Introduction to Object-Oriented MATLAB

Antti Löytynoja, Application Engineer
Demonstration: Procedural Programming

- Simulate a bank account using a structure and some functions
  - Structures (data)
  - Functions (actions)
Procedural Programming

- Easy to learn
- Minimal planning

- No formal relationship between data and functions
- Every detail is exposed
Why Object-Oriented Programming?

- Build robust applications
  - Build a relationship between data and actions
  - Control data and actions

- Design building blocks for developing applications
  - Eases code reuse
  - Enhance collaboration
Progression of Programming Techniques

Level of Abstraction / Sophistication

Application Complexity

Data

value

variable

structure

class

Actions

function

script

command line
Object-Oriented Terminology

- **Class**
  - Blueprint of an idea
  - *Properties* (data)
  - *Methods* (actions)

- **Object**
  - Specific example of a *class*
  - *Instance*
  - Any number of objects of a class can exist
Demonstration: Building a Simple Class

- Describe a bank account using a class
  - Properties (data)
  - Methods (actions)

- Use the object like a structure
Objects

- Easy to create
- Manage their own data
- Interchangeable with a structure
  - No other code changes required
  - Properties behave similar to field names
  - Can’t add fields arbitrarily

```plaintext
%% Create an object
account = Account

%% Set properties
account.balance = 1000;
account.interest = 0.01;
account.number = 1234;

%% add accumulation function inside
account = account.accumulate();
```
Objects with Methods

- Have immediate access to their own data *(properties)*

- Allow you to overload existing functions

- Allow you to perform custom actions at creation and deletion *(constructor and destructor)*
Encapsulation

Account

- Number
- Credit history
- Accumulate
- Address
- etc.
- Withdraw
- Name
- VIP status
- Deposit
- Deactivate
Encapsulation

- Separates the interface from the implementation
- Simplifies object use
- Becomes a building block
Demonstration: Applying Attributes

- Control access
  - Access = public
  - Access = protected

- Restrict modification
  - Constant
  - Dependent
Using a Class as a Building Block

All accounts

All private accounts

Normal account

Credit account
Demonstration: Creating a Credit Account

- Define a new class `CreditAccount`

- *Inherit* from the existing class `Account` to reuse code

- Add two additional properties, overload withdraw -function

- Use the credit account
Inheritance

- **Subclass** substitutes for the **superclass**
- Allows re-envisioning and re-implementing the **superclass**
- Builds on proven code
- Allows inheriting from the base MATLAB classes
Summary

- O-O MATLAB is a sophisticated way of building MATLAB apps
  - Follows the same principles as any other O-O language

- Builds robust code (encapsulation)

- Allows you to reuse code (inheritance)

- Good when building larger apps and if many people contributing to/using code
Additional material

Object-Oriented Programming in MATLAB

Develop complex technical computing applications

Object-oriented programming is a formal programming approach that combines data and associated actions (methods) into logical structures (objects). This approach improves the ability to manage software complexity—particularly important when developing and maintaining large applications and data structures.

The object-oriented programming capabilities of the MATLAB® language enable you to develop complex technical computing applications faster than with other languages, such as C++, C#, and Java™. You can define classes and apply standard object-oriented design patterns in MATLAB that enable code reuse, inheritance, encapsulation, and reference behavior without engaging in the low-level housekeeping tasks required by other languages.

Object-oriented programming in MATLAB involves using:

- Class definition files, enabling definition of properties, methods, and events
- Classes with reference behavior, aiding the creation of data structures such as linked lists
- Events and listeners, allowing the monitoring of object property changes and actions

Examples and How To

- Introduction to Object-Oriented Programming in MATLAB (MATLAB Digest article)
- Inside MATLAB Objects (MATLAB Digest article)
- Sample code comparisons - MATLAB and C++ (MATLAB Central)
- Sample code comparisons - MATLAB, C++, Java, Python, and Ruby (MATLAB Central)
- Object-Oriented Programming in MATLAB 48:12 (Webinar)

http://se.mathworks.com/discovery/object-oriented-programming.html
Thank You!