

1. **\*\*Associação de Composição (GameLibrary e Game):\*\***

- A classe `GameLibrary` possui uma associação de composição com a classe `Game`. Isso significa que a `GameLibrary` é responsável pela criação e destruição dos objetos `Game`. Os jogos são parte integrante da biblioteca e são destruídos quando a biblioteca é destruída.

2. **\*\*Associação de Agregação (RentalSystem e Customer):\*\***

- A classe `RentalSystem` possui uma associação de agregação com a classe `Customer`. Isso sugere uma relação mais fraca em comparação com a composição. Se o sistema de aluguel for destruído, os clientes ainda podem existir independentemente.

3. **\*\*Associação de Dependência (RentalRecord e Rental):\*\***

- A classe `RentalRecord` tem uma dependência em relação à classe `Rental`, pois ela depende de instâncias de `Rental` para serem criadas em seu construtor. No entanto, não há uma relação de longo prazo, e `Rental` pode existir independentemente de `RentalRecord`.

4. **\*\*Herança (RentalException e Exception):\*\***

- A classe `RentalException` herda da classe `Exception`, estabelecendo uma relação de herança. Isso significa que `RentalException` é uma especialização de `Exception` e herda propriedades e comportamentos dessa classe mais genérica.