

# DFIR450-EnScript Programming

## Syllabus



### Training facilities

**Los Angeles, CA (Pasadena, CA)**  
1055 East Colorado Boulevard  
Suite 400  
Pasadena, CA 91106-2375

**Washington, DC  
(Gaithersburg, MD)**  
9711 Washingtonian Blvd  
6th floor, Room 601 (Paris Room)  
Gaithersburg, MD 20878

**London, UK (Reading)**  
420 Thames Valley Park Drive  
Earley, Reading  
Berkshire RG6 1PT

**Munich, Germany (Grasbrunn)**  
Werner-von-Siemens-Ring 20  
85630 Grasbrunn/München  
Germany

For a complete listing of locations, including Authorized Training Partners around the world, please visit [opentext.com/learning-services/learning-paths](https://opentext.com/learning-services/learning-paths).

### Day 1

Day one begins with a brief session addressing any questions generated from reviewing the EnScript Fundamentals material, which the students should have done prior to class.

After that, students will learn about creating EnScript applications to complete common case examination functions. Students will complete a practical exercise, allowing them to use their newly learned techniques.

#### Day 1 covers:

- Working with case data.
- Understanding how to iterate open cases and the entries they contain
- Learning how to use the EnScript language to identify highlighted data within the current case
- Working with file data
- Learning to read data associated with entries and records in the current case
- Reading string data
- Writing data to the local system
- Opening files
- Creating folders
- Writing string data to files

- Bookmarking
- Creating bookmark folders
- Working with different bookmark types including note bookmarks, item bookmarks, text bookmarks, decode bookmarks and data bookmarks
- Bookmarking XML Data

## Day 2

Day two begins with instruction on using EnScript programs to aid in searching case data. Next, students will learn how to search through compound files. The final lesson of the day covers how to use logical evidence files to store entry data from a case. Practical exercises will be administered throughout the day so students can test their new skills.

### Day 2 covers:

- Learning to create and use EnScript programs to instigate searching and bookmarking operations
- Working with compound files
- Mounting Microsoft® Word docx files as collections of zipped XML streams
- Searching for XML values with a specific name and path
- Bookmarking XML data of note
- Working with logical evidence files
- Opening logical evidence files
- Understanding logical evidence file options
- Storing data in logical evidence files using a particular path

## Day 3

Day three begins with instruction on examining Windows® Registry data and continues with a lesson on how EnScript programming can provide access to SQLite database content. During the final lesson of the day, students will learn how to create custom lists through EnScript programming. The day concludes with a practical exercise.

### Day 3 covers:

- Examining Windows Registry data
- Accessing system-wide Registry hive files (SAM, Security, Software, System, etc.)
- Reading user-specific Registry data
- Finding Registry data given a specific name/path
- Working with SQLite database files.
- Querying SQLite data
- Writing SQLite data to a tab-delimited spreadsheet
- Creating and bookmarking custom lists
- Understanding and inheriting NodeClass and NameListClass to create custom lists
- Creating custom HandlerClass objects to bookmark custom-list data

## Learn more

[Contact us directly](#) ›

## Day 4

Day four begins with instruction on how to create result sets, which allow the examiner to identify items of potential interest, regardless of whether those items are records or entries.

The final lesson of the course provides the students with an understanding of the purpose and use of dialog boxes within EnScript programming.

### Day 4 covers:

- Creating result sets
- Building dialog boxes
- Understanding dialog box basics
- Multi-tabbed dialog boxes and Wizard dialog boxes
- Understanding virtual methods/functions
- Using different dialog controls
- Buttons
- String-edit controls, path-edit controls and enumerated-value edit controls
- Handling events
- Validating user input