Gladiator – Class Diagram

A media streaming company called Gladiator Prime rents streamed movies to customers. Movies are rented for a certain number of days, and the price of the movie is deducted from the customers balance when they rent the movie. Some movies are available to rent, but others have been retired, or are still at pre-release status. A customer can rent as many movies as they like if they have enough money.

1. Class Diagram

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | **StreamedMovie** | | private double price;  private int id;  private String name;  private int numDays;  private Status status; | | public StreamedMovie(int id, String name, double price, int numDays, Status status)  public boolean matches(int id)  public boolean rent()  public double getPrice()  public boolean isAvailable()  public String getName()  public void show()  public String toString() | | |  | | --- | | **Customer** | | private int id;  private String name;  private double balance = 100.0;  private LL<StreamedMovie> movies; | | public Customer(int id, String name)  public void rent(StreamedMovie movie)  public void show()  public String toString()  public String getName()  public double getBalance()  public boolean matches(int id) | | |  | | --- | | **<<ENUM>>**  **Status** | | FOR\_RENT,  UNAVAILABLE | |  | |

1. Identify common attributes and methods between StreamedMovie and Customer - are there common features that can be abstracted into a parent class? Remember: **common methods MUST have the same signature.** What can we call the parent?

Record / Entity

1. Update your class diagram with the new parent class - make sure you remove common features from the child classes.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | **StreamedMovie** | | private double price;  private int numDays;  private Status status; | | public StreamedMovie(int id, String name, double price, int numDays, Status status) // super call  public boolean rent()  public double getPrice()  public boolean isAvailable()  public String toString() | | |  | | --- | | **Record** | | protected int id;  protected String name; | | public Record(int id, String name)  public boolean matches(int id)  public String getName()  public void show()  public String toString() // will be overridden |  |  | | --- | | **Customer** | | private double balance = 100.0;  private LL<StreamedMovie> movies; | | public Customer(int id, String name) // super call  public void rent(StreamedMovie movie)  public double getBalance()  public String toString() | | |  | | --- | | **<<ENUM>>**  **Status** | | FOR\_RENT,  UNAVAILABLE | |  | |