

StockControllerCommandImplTest	
startMockTest()	void
createFixedEqualMockTesting()	void
createFixedSeparateMockTesting()	void
createFixedSeparateOnGoingMockTesting()	void
createFixedSeparateOnGoingMockTesting2()	void
quitWhenCreating()	void
quitWhenBuying()	void
quitWhenBuying2()	void
quitWhenBuying3()	void
quitWhenBuyingByAmount()	void
quitWhenBuyingByAmount2()	void
quitWhenBuyingByAmount3()	void
quitDetermineCost()	void
quitDetermineValue()	void
quitDetermineValue2()	void
quitDetermineValue3()	void
quitTesting()	void
determineCommissionFee()	void
determineCommissionFeeQuit()	void
buyByPercentage()	void
buyByPercentageQuit()	void
buyByPercentageQuit2()	void
buyByPercentageQuit3()	void
buyByPercentageQuit4()	void
savePortfolio()	void
quitWhenSavePortfolio()	void
quitWhenSavePortfolio2()	void
savePercentage()	void
quitWhenSavePercentage()	void
quitWhenSavePercentage2()	void
loadPortfolio()	void
loadPortfolio2()	void
loadPercentage()	void
loadPercentage2()	void
realCreateFixedTesting()	void
realTesting()	void

StockController	
setView(IView)	void
processCommand(String)	String

StockControllerImpl	
model	StockModel
view	IView
mainView	IView
createView	IView
getAllStateView	IView
getStateView	IView
determineCostView	IView
determineFeeView	IView
determineValueView	IView
buyView	IView
buyPercentageView	IView
buyAmountView	IView
createFixedView	IView
createPercentageView	IView
savePortfolioView	IView
savePercentageView	IView
loadPortfolioView	IView
loadPercentageView	IView
StockControllerImpl(StockModel, IView, IView, IView, IView, IView, IView, IView, IView, IView, IView, IView, IView, IView, IView)	
setView(IView)	void
configureKeyListener()	void
configureButtonListener()	void
processCommand(String)	String
getBuilder()	StockControllerBuilderImpl

StockControllerCommand	
start(StockModel)	void

StockControllerCommandImpl	
in	Readable
out	Appendable
StockControllerCommandImpl(Readable, Appendable)	
output(String)	void
start(StockModel)	void

StockControllerImplTest	
processCommand()	void

Package commands

Package utility
