
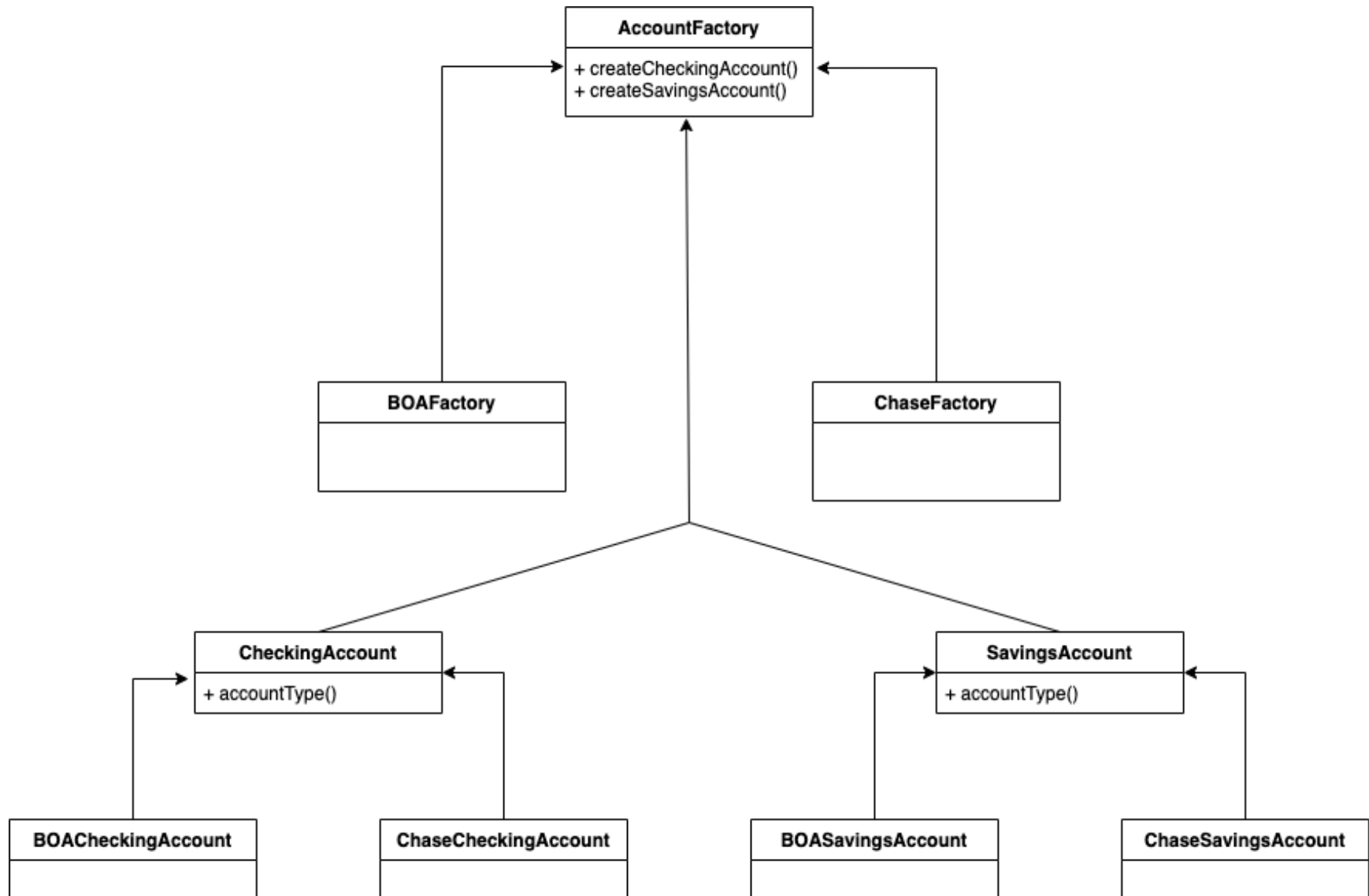


---

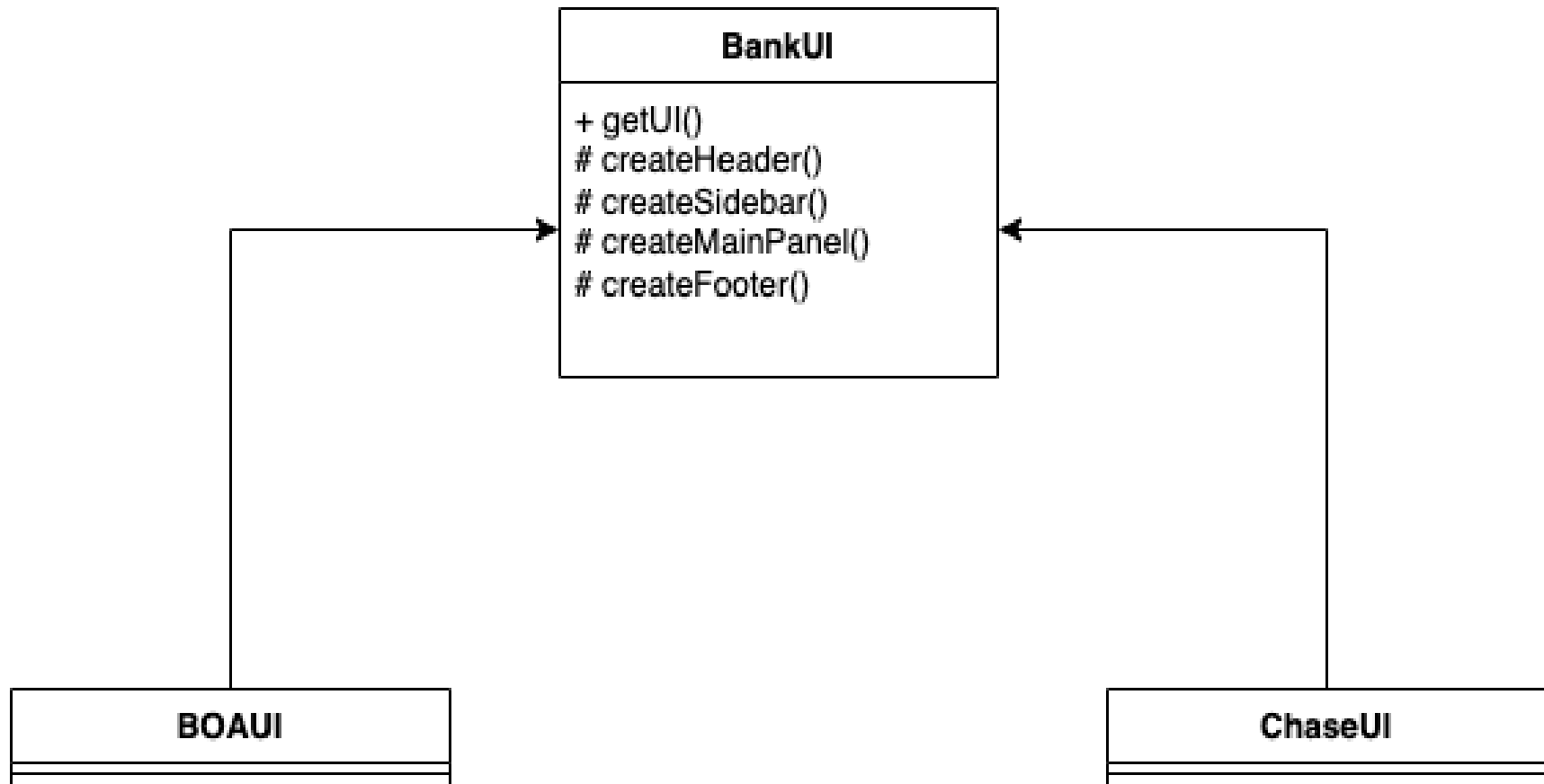
## Homework Pattern Summary

 <b>Part</b>	<b>Description</b>	<b>Design Pattern Used</b>	<b>Notes</b>
<b>Part 1</b>	Banks offering their own versions of Checking and Savings accounts	<b>Abstract Factory Pattern</b>	AccountFactory is the abstract factory; BOAFactory / ChaseFactory are concrete factories.
<b>Part 2</b>	Different banks customize the UI using the same sequence of steps	<b>Template Method Pattern</b>	BankUI is the template class; BOAUI / ChaseUI provide the concrete implementations.
<b>Part 3 – Version 1</b>	Customers (Standard, Preferred, Business) created using one central factory	<b>Simple Factory Pattern</b>	SimpleCustomerFactory is the central factory; CustomerManager uses it.
<b>Part 3 – Version 2</b>	A creator class (CustomerStore) uses a factory method to create customers	<b>Factory Method Pattern</b>	CustomerStore is the abstract creator; BankCustomerStore overrides createCustomer()

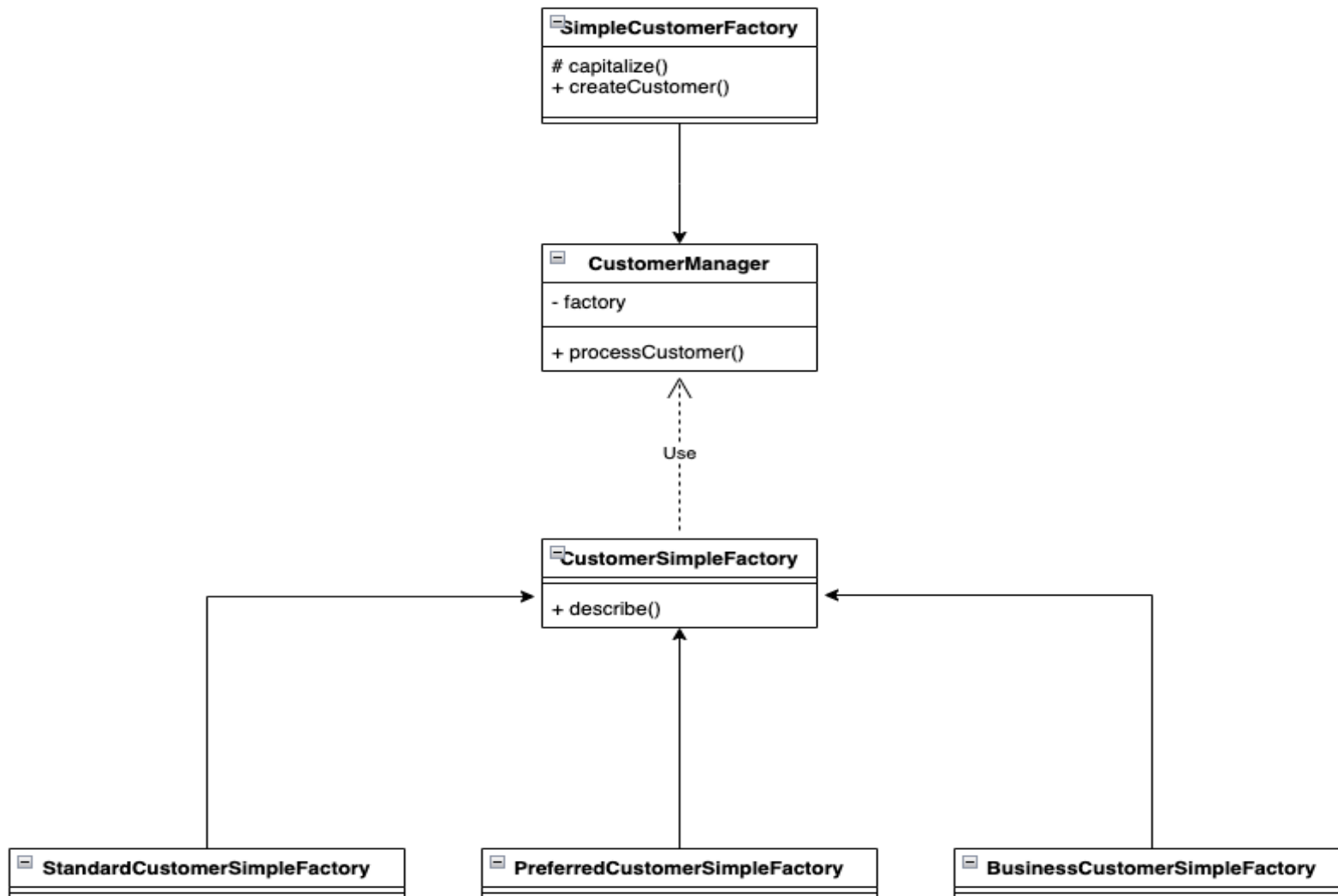
## Part 1



## Part 2



## Part 3 A



## Part 3 B

