



---

# IMGD 3000 - Technical Game Development I: Object-Oriented Development

by  
Robert W. Lindeman  
gogo@wpi.edu

---



## Design Patterns

---

- The Interface
- The Singleton
- Object Factory

## The Interface

- In Java, we can specify a contract for a given class using *interfaces*
- C++ has no formal concept of this
- We can implement them using a class with:
  - No data members
  - All purely virtual functions
- Example from C4
  - Colliders

## The Singleton

- Sometimes we only want a single instance to exist for a given object
  - Game
  - Audio manager
  - Game-asset data manager
  - Different interface elements
- We can do this in C++ using a class with static member
- Examples from C4
  - TheWorld, TheGame, TheAudioManager
- Can be done generically using templates

## Object Factories

---

- ❑ There are many types of objects that are similar
- ❑ We don't know at compile time what they are
- ❑ Save and load should remain flexible
- ❑ How can we efficiently access them?
- ❑ Object Factories allow us to solve this type of problem
- ❑ Takes advantage of ***polymorphism***