minutes): Write a simple TACL macro
Module 5 Writing Functions: #IF	<ul style="list-style-type: none"> • Write functions that use the TACL #IF THEN ELSE construct • Making string comparisons 	<ul style="list-style-type: none"> • Using #IF NOT • Lab exercise (1 hour): Write a macro that accepts one or more arguments and uses #IF
Module 6 Writing Functions: #LOOP	<ul style="list-style-type: none"> • Using the #LOOP DO UNTIL construct • Using the #LOOP WHILE DO construct 	<ul style="list-style-type: none"> • Lab exercise (1 hour): Write a macro that uses a #LOOP construct
Module 7 Writing Functions: #CASE	<ul style="list-style-type: none"> • Syntax and use of the #CASE construct 	
Module 8 Debugging TACL Functions	<ul style="list-style-type: none"> • Using the TACL debugging facility to aid in getting functions to work properly • Lab exercise (2 hours): • Start and stop the debugger 	<ul style="list-style-type: none"> • Set and clear breakpoints • Display and modify the contents of a variable • Single step through functions and resume execution • Write a function that makes use of the #CASE built-in

Module 9 Writing Functions: Routines	<ul style="list-style-type: none"> • Writing “Routine” type functions; using #ARGUMENT, #MORE AND #REST • Describe the additional capabilities that routines offer that macros do not 	<ul style="list-style-type: none"> • Describe the use of the built-ins: #MYSYSTEM #PROCESSORSTATUS, and #PROCESSORTYPE • Lab exercise (2 hours): <ul style="list-style-type: none"> – Modify TACL programs to use routine type functions
Module 10 INLINE Processing	<ul style="list-style-type: none"> • Performing process I/O using the INLINE facility • Controlling the display of process output • Describe the use of #INLINEPREFIX, INLPREFIX, #INLINETO, and INLTO 	<ul style="list-style-type: none"> • Lab exercise (1 hour) <ul style="list-style-type: none"> – Use the INLINE facility to interface with the PERUSE utility
Onsite-Delivery Equipment Requirements	<ul style="list-style-type: none"> • NonStop operating system D30 or later for NonStop K-series servers, or any Gxx version for NonStop S-series servers 	<ul style="list-style-type: none"> • Pathway/TS (version D30 or later), NonStop TMF

Learn more at
hpe.com/ww/learnNonStop

Follow us:

