| < <actor class="">> Librarian</actor> | | < <interface>> returnForm</interface> | |
|---|----|--|-----------|
| - Private : employeeID : Int = 0 | 0* | - bookID : int = 0 | |
| + Constructur : Librarian() | | - title : Optional : String = Null - copy# : Optional : int = 0 - customerID : int = 0 | |
| + :Librarian(Int) + collectBooks() - Self/Real-World function | | - customeriD : int = 0 - status : String = Null | |
| + returnForm() - Calls the returnForm interface | | + Constructur : returnForm() | |
| | | + :returnForm(Int, (Optional) String, | |
| | | (Optional) Int, Int) + returnHandler(Int, | |
| | | (Optional) String, (Optional) Int, | |
| | 1 | Int)- Calls the returnHandler co + getStatus() - Gets the current status | ontroller |
| | | 1 | |
| | | | |
| | | 1 | |
| | | < <controller>></controller> | |
| | | returnHandler - bookID : int = 0 | |
| | | - title : Optional : String = Null - copy# : Optional : int = 0 | |
| | | - customerID : int = 0 - status : String = Null | |
| | | + Constructur : returnHandler() | |
| | | + :returnHandler(Int, (Optional) String, | |
| | | (Optional) Int, Int) + aBT:bookReturns(Int, | |
| | | (Optional) String, (Optional) Int, | |
| | | Int) - Continues the return proce | :SS |
| | | 1 | |
| | | 1* | |
| | | aBT:bookReturns | |
| | | - bookID : int = 0 - title : Optional : String = Null | |
| | | - copy# : Optional : int = 0 - customerID : int = 0 | |
| | | - status : String = Null | |
| | | + Constructur aBT:bookReturns() + aBT:bookReturns(Int, | _ |
| | | (Optional) String, (Optional) Int, | |
| | | Int) + bookLoaned() - Continues the return process | i |
| | | 1 | |
| | | | |
| | | :bookLoaned | |
| | | - bookID : int = 0 | |
| | | - title : Optional : String = Null - copy# : Optional : int = 0 - customerID : int = 0 | |
| | | - status : String = Null - Private : loanID : int = 0 | |
| | | - Private : loanStartDate = datetime = Null - Private : loanEndDate = datetime = Null - Private : Status : String = Null | |
| | | | |
| | | + Constructur :bookLoaned() + :bookLoaned(Int, (Optional) String, | |
| | | (Optional) Int, Int) | |
| | | + updateLoanStatus(bookID, title, copy#) - Cha book into a pending/locked/being modified state + bookDisplayStatus() - Continues the return pi | е |
| | | 1 | |
| | | | |
| | | 1 | |
| | | :bookDisplayStatus | |
| | | - bookID : int = 0 - title : Optional : String = Null - copy# : Optional : int = 0 | |
| | | - customerID : int = 0 - status : String = Null | |
| | | + Constructur :bookDisplayStatus() | |
| | | + :bookDisplayStatus(Int, (Optional) String, | |
| | | (Optional) Int, Int) + updateAvailable(bookID, title, copy#) - Marks | the book |
| | | as returned/available to be loaned out + customerLoans() - Continues the return proce | |
| | | 1 | |
| | | | |
| | | 1 | |
| | | :customerLoans | |
| | | - bookID : int = 0 - title : Optional : String = Null | |
| | | - copy# : Optional : int = 0 - customerID : int = 0 - status : String = Null | |
| | | - Private : firstName : String = Null - Private : lastName : String = Null | |
| | | - Private : phoneNumber : int = 0 - Private : pastLoans(loanID, bookID, copy#, loanStartDate, loanEndDat | to) : |
| | | String = Null | , . |
| | | + Constructur :customerLoans() + :customerLoans(Int, | |
| | | (Optional) String, (Optional) Int, | |
| | | Int, String, String, int, Str + removeLoanedBook(bookID, title, copy#, | σ, |
| | | customerID) - Marks the as returned and removes it from pending loans creates a history of the loan | |
| | | + loanItem() - Continues the return process | |
| | | | |
| | | 0* | |
| | | :loanItem | |
| | | - bookID : int = 0 - title : Optional : String = Null | |
| | | - copy# : Optional : int = 0 - customerID : int = 0 - status : String = Null | |
| | | _ | |
| | 1 | + Constructur :bookDisplayStatus() + :bookDisplayStatus(Int, | |
| | | (Optional) Int, Int) | |
| | | + removeLoanedBook(bookID, title, copy#, customerID) - Marks the as returned and removes it from pending loans | |
| | | creates a history of the loan + setStatus() - Marks the transaction as comple | eted, and |
| | | removes the pending/locked state, and sets a smessage + returnForm(bookID, title, copy#, status) - | uccess |
| | | Sends back to the return form | |