TAL/pTAL Programming U4198S

This course teaches a basic understanding of the Transaction Application Language (TAL) and its usage. Through a series of exercises and labs, students will gain sufficient understanding of syntax and operations to develop and maintain TAL and pTAL programs.

HPE course number	U4198S
Course length	4 days
Delivery mode	ILT/VILT
View schedule, local pricing, and register	**View now**
View related courses	**View now**

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

Systems programmers or maintainers who want to become proficient TAL or pTAL programmers.

Benefits to you

- Introduction to TAL/pTAL
- Program organization and general syntax
- Simple data types and arrays
- Data transfer, program control, and data scan statements
- Operators and expressions
- Pointers and addressing
- Procedures and subprocedures
- The Common Run-Time Environment; Interfacing C and TAL
- Building native mode programs
- Debugging with the Inspect tool

Prerequisites

- Concepts and Facilities course
  Recommended: At least six months of programming experience, preferably with C, PASCAL, or other block-structured procedure-based languages. Alternatively, one year of COBOL 85 programming experience

* Realize Technology Value with Training, IDC Infographic 2037, Sponsored by HPE, October 2017

Detailed course outline

Module 1: Introduction to TAL/pTAL	• Introduce the Tandem Transaction Application Language (TAL and pTAL) • Describe TAL/pTAL programming in the Guardian environment • Describe tools to assist the developer in writing and debugging TAL and pTAL programs
Module 2: Program organization and general syntax	• Discuss basic syntax and organization of TAL programs • Discuss identifiers, begin-end blocks, constants, operators, variable declarations, and procedure declarations • Describe compiler (both TAL and pTAL) directives that control listings • Describe sourcing from another file • Explain the environment set up for the Inspect product and compile for syntax only • Demonstrate basic Inspect software commands
Module 3: Simple data types and arrays program flow statements	• Describe how data is stored and what facilities TAL provides to access data • Discuss correct data types for various purposes, definitions, literals, labels, and data equivalencing • Describe types of program flow statements and their usage • Describe statements for CASE, IF, program control, bit extraction, and manipulation • Lab Exercise • Use constructs such as the CASE statement and WHILE loop
Module 4: Terminal I/O	• Discuss how to perform terminal I/O in programs • Discuss error handling and data conversion • Lab Exercise • Perform terminal I/O to prompt for data and implement data conversion
Module 5: Operators and expressions	• Describe types of operators and expressions • Discuss arithmetic expressions and conditional expressions • Discuss special expressions such as: assignment, CASE, IF, and group comparison expressions
Module 6: Pointers and addressing	• Describe direct and indirect data access, pointers, and structures • Discuss additional pTAL pointer data types • Discuss data transfer and scan statements to illustrate use of pointers and structures • Describe pTAL caveats • Lab Exercise • Use pointers, addressing, and data movement
Module 7: Procedures and subprocedures	• Describe procedures and subprocedures, with or without parameter passing • Discuss procedure declaration, subprocedure declaration, and procedure calls and returns • Explain the use of formal and actual parameters in procedure calls and returns • Discuss TAL and pTAL considerations for Guardian procedures that are obsolete for pTAL procedures • Lab Exercise • Code procedures, with and without parameters • Invoke procedures, with and without parameters
Module 8: Interfacing C and TAL	• Describe the Common Run-Time Environment (CRE) • Demonstrate how to interface C and TAL and memory model considerations • Discuss the use of development tools such as Bind, nld, and noft • Lab Exercise • Build runnable units from separately compiled C and/or TAL objects

Next steps

Guardian API Programming.

Learn more at
hpe.com/ww/learnnonstop

Follow us:

Hewlett Packard
Enterprise