

Driver: Mox, Observer: Andrii

Erstellt eine neue Klasse Customer.

Importiert java.util.LinkedList

Die Klasse hat Name name und eine Liste ausgeliehenen Filmen List<Movie> loanedMovies

Constructor dazu

Kundenname wird übergeben

Liste wird erstellt

Driver: Andrii, Observer: Mox

Ändert Autherschaft der Klasse Customer auf Mox, Andrii

Fügt der Klasse Movie ein Attribut Customer zu. Autogenerated Setter/Getter dazu.

Fügt in Customer.loanMovie(Movie movie) eine Anweisung zu, die das Movie zur Liste der Ausgeliehenen Filme zufügt.

Fügt weiterhin einen Vergleich ein, ob der Film schon ausgeliehen ist. Und fügt ein, dass in Movie der Ausleiher gesetzt wird.

Ändert rückgabewert auf boolean. Wenn das Ausleihen funktioniert hat, wird true, sonst false zurückgegeben.

Driver : Mox, Observer : Andrii

Ändert die Methode loanedMovie() so, dass ein Kunde nur Filme gleichzeitig ausleihen darf, dadurch loanedMovies.size() < 5

Erweitert die Methode loanedMovie() so, wenn in der Falle das die Methode false liefert, eine richtige Ausgabe ausgegeben werden könnte.

Hinzufügt eine String Methode, die die Name der Kunde liefert.

Schreibt die main-Methode für das Testen neuen Features.

Wird zwei Kunden eingelegt und sieben Filmen, um zu prüfen, was passieren würde, wenn ein Kunde mehr als 5 Filme ausleihen will, oder zwei Kunden versuchen den gleichen Film auszuleihen.

In die main-Methode Kunde1 leiht ein Film aus und die restliche Filmen leiht Kunde2.
Ausgabe: Kunde2 hat schon 5 Filme ausgeliehen.

In die main-Methode versucht Kunde1 einen verliehenen Film auszuleihen.
Ausgabe: Film ist schon an Kunde1 mox ausgeliehen.

Erfahrungsbericht mox für das Pair-Programming

Es ist eine neue Erfahrung, nicht mehr nur vollendeten Code zu Gesicht zu bekommen, sondern auch, wie dieser entsteht. Insbesondere hat mich dabei überrascht, wie sehr sich einzelne Menschen in der Art unterscheiden können, in der sie Code schreiben. Während man bei größeren Codefragmenten, die man erst im fertigen Zustand sieht, oft von ihrer schieren Masse überwältigt ist, ist es im Gegensatz dazu eher entspannend bis langweilig, der Entstehung zuzuschauen. Dabei kann man übersehen, welche Entscheidungen der andere in der Zeit treffen muss, die man möglicherweise selbst schlichtweg übersehen hat.