

Undocumented Matlab

unbelievable features; unbelievable quality; unbelievable cost effectiveness; unbelievable service

OBJECT ORIENTED MATLAB PROGRAMMING

WEBINAR OUTLINE

1. Introduction to MATLAB OOP

- a. comparing paradigms: OOP vs. procedural programming
- b. importance of OOP for development and maintainability
- c. MATLAB OOP is gaining momentum; MATLAB code is increasingly using OOP
- d. benefits and drawbacks of MATLAB OOP
- e. MATLAB OOP's historic evolution and future outlook



2. Programming Object-Oriented MATLAB

- a. components of MATLAB OOP: packages, classes, properties, methods, and events
- b. the format of a Matlab class
- c. offline & run-time inspection of class components
- d. constructors and destructors
- e. handle vs. value classes
- f. class folders, files and attributes
- g. handling accessibility attributes
- h. bypassing property restrictions (carefully!)
- i. dependent (on-the-fly) properties
- j. specifying property types
- k. property setter and getter methods

Summary

You will learn how to:

- create high-quality, maintainable Matlab programs
- use the modern object-oriented programming paradigm, replacing "spaghetti code"

3. Advanced Object-Oriented MATLAB

- a. copying objects (deep/shallow copy)
- b. class inheritance
- c. overloading class components
- d. using static classes
- e. object pooling
- f. the singleton design pattern
- g. raising and listening to events
- h. using enumerated values
- i. runtime performance aspects

Target audience

Matlab users of any level, from beginners to advanced, who wish to improve their program's maintainability and usability.

Familiarity with the Matlab environment and programming is assumed.