Name: Exercise_5: Observer Pattern: Standard Library Container and Template Exception Class

Package: Exercise_5
Version: 1.0
Author: Thomas Batt

«forward declaration» ${\bf Clock_Observer_Application} :: {\bf cClockObserverApplicationBuilder}$ Common::icSubject Common::icObserver mClock: cClock + ~icSubject() ~icObserver() mptrAnalogClock: std::shared_ptr<cAnalogClock> attach(ptrObserver: std::shared_ptr<icObserver>): void update(): void mptrDigitalClock: std::shared ptr<cDigitalClock> detach(ptrObserver: std::shared ptr<icObserver>): void show(): void {query} mptrAnalogDigitalClock: std::shared_ptr<cAnalogDigitalClock> notify(): void mptrClockArray: std::shared ptr<tcClockArray<10>> mptrClockContainer: std::shared ptr<cClockContainer> cClockObserverApplicationBuilder() ~cClockObserverApplicationBuilder() execute(): void -mClock Clock::cClock buildRelations(): void -mptrClockArray ArrayLength: uint32_t Clock_Source::cClockContainer Clock_Source::tcClockArray mObserverPtrList: list<std::shared ptr<icObserver>> cClockContainer() ~cClockContainer() attach(ptrObserver: std::shared_ptr<icObserver>): void < T->std::shared_ptr<icObserver>, ArrayLength->ArrayLength > detach(ptrObserver: std::shared ptr<icObserver>): void -mObserverPtrArray notify(): void Element_T: typename ArrayLength: uint32_t Common::tcArray -mptrClockContainer

Range_T : typename ThrowingObject_T : typename Std::exception User_Exception::extcRangeException - mOutOfRangeValue: Range_T = 0 - mptrThrowingObject: ThrowingObject_T* = nullptr + extcRangeException() + extcRangeException(OutOfRangeValue: Range_T, ptrThrowingObject: ThrowingObject_T* const) + "extcRangeException() + getOutOfRangeValue(): Range_T {query} + getThrowingObject(): ThrowingObject_T* {query} + what(): char* {query}