



CLASS DEVELOPMENT

Analysis, Design, and Implementation/Programming



THE TWO-HAT TECHNIQUE

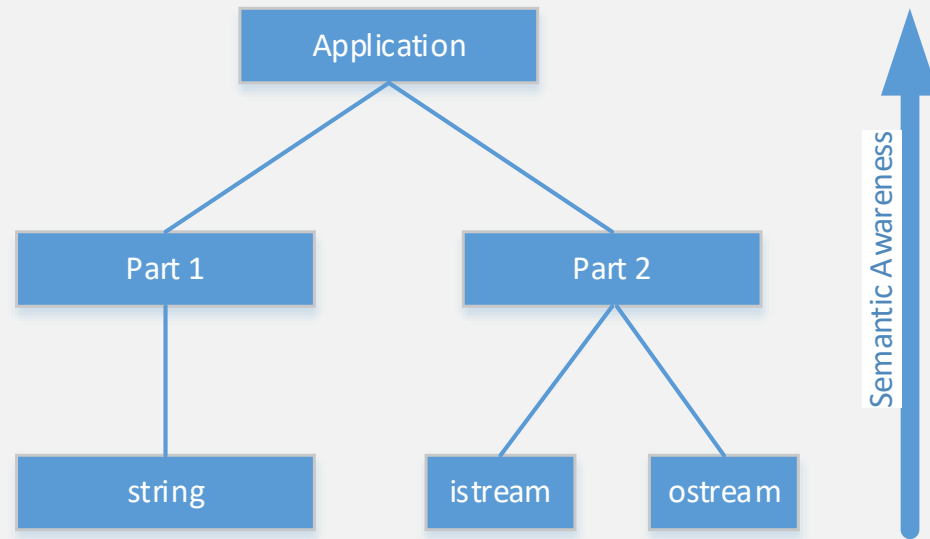
SERVER OR SUPPLIER



CLIENT OR USER

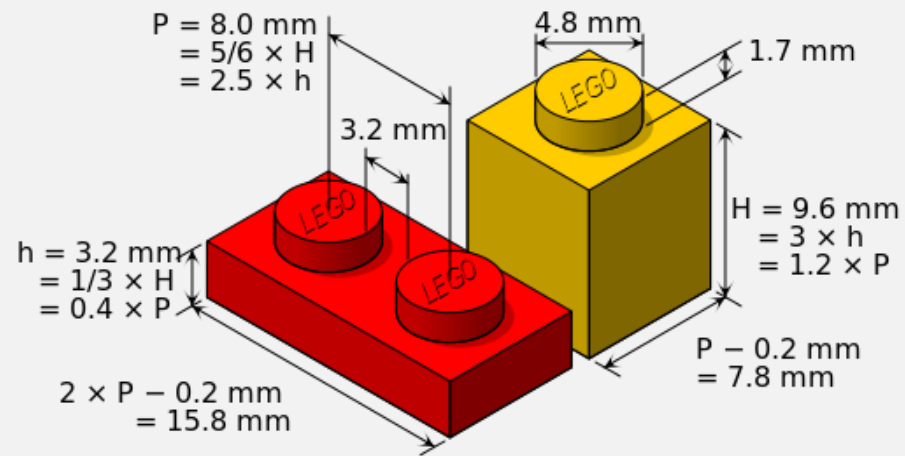


WHAT DO CLASSES “KNOW”

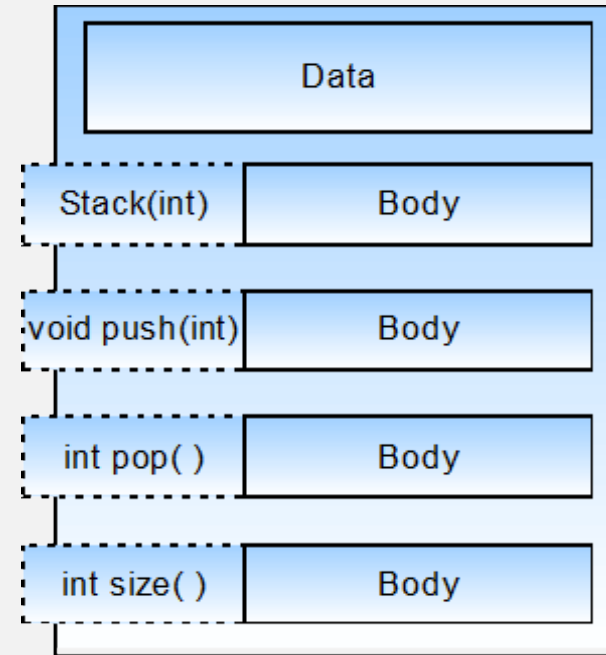


- The semantic content of a class may be abstractly represented by a hierarchy
- Classes near the top are problem-oriented
- Class near the bottom are service-oriented
 - Designed as a “shopping list”
 - May provide more services than needed in a given program – for example the string class

THE PUBLIC INTERFACE



by Cmglee - via Wikimedia Commons





DESIGN CONSIDERATIONS

ATTRIBUTES / VARIABLES

- Are typically private
- Be able to articulate a good reason before making them non-private
- Consider access functions instead
 - Provides stronger encapsulation
 - Separates the interface from the implementation
 - Non-private attributes are not easily removed or modified in future version

OPERATIONS / FUNCTIONS

- Are typically public
- Make “helper” functions private to prevent them from becoming part of the public interface
- The function implementation (i.e., the body) can change while the header or signature remains unchanged