

Abstract classes

(OOP Core Concept 4 - Abstraction)

Learning objectives

- Abstract classes and methods
- Final classes, methods and variables

Abstract classes

- ▶ Abstract classes cannot be instantiated
- ▶ They can be useful as a Superclass with functionality that can be inherited
- ▶ They can also be useful as general object references
- ▶ They are created by adding the keyword `abstract` before the class

Why use abstract classes?

- ▶ It is a limitation of the possibilities of a class
- ▶ But sometimes a class is not meant to be used to create objects out of it
- ▶ It's a way to make it work only as a Superclass

Demo 1 - Abstract classes

- ▶ Create an abstract class
- ▶ Using an abstract class

Abstract methods

- ▶ Abstract classes can have abstract methods (other classes cannot)
- ▶ Abstract methods has no implementation (no body)
- ▶ So abstract methods does not provide a reusable implementation, it forces a non abstract subclass to override it with an implementation

Final class

- ▶ Final classes cannot be extended, they are used to be instantiated
- ▶ Final classes are kind of the opposite of abstract classes in that way

Final methods

- ▶ Final methods cannot be overridden, they are inherited and cannot be changed by the subclass
- ▶ Final methods are kind of the opposite of abstract methods in that way

Final variable

- ▶ Final variables cannot change after it has been initialized with a value
- ▶ They cannot be assigned a new value if they are initialized

Demo 2 - Abstract and final methods

- Abstract methods
- Final methods

Exercise 1 - Abstract classes

- ▶ Continue on the exercise from the Polymorphism module
- ▶ Make the Product class abstract (because no product is only a Product), does anything else need to change?
- ▶ Then make the description method in Product abstract (because there is not enough information in the Product, let's instead force the subclasses to provide their own implementations of this method)
- ▶ Will the project still run? Did anything need to change?

Learning objectives

- Abstract classes and methods
- Final classes, methods and variables