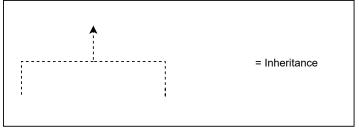
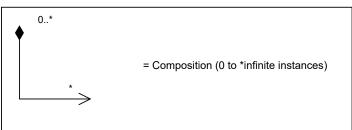
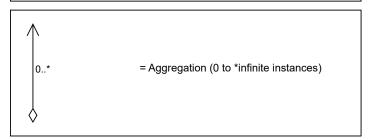
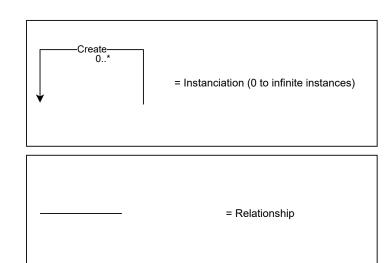
Legend:

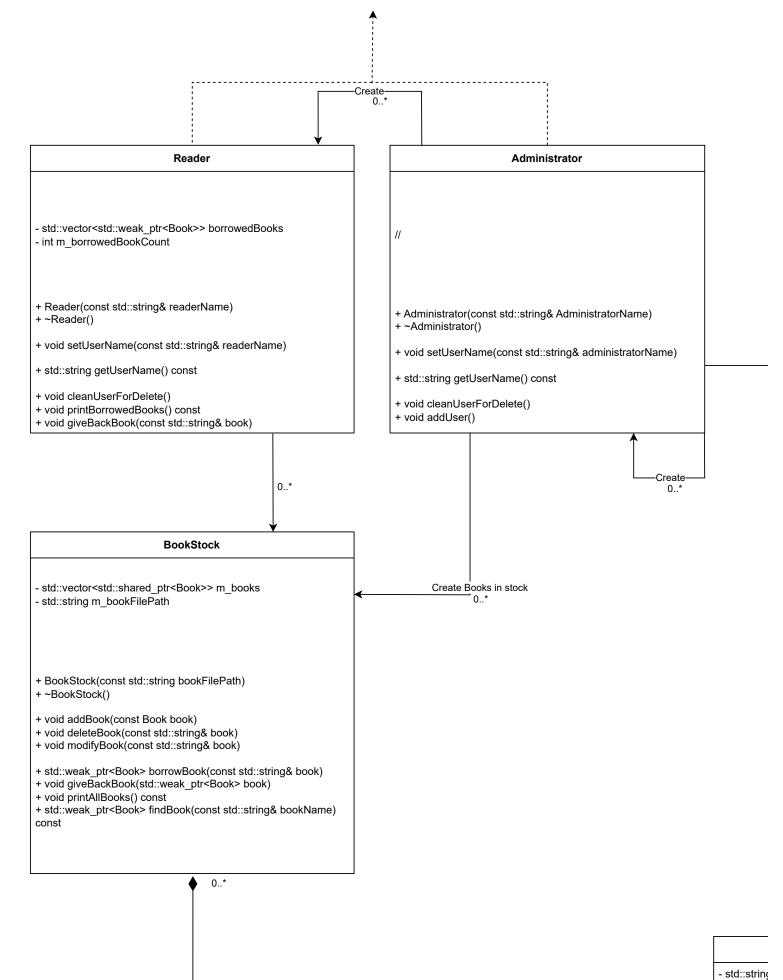








< <interface>> User</interface>				
- std::string m_userName - char m_userType				
+ User(const std::string& userName, const char userType) + ~User()				
+ void setUserName(const std::string& userName)				
+ std::string getUserName() const				
+ void cleanUserForDelete()				



- std::weak

Create Authors in Pool

AuthorPool

- std::vector<std::shared_ptr<Author>> m_authors
- std::string m_authorsFilePath
- + AuthorPool(const std::string authorFilePath) + ~AuthorPool()

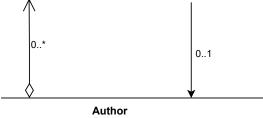
- + void addAuthor(const author author)
 + void deleteAuthor(const std::string& author)
- + void modifyAuthor(const std::string& author)
- + void printAllAuthors() const + std::weak_ptr<Author> findAuthor(const std::string& authorName) const

1..*

Book

	- มบบหับสเห - int m_pul - bool m_is
	+ Book(co category, i + ~Book()
	- void setT - void setA - void setC - void setF - void setIs
*>	- std::strinç - Author ge - bookCate - int getPu - bool isBc
	+ void prin + void mod
	- void dele
	- std::strinç - std::vecto
	+ Author(c + ~Author(
	+ void set/
	+ std::strin
	- void dele - void dele

egory m_category blicationDate sBorrowed nst std::string& title, Author author, bookCategory int publicationDate) Title(const std::string& title) Author(const Author& author) Category(const bookCategory category) PublicationDate(const int publicationDate) sBorrowed(const bool isBorrowed) g getTitle() const etAuthor() const egory getCategory() const iblicationDate() const prrowed() const ntBook() const difyBook() >teThisBookInAuthor() 0..* 0..1



g m_authorName or<std::weak_ptr<Book>> m_books

const std::string& authorName)

AuthorName(const std::string& authorName)

ng getAuthorName() const

steAllAuthorBooks()

steAuthorBook(const std::string& book)

<enumeration> bookCategory

Sci-Fi Classic Autobiography Roman **Fantasy** Thriller Essay