

[illegible]

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

The Booch method is a technique used in software engineering. It is an object modeling language and methodology that was widely used in object-oriented analysis and design. It was developed by Booch while at Rational Software.

Booch developed the Booch method of software development, which he presents in his 1991/94 book, Object Oriented Analysis and Design With Applications. He advises adding more classes to simplify complex code.

■ **Subtyping** is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>

■ In a number of object-oriented languages, subtyping is called interface inheritance, with inheritance referred to as implementation inheritance.

■ **Subtyping** is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>

■ **Subtyping** is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>

■ **Subtyping** is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>

■ **Subtyping** is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>

wiki > Subtyping should not be confused with the notion of (class or object) inheritance from object-oriented languages =>

> subtyping is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>

> In a number of object-oriented languages, subtyping is called interface inheritance, with inheritance referred to as implementation inheritance.

■ **Subtyping** is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>

People (including me) have made the mistake that this is about inheritance.

■ **Subtyping** is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>

■ **Subtyping** is a relation between types (interfaces in object-oriented parlance) whereas inheritance is a relation between implementations stemming from a language feature that allows new objects to be created from existing ones. =>