# JAVA ABSTRACT CLASSES AND METHODS

Efthimios Alepis

# ABSTRACT CLASS

*abstract class ClassA*

An abstract class is a class that is declared abstract

An abstract class may or may not include abstract methods

Abstract classes cannot be instantiated

Abstract classes can be subclassed

If a class includes abstract methods, then the class itself must be declared abstract

When an abstract class is subclassed, the subclass usually provides implementations for all of the abstract methods in its parent class. However, if it does not, then the subclass must also be declared abstract

# ABSTRACT METHOD

An abstract method is a method that is declared without an implementation (Where have you seen this?..)

abstract void dosomething();

# SUBCLASSES OF ABSTRACT CLASS

What happens when an abstract class is subclassed?

- Either the subclass provides implementations for all abstract methods of its parent class
- Or the subclass must also be declared as abstract…

# ABSTRACT CLASS VS INTERFACE

Similarities:

- Both cannot be instantiated
- Both can contain methods without implementation

Dissimilarities:

- Only Abstract classes can contain concrete methods
- An abstract class can extend only one abstract (or non abstract) class. Interfaces can extend a number of interfaces
- In an abstract class keyword "abstract" is mandatory to declare a method as abstract. In an interface it is optional and is almost never used
- Abstract classes can have public, protected and private concrete methods. Interfaces have only public methods