

Answer the following two questions:

- How does your design implement the four pillars of OOP (abstraction, encapsulation, inheritance and composition, and polymorphism)?

Abstraction - We used an abstract class that is not fully implemented and has hook methods that require child classes to implement

Encapsulation - We have private attributes that are indicated by the underscore and is only accessible through getters and setters.

Inheritance/Composition - Chequing and Savings inherits attributes and methods from the Abstract class

Polymorphism - We have hook methods such as `get_type` in the abstract class that performs a different function in the child classes.

- Why are your classes good abstractions (i.e., models) of the real-world entities they represent?

Our classes represent the normal attributes and methods that customers and employees of a bank interact with on a regular basis. This makes them good abstractions of real world entities. All our classes act as if they are real world entities, all the functions they have are mirrored in the real world.