

Introducing Object-Oriented Programming (OOP) in C++



Goals

- By the end of this lecture, you should ...
- Understand the three pillars of Object-Oriented Programming: Inheritance, Encapsulation and Polymorphism.
- Understand what an object is.
- Understand object attributes, methods and events.
- Understand how programmers use APIs.



75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

Programming Languages

- Programming languages allow programmers to code software.
- The three major families of languages are:
 - Machine languages
 - Assembly languages
 - High-Level languages

01:17 09:06

LERNEN
HYPER ACADEMY

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

Machine Languages

- Comprised of 1s and 0s
- The “native” language of a computer
- Difficult to program – one misplaced 1 or 0 will cause the program to fail.
- Example of code:

1110100010101	111010101110
10111010110100	10100011110111

01:39 09:06

LERNEN
HYPER ACADEMY

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

Assembly Languages

- Assembly languages are a step towards easier programming.
- Assembly languages are comprised of a set of elemental commands which are tied to a specific processor.
- Assembly language code needs to be translated to machine language before the computer processes it.
- Example:
`ADD 1001010, 1011010`

02:14 09:06

85%

LERNEN
HYPADREMY

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

High-Level Languages

- High-level languages represent a giant leap towards easier programming.
- The syntax of HL languages is similar to English.
- Historically, we divide HL languages into two groups:
 - Procedural languages
 - Object-Oriented languages (OOP)

02:48 09:06

85%

LERNEN
HYPADREMY

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

Procedural Languages

- Early high-level languages are typically called procedural languages.
- Procedural languages are characterized by sequential sets of linear commands. The focus of such languages is on *structure*.
- Examples include C, COBOL, Fortran, LISP, Perl, HTML, VBScript

03:43 09:06

85%

LERNEN
HUTEN
ADEMY

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

Object-Oriented Languages

- Most object-oriented languages are high-level languages.
- The focus of OOP languages is not on structure, but on *modeling data*.
- Programmers code using “blueprints” of data models called *classes*.
- Examples of OOP languages include C++, Visual Basic.NET and Java.

04:16 09:06

85%

LERNEN
HUTEN
ADEMY

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

Object Oriented Programming


- **Object** – Unique programming entity that has *methods*, has *attributes* and can react to *events*.
- **Method** – Things which an object can do; the “verbs” of objects. In code, usually can be identified by an “action” word -- *Hide, Show*

05:06

09:06

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help




Object Oriented Programming

- **Attribute** – Things which describe an object; the “adjectives” of objects. In code, usually can be identified by a “descriptive” word – *Enabled, BackColor*
- **Events** – Forces external to an object to which that object can react. In code, usually attached to an event procedure

05:42

09:06



75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help


Object Oriented Programming

- **Class** – Provides a way to create new objects based on a “meta-definition” of an object (Example: The automobile **class**)
- **Constructors** – Special methods used to create new instances of a class (Example: A Honda Civic is an **instance** of the automobile **class**.)

06:10 09:06

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help




OOP - Encapsulation

- Incorporation into a class of data & operations in one package
- Data can only be accessed through that package
- “Information Hiding”

06:39 09:06

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help



75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

OOP - Inheritance

- Allows programmers to create new classes based on an existing class
- Methods and attributes from the parent class are inherited by the newly-created class
- New methods and attributes can be created in the new class, but don't affect the parent class's definition


06:59

09:06

80%

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help



OOP - Polymorphism

- Creating methods which describe the way to do some general function
(Example: The "drive" method in the automobile class)
- Polymorphic methods can adapt to specific types of objects.


07:17

09:06

80%

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help



75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

Classes and Objects

- A **class** is a data type that allows programmers to create objects. A class provides a definition for an object, describing an object's attributes (data) and methods (operations).
- An object is an *instance* of a class. With one class, you can have as many objects as required.
- This is analogous to a variable and a data type, the class is the data type and the object is the variable.

LERNEN
HUB ACADEMY

07:33 09:06

75 Understand basics of OOP.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

What is an API?

- API stands for Application Programming Interface
- Allows programmers to extend the current language to included customized components
- Most modern languages incorporate APIs

LERNEN
HUB ACADEMY

08:05 09:06

