Inheritance Encyclopedia Article

Inheritance

The following sections of this BookRags Literature Study Guide is offprint from Gale's For Students Series: Presenting Analysis, Context, and Criticism on Commonly Studied Works: Introduction, Author Biography, Plot Summary, Characters, Themes, Style, Historical Context, Critical Overview, Criticism and Critical Essays, Media Adaptations, Topics for Further Study, Compare & Contrast, What Do I Read Next?, For Further Study, and Sources.

(c)1998-2002; (c)2002 by Gale. Gale is an imprint of The Gale Group, Inc., a division of Thomson Learning, Inc. Gale and Design and Thomson Learning are trademarks used herein under license.

The following sections, if they exist, are offprint from Beacham's Encyclopedia of Popular Fiction: "Social Concerns", "Thematic Overview", "Techniques", "Literary Precedents", "Key Questions", "Related Titles", "Adaptations", "Related Web Sites". (c)1994-2005, by Walton Beacham.

The following sections, if they exist, are offprint from Beacham's Guide to Literature for Young Adults: "About the Author", "Overview", "Setting", "Literary Qualities", "Social Sensitivity", "Topics for Discussion", "Ideas for Reports and Papers". (c)1994-2005, by Walton Beacham.

All other sections in this Literature Study Guide are owned and copyrighted by BookRags, Inc.

Contents

Inheritance Encyclopedia Article.....1

Contents.....2

Inheritance.....3

Inheritance

In **object-oriented programming**, inheritance is the ability to derive new classes from existing classes. The existing **class** from which the subclass is derived is called the **base class** (sometimes called the superclass); the **derived class** is called either, simply, the derived class, or it's called the subclass. A subclass inherits from the base class, meaning that it has **access** to the variables and methods of the base class without having to redefine them. A subclass can also define its own variables and methods.

Any time a new **variable** or **method** is declared in the base class, it becomes immediately available to the subclass via inheritance. The same holds true for changes made to the base class. Changes to the base class are immediately available to the subclass via inheritance.

A class can inherit from multiple base classes. This is referred to as **multiple inheritance**. In multiple inheritance, the new subclass derives all the variables and methods offered by each of the base classes. This can cause **conflict** if more than one of the base classes has the same variable or method. These inheritance clashes are resolved by the subclass either redefining the conflicting variable or method for itself or by specifying which inheritance is preferred.

Inheritance speeds up the development process. It organizes the structure of programs in a modular fashion and increases reuse. In addition it shortens testing time by providing inherent validity to the subclass. If a method works in the base class and is inherited by a subclass, it will also work in the subclass.