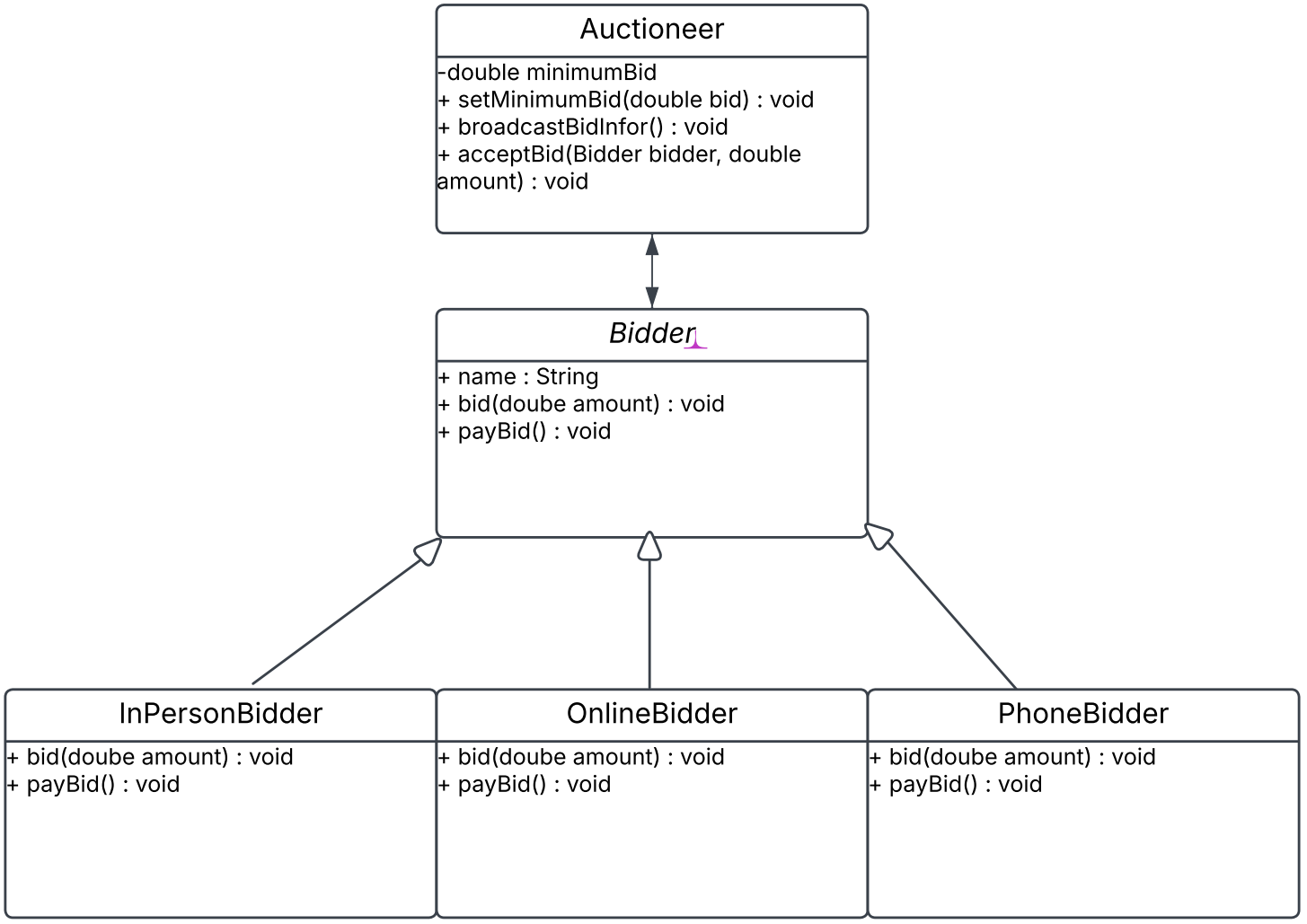
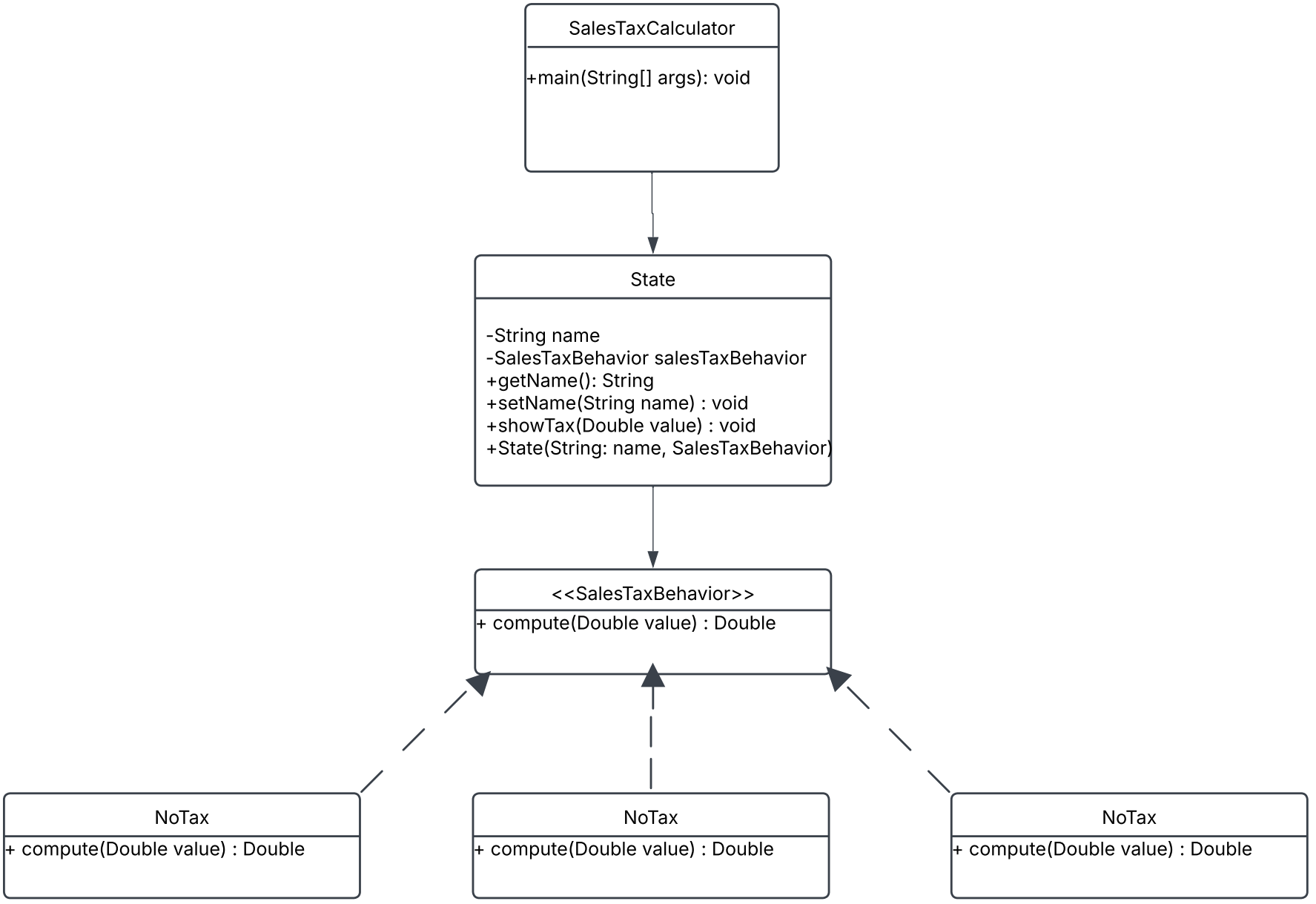

Auctioneer
-double minimumBid
+ setMinimumBid(double bid) : void
+ broadcastBidInfor() : void
+ acceptBid(Bidder bidder, double amount) : void
Bidder
+ name : String
+ bid(doube amount) : void
+ payBid() : void
InPersonBidder
+ bid(doube amount) : void
+ payBid() : void
OnlineBidder
+ bid(doube amount) : void
+ payBid() : void
PhoneBidder
+ bid(doube amount) : void
+ payBid() : void

1. **IS-A** ; the line with a hollow triangle pointing to the parent class shows inheritance, as it extends that class, in this case the display() method, therefore making this an IS-A example
2. **IS-A** ; the line with a hollow triangle pointing to the parent class shows inheritance, as it extends that class, in this case the display() method, therefore making this an IS-A example
3. **IS-A** ; the line with a hollow triangle pointing to the parent class shows inheritance, as it extends that class, in this case the display() method, therefore making this an IS-A example
4. **IS-A** ; the line with a hollow triangle pointing to the parent class shows inheritance, as it extends that class, in this case the display() method, therefore making this an IS-A example
5. **HAS-A** ; the solid arrow line represents association, as that interface contains an object of the class it associates with, in this case FlyBehavior making it a HAS-A example
6. **HAS-A** ; the solid arrow line represents association, as that class contains an object of the class it associates with, in this case QuackBehavior making it a HAS-A example
7. **IS-A** ; the line with a hollow triangle pointing to the parent class shows inheritance, as it extends that class, in this case the quack() method, therefore making this an IS-A example
8. **IS-A** ; the line with a hollow triangle pointing to the parent class shows inheritance, as it extends that class, in this case the quack() method, therefore making this an IS-A example
9. **IS-A** ; the line with a hollow triangle pointing to the parent class shows inheritance, as it extends that class, in this case the quack() method, therefore making this an IS-A example


SalesTaxCalculator
+main(String[] args): void
State
-String name
-SalesTaxBehavior salesTaxBehavior
+getName(): String
+setName(String name) : void
+showTax(Double value) : void
+State(String: name, SalesTaxBehavior)
<<SalesTaxBehavior>>
+ compute(Double value) : Double
NoTax
+ compute(Double value) : Double
NoTax
+ compute(Double value) : Double
NoTax
+ compute(Double value) : Double