

My program is designed to use features of OOP such as encapsulation and polymorphism. My program encapsulates data and methods within classes such as the SimpleDate class, this hides implementation details and ensures that the data is accessed and modified only through methods that are defined within the class. I use polymorphism in the printDates method in the DateApp class as it accepts an ArrayList of SimpleDate objects but can also accept an ArrayList of objects that inherit from the SimpleDate class, this means that the method can be used with a wider range of input times, making the program more flexible as if additional information is added the printDates can be modified to accept this without needing to be rewritten. I aimed to add the best modularity I could to the program with separate classes responsible for one aspect of the data processing e.g the DateParser is only responsible for parsing the data. This makes the code easier to understand, maintain and modify. The SimpleDate class also shows a form of abstraction as the class abstracts the concept of a date as a combination of year, month and day and provides methods for dates without exposing the implementation details.