	Concept Level(Basic, Intermediate, Advanced)	Concept	Where to find it		class	line	How it was demonstrated
1	intermediate	Recursion	1- Recursion in Practice	sizeHelper(Node current)	LinkList	109	Linked Lists
				printHelper(Node current)		130	
2	Intermediate	Abstract Classes and Interfaces	2- Abstract Classes and Interfaces	interface IBest			abstract class implement interface
				class Movie			
3	Intermediate	Abstracted Linked Lists	3 -Abstracted Linked Lists	class EmptyNode			abstract linked list is created by interface MovieStoreList<T> and implement with class MovieListImpl
				class MovieNode			
				class TheaterNode			
				interface MovieStoreList<T>			
4	Intermediate	Higher order functions map, filter, and fold	4- Higher order functions map, filter, and fold	count()	class HigherOrderFunctions	36	stream, map, filter, reduce
				printIsGoodMovie()		47	
				printTitleMovie()		55	
				printReleaseDateMovie()		62	
				printRatingMovie()		69	
				sortReleaseDate()		77	
				maxRating()		86	
				getSizeByGenres(MovieGenres genres)		104	
5	Intermediate	Hierarchical Data Representation	5- Hierarchical Data Representation	AbstractTreeNode<T>			Hierarchical Data Representation is created by TreeNode <T> and implement with class PersonFamilyImpl
				GroupNode<T>			
				LeafNode<T>			
				TreeNode <T>			
6	Intermediate	MVC Design	6- MVC Design	class MovieStore			Model, View, and Controller
				class MovieStoreView			
				class MovieStoreViewController			
7	Intermediate	SOLID Principles	7- SOLID Principles	1- Single Responsibility Principle			please check main class for explanation
				2- OpenClosed Principle			
				3- Liskov Substitution Principle			
				4- Interface Segregation Principle			
				5- Dependency Inversion Principle			
8	Intermediate	Observer Patterns	8- Any Design Pattern	class MovieStore			is responsible for notifying observers
				interface MovieStoreSubject			subject interface
				class User			represents a user who is also an observer
				interface UserObserver			defines the update method