Overall

Name	Annie Tran		
	Assigned	Possible	Comment
Movie	17	17	
Video	13	13	
ZipCode	7.5	7.5	
Area	12.5	12.5	
Customer	29.5	30	
Genre	5	5	
Store	20.5	20.5	
Reservation	10.5	10.5	
Rental	14.75	16.25	
CommunicationMethod	11.5	11.5	
Employee	15	15.5	
Rating	8	8	
Total	164.75	167.25	99

Movie

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
Integer property for Year is created with appropriate getters / setters	1	1	
String property for Title is created with appropriate getters / setters	1	1	
Integer property for RunningTimeInMinutes is created with appropriate getters / setters	1	1	
String property for Director is created with appropriate getters / setters	1	1	
String property for Language is created with appropriate getters / setters	1	1	
String property for Rating is created with appropriate getters / setters	1	1	
Genre property for PrimaryGenre is created with appropriate getters / setters	1	1	
Ilist<Reservation> property for Reservations is created with appropriate getters / setters	1	1	
All properties are marked as virtual	1	1	
No argument constructor created	1	1	
Reservations property initialized to empty list in constructor	0.5	0.5	

AddReservation method exists	0.5	0.5
AddReservation method marked as virtual	0.5	0.5
AddReservation takes a Customer as a parameters	0.5	0.5
AddReservation's return type is Reservation	0.5	0.5
GetHashCode method overridden	0.5	0.5
GetHashCode method returns object's ID property	0.5	0.5
Equals method overridden	0.5	0.5
Equals method returns False if object being compared is not a Movie	0.5	0.5
Equals method returns False if a movie with a different title is compared	0.5	0.5
Equals method returns False if a movie with a different year is compared	0.5	0.5
Equals method returns True if two Movies with the same Title and Year are compared	0.5	0.5
Total	17	17

Video

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
Movie Property exists with appropriate getters / setters	1	1	
DateTime property for Purchase Date exists with appropriate getters / setters	1	1	
Boolean NewArrival property exists with appropriate getters / setters	1	1	
Store property exists with appropriate getters / setters	1	1	
All properties are marked as virtual	1	1	
No argument constrctor exists	1	1	
Constructor sets PurchaseDate to the current date / time	1	1	
Constructor uses DateFactory to initialize PurchaseDate	1	1	
Constructor sets NewArrival to true	1	1	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns object's ID property	0.5	0.5	
Equals method overridden	0.5	0.5	
Equals method returns False if object being compared is not a Video	0.5	0.5	
Equals method returns False if a Video with a different ID is compared	0.5	0.5	

Equals method returns True if Video with same ID is compared	0.5	0.5
Total	13	13

ZipCode

Item	Assigned	Possible	Comment
String property for Code is created with appropriate getters / setters	1	1	
String property for City is created with appropriate getters / setters	1	1	
String property for State is created with appropriate getters / setters	1	1	
All properties are marked as virtual	1	1	
No argument constructor exists	1	1	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns Code property's GetHashCode value	0.5	0.5	
Equals method overridden	0.5	0.5	
Equals method returns False if object being compared is not a ZipCode	0.5	0.5	
Equals method returns result of using Equals on Code property	0.5	0.5	
Total	7.5	7.5	

Area

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
String property for Name is created with appropriate getters / setters	1	1	
ISet<ZipCode> property named ZipCodes exists with appropriate getters / setters	1	1	
All properties are marked as virtual	1	1	
No argument constructor exists	1	1	
Constructor initializes ZipCodes to be an empty set	0.5	0.5	
AddZipCode method exists	0.5	0.5	
AddZipCode method marked as virtual	0.5	0.5	
AddZipCode method takes a ZipCode as a parameter	0.5	0.5	
AddZipCode's return type is void	0.5	0.5	
RemoveZipCode method exists	0.5	0.5	
RemoveZipCode method marked as virtual	0.5	0.5	
RemoveZipCode method takes a ZipCode as a parameter	0.5	0.5	
RemoveZipCode's return type is void	0.5	0.5	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns object's ID property	0.5	0.5	
Equals method overridden	0.5	0.5	

Equals method returns False if object being compared is not an Area	0.5	0.5
Equals method returns False if an Area with a different Nane is compared	0.5	0.5
Equals method returns True if two Areas with the same Name are compared	0.5	0.5
Total	12.5	12.5

Customer

Item	Assigned	Possible	Comment
No argument constructor exists	1	1	
Rentals property initialized to empty list in constructor	0.5	0.5	
PreferredStores property initialized to empty list in constructor	0.5	0.5	
CommunicationTypes property initialized to empty set in constructor	0.5	0.5	
Integer property for ID is created with appropriate getters / setters	1	1	
Name property for Name is created with appropriate getters / setters	1	1	
String property for EmailAddress created with appropriate getters/setters	1	1	
String property for StreetAddress created with appropriate getters / setters	1	1	
String property for Password created with appropriate getters/setters	1	1	
String property for Phone is created with appropriate getters / setters	1	1	
ZipCode property created for ZipCode with appropriate getters / setters	1	1	
Reservation property created for Reservation with appropriate getters / setters	1	1	

Customer

Ilist<Rental> property created for Rentals with appropriate getters / setters	1	1
Ilist<Store> property created for PreferredStores with appropriate getters / setters	1	1
ISet<CommunicationMethod> property created for CommunicationTypes with appropriate getters / setters	1	1
String property created for FullName	1	1
Only getter exists for FullName property	0.5	0.5
All properties are marked as virtual	1	1
AddReservation method exists	0.5	0.5
AddReservation method marked as virtual	0.5	0.5
AddReservation takes a Movie as a parameter	0.5	0.5
AddReservation's return type is Reservation	0.5	0.5
Rent method exists	0.5	0.5
Rent method marked as virtual	0.5	0.5
Rent takes a Video as a parameter	0.5	0.5
Rent's return type is Rental	0.5	0.5
Allow method exists	0.5	0.5
Allow method marked as virtual	0.5	0.5
Allow takes a Communication Method as a parameter	0.5	0.5
Allow's return type is void	0.5	0.5
Deny method exists	0.5	0.5

Deny method marked as virtual	0.5	0.5	
Deny takes a Communication Method as a parameter	0.5	0.5	
Deny's return type is void	0.5	0.5	
AddPreferredStore method exists	0.5	0.5	
AddPreferredStore takes Store as argument	0.5	0.5	
AddPreferredStore takes integer as argument	0.5	0.5	
Default value of integer argument is -1	0	0.5	Add =-1 after the declaration of the pos parameter to accomplish this.
RemovePreferredStore exists	0.5	0.5	
RemovePreferredStore takes Store as argument	0.5	0.5	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns object's ID property	0.5	0.5	
Equals method overridden	0.5	0.5	
Equals method returns False if object being compared is not a Customer	0.5	0.5	
Equals method returns False if a Customer with a **different** email address is compared	0.5	0.5	
Equals method returns True if two Customers with the **same** email address are compared	0.5	0.5	
Total	29.5	30	

Genre

Item	Assigned	Possible	Comment
String property for Genre is created with appropriate getters / setters	1	1	
Genre property marked as virtual	0.5	0.5	
No argument constructor exists	1	1	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns Genre property's GetHashCode value	0.5	0.5	
Equals method overridden	0.5	0.5	
Equals method returns False if object being compared is not a Genre	0.5	0.5	
Equals method returns result of using Equals on Genre property	0.5	0.5	
Total	5	5	

Store

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
String property for StreetAddress is created with appropriate getters / setters	1	1	
ZipCode property named ZipCode is created with appropriate getters / setters	1	1	
String property for PhoneNumber is created with appropriate getters / setters	1	1	
Ilist<Employee> created for Managers	1	1	
Ilist<Video> created for Videos	1	1	
All properties are marked as virtual	1	1	
No argument constructor exists	1	1	
Constructor initializes Managers collection to empty list	0.5	0.5	
Constructor initializes Videos collection to empty list	0.5	0.5	
AddManager method created	0.5	0.5	
AddManager marked as virtual	0.5	0.5	
AddManager takes Employee object as parameter	0.5	0.5	
AddManager's return type is void	0.5	0.5	
RemoveManager method created	0.5	0.5	
RemoveManager marked as virtual	0.5	0.5	

RemoveManager takes Employee object as parameter	0.5	0.5
RemoveManager's return type is void	0.5	0.5
AddVideo method created	0.5	0.5
AddVideo marked as virtual	0.5	0.5
AddVideo takes Video object as parameter	0.5	0.5
AddVideos's return type is void	0.5	0.5
RemoveVideo method created	0.5	0.5
RemoveVideo marked as virtual	0.5	0.5
RemoveVideo takes Video object as parameter	0.5	0.5
RemoveVideos's return type is void	0.5	0.5
GetHashCode method overridden	0.5	0.5
GetHashCode method returns object's ID property	0.5	0.5
Equals method overridden	0.5	0.5
Equals method returns False if object being compared is not a Store	0.5	0.5
Equals method returns False if a Store with a different StreetAdress is compared	0.5	0.5
Equals method returns False if a store with a different ZipCode is compared	0.5	0.5
Equals method returns True if two Stores with the same StreetAddress and ZipCode are compared	0.5	0.5
Total	20.5	20.5

Reservation

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
Movie property for Movie created with appropriate getters / setters	1	1	
Customer property named Customer is created with appropriate getters / setters	1	1	
DateTime property for ReservationDate is created with appropriate getters / setters	1	1	
All properties marked as virtual	1	1	
No argument constructor exists	1	1	
ReservationDate defaults to current date in the constructor	0.5	0.5	
DateFactory class is used to get the current date	0.5	0.5	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns object's ID property	0.5	0.5	
Equals method overridden	0.5	0.5	
Equals method returns False if object being compared is not a Reservation	0.5	0.5	
Equals method returns False if a Reservation with a different **Movie** is compared	0.5	0.5	
Equals method returns False if a Reservation with a different **Customer** is compared	0.5	0.5	

Equals method returns True if two Reservations with the same customer and movie are compared	0.5	0.5
Total	10.5	10.5

Rental

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
Customer property created with appropriate getters / setters			
Video property created with appropriate getters / setters	1	1	
DateTime property for RentalDate is created with appropriate getters / setters	1	1	
DateTime property for DueDate is created with appropriate getters / setters	1	1	
DateTime property for ReturnDate is created with appropriate getters and setters	1	1	
Rating object created with appropriate getters / setters	0.5	0.5	
All properties marked as virtual	1	1	
No argument constructor exists	1	1	This constructor checks the Video property, which will always be null in this constructor. That check should only happen in the other constructor
RentalDate defaults to current date in the constructor	0.5	0.5	
DateFactory class is used to get the current date	0.5	0.5	
Constructor that takes a Customer and a Video exists	0	0.5	It takes just a video, not a Customer

Constructor calls the no-argument constructor	0.5	0.5	It duplicates the functionality of the no-arg constructor by setting the RentalDate to the CurrentDate. It would be better to call the no-arg constructor instead
Constructor sets Customer property to customer parameter	0	0.5	
Constructor sets Video property to video parameter	0	0.5	
DueDate defaults to 7 days from rental date in the constructor for non new-releases	0.5	0.5	
DueDate defaults to 3 days from rental date in the constructor for new releases	0.5	0.5	
Return method exists	0.5	0.5	
Return method marked as virtual	0.5	0.5	
Return's return type is ReturnReceipt	0.5	0.5	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns object's ID property	0.5	0.5	
Equals method overridden	0.5	0.5	
Equals method returns False if object being compared is not a Rental	0.5	0.5	
Equals method returns False if a Rental with a different **Video** is compared	0.25	0.25	

Equals method returns False if a Rental with a different **Customer** is compared	0.25	0.25
Equals method returns False if a Rental with a different **RentalDate** is comapred	0.25	0.25
Equals method returns True if two Rentals with the same values for Video, Customer, and RentalDate are compared	0.5	0.5
Total	14.75	16.25

CommunicationMethod

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
String property for Name created with appropriate getters / setters	1	1	
int property for Frequency created with appropriate getters / setters	1	1	
TimeUnit enum created	0.5	0.5	
TimeUnit enum has values *Day* , *Week* , *Month* , and *Year*	0.5	0.5	
TimeUnit property named Units created with appropriate getters / setters	1	1	
ISet<Customer> named Customers created with appropriate getters and setters	1	1	
All properties are marked as virtual	1	1	
No argument constructor exists	1	1	
Customers collection initialized in constructor	0.5	0.5	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns hash code associated with the Name	0.5	0.5	
Equals method overridden	0.5	0.5	
Equals method returns False if object being compared is not a CommunicationMethod	0.5	0.5	

Equals method returns False if a CommunicationMethod with a different Name is compared	0.5	0.5
Equals method returns True if two CommunicationMethods with the same Name are compared	0.5	0.5
Total	11.5	11.5

Employee

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
Name property for Name created with appropriate getters / setters	1	1	
DateTime property for DateHired created with appropriate getters / setters	1	1	
DateTime property for DateOfBirth created with appropriate getters / setters	1	1	
String property for Username created with appropriate getters / setters	0.5	1	It is called UserName instead of Username
String property for Password created with appropriate getters / setters	1	1	
Store property named Store created with appropriate getters / setters	1	1	
bool IsManager property created with only getter	1	1	
Employee property named Supervisor created with appropriate getters / setters	1	1	
All properties marked as virtual	1	1	
No argument constructor exists	1	1	
DateHired defaulted to the current date and time in the constructor	0.5	0.5	
DateFactory class used to access current date / time	1	1	

GetHashCode method overridden	0.5	0.5
GetHashCode method returns object's ID property	0.5	0.5
Equals method overridden	0.5	0.5
Equals method returns False if object being compared is not an Employee	0.5	0.5
Equals method returns True if two Employees with the same Name and Date of Birth are compared	0.5	0.5
Equals method returns False if two employees with different **names** are compared	0.25	0.25
Equals method returns False if two employees with different **birth dates** are compared	0.25	0.25
Total	15	15.5

Rating

Item	Assigned	Possible	Comment
Integer property for ID is created with appropriate getters / setters	1	1	
Integer property for Score created with appropriate getters / setters	1	1	
String property for Comment created with appropriate getters / setters	1	1	
All properties marked as virtual	1	1	
No argument constructor exists	1	1	
GetHashCode method overridden	0.5	0.5	
GetHashCode method returns object's ID property	0.5	0.5	
Equals method overridden	0.5	0.5	
Equals method returns False if object being compared is not a Rating	0.5	0.5	
Equals method returns False if a Rating with a different ID is compared	0.5	0.5	
Equals method returns True if two Ratings with the same ID are compared	0.5	0.5	
Total	8	8	