

Yasmeen Allahaleh

Professor Conrad

CIS 18A- Java

2 February 2024

### *Abstract Classes and Interfaces*

Abstract classes can serve an important role when it comes to software reusability to decrease time. When using abstract classes, its members can be used in other classes when extended, but the class itself can not be instantiated. This helps generalize members to use them across many different other classes and serves as an organizing template for object oriented programming. This helps save time and space in one's code since you don't have to redeclare members through different classes every single time. Methods of an abstract class can be reused by other subclasses. This can help reduce the number of errors in a program due to the fact that an object can not be created, so "this protects code from being used incorrectly"(Gillis). If errors do occur in the code, it would be also easier to track since the software reusability can shorten and condense the code, where it is easier to navigate. Just like abstract classes, interfaces can not instantiate an object as they are not actually classes. Both abstract classes and interfaces are useful when it comes to inheritance, but each has their perks and weaknesses. The concept of abstract and interfaces seem so similar, so knowing when to use which one is important to maximize the efficiency of your code. For abstract classes according to Zsolt Nagy from QuickStart, abstract classes are best "used for objects that are closely related" and interfaces are in better use when "providing a common functionality to unrelated classes"(Nagy). One thing to consider when using interfaces though is that all methods have to be public and fields are automatically final and static, which could be a deal breaker for many compared to abstract classes which can contain non-static fields and non-public methods. You can implement as many interfaces as you want, but can only extend one abstract class.

Works Cited:

Deitel, Paul J., and Harvey M. Deitel. *Java: How to Program*. Pearson, 2018.

“Difference between Abstract Class and Interface in Java.” *GeeksforGeeks*, 5 Apr. 2023, [www.geeksforgeeks.org/difference-between-abstract-class-and-interface-in-java/](http://www.geeksforgeeks.org/difference-between-abstract-class-and-interface-in-java/). Accessed 02 Feb. 2024.

Gillis, Alexander S. “What Is an Abstract Class?: Definition from TechTarget.” *TheServerSide.Com*, 22 Feb. 2023, [www.theserverside.com/definition/abstract-class](http://www.theserverside.com/definition/abstract-class). Accessed 02 Feb. 2024.

Nagy, Zsolt. “When & How to Use Abstract Class and Interface.” *QuickStart*, 14 Sept. 2023, [www.quickstart.com/programming-language/when-and-how-to-use-abstract-class-and-interface/](http://www.quickstart.com/programming-language/when-and-how-to-use-abstract-class-and-interface/). Accessed 02 Feb. 2024.