Inheritance and Polymorphism II

Session 5

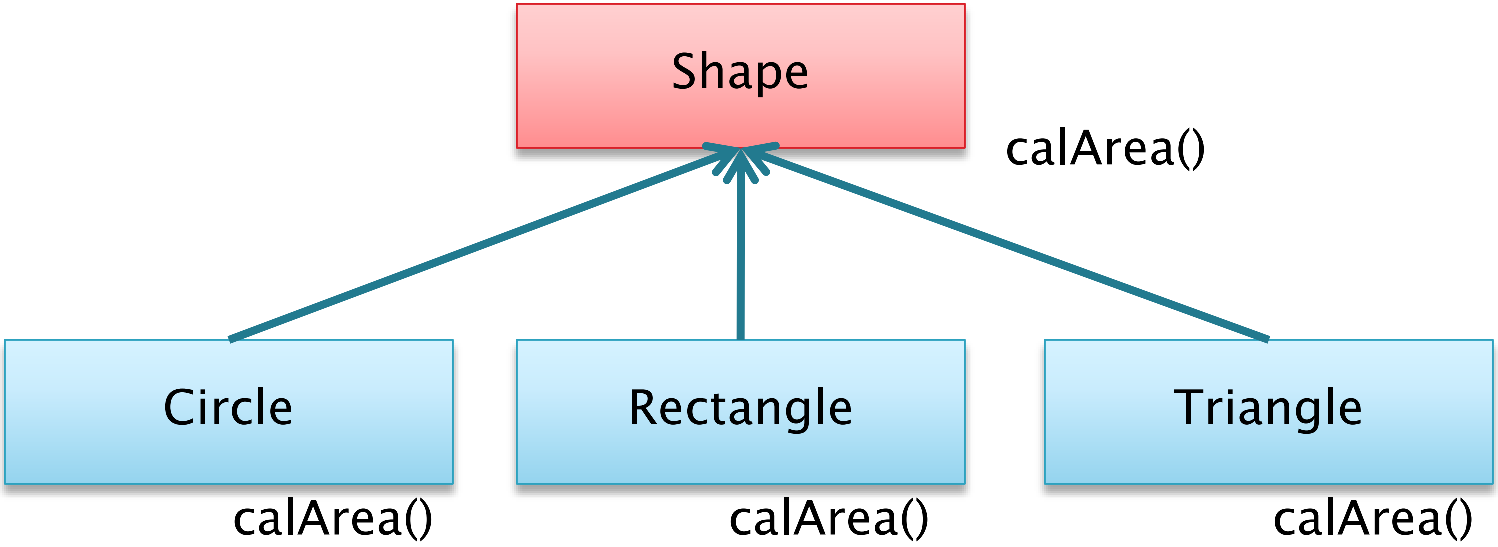
Contents

- Difference between concrete class and abstract class
- Need of abstract method
- Abstract class and method.
- Interface
- Need of interface
- Implementation of interface

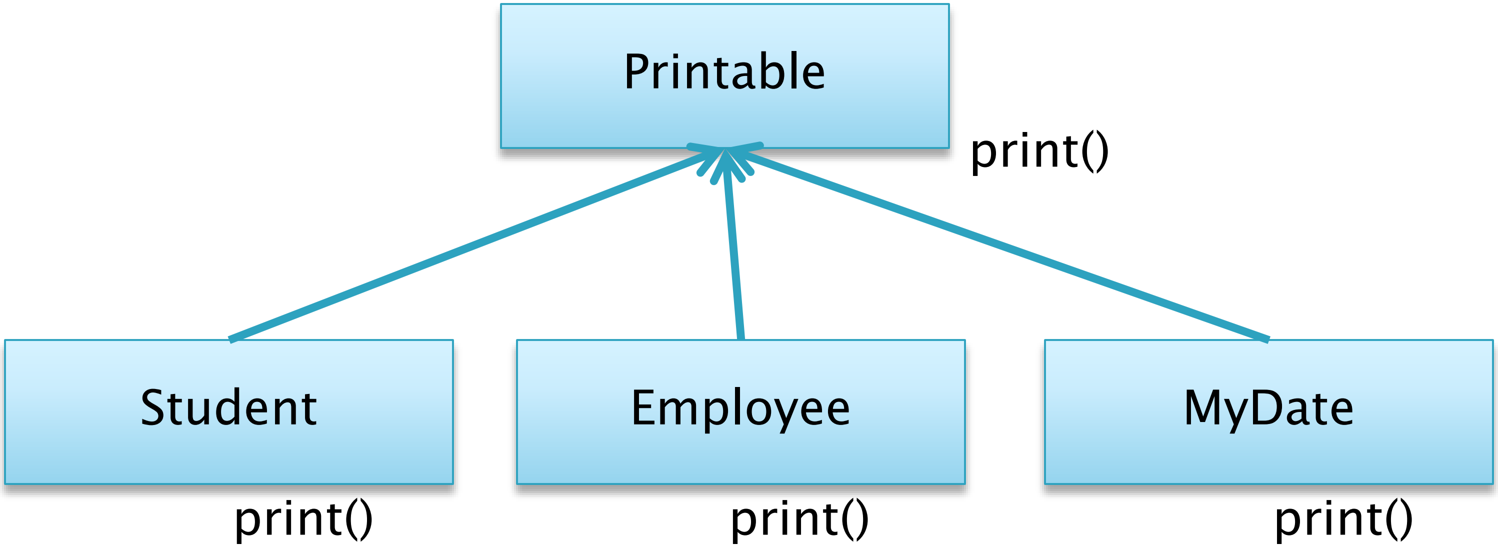
Concrete and Abstract class

- Concrete class defines the functionality of an object.
- This class can be instantiated.
- Abstract class
  - A class that contains non-implemented, generic methods.
  - Subsequent derived class need to provide implementation of such methods.
  - Can’t be instantiated. / Object cannot be created.
  - It may contain abstract as well as non abstract methods.
    - Abstract methods do not have implementation.

Need of Abstract class.



Interface


Printable
print()
Student
Employee
MyDate
print()
print()
print()

Interface

- Interface allows to have common methods in the classes which are not in the hierarchy.

Interface

- An interface defines a contract between a provider and consumer. (where provider is interface and class is consumer).
  - Interface enables common design in classes which are not in hierarchy.
- Interface defines a standard set of methods, events, properties
- A class implementing interface has to provide the functionality of methods declared in interface.

Features of Interface

- Interface is collection of constants and abstract methods.
  - All methods of interface are abstract by default.
- A subclass can have only one super class in java but a class can implement any number of interfaces.
  - This feature gives **look and feel** of **Multiple inheritance.**

Abstract Class vs. Interface

	Abstract class	Interface
Methods	Class can have at least 1 abstract method	All methods are abstract by default
Best suited for	Classes in hierarchy	Contract based provider model
Component versioning	By updating abstract class all derived classes are automatically updated.	Interfaces are immutable.

Thank
You!