

# SOFTWARE ENGINEERING

## CSE 470 – Class Diagram Design

BRAC University



Inspiring Excellence

# Guidelines

## for Analyzing Requirements / Use Cases

- A common or improper noun implies a class of objects
- A proper noun implies an instance of a class
- A collective noun implies a class of objects made up of groups of instances of another class



# Guidelines for Analyzing Requirements / Use Cases (2)

---

- An adjective implies an attribute of an object
- A doing verb implies an operation
- A being verb implies a relationship between an object and its class
- A having verb implies an aggregation or association relationship



# Guidelines for Analyzing Requirements / Use Cases (3)

---

- A transitive verb implies an operation
- An intransitive verb implies an exception
- A predicate or descriptive verb phrase implies an operation
- An adverb implies an attribute of a relationship or an operation



# Class Diagram

---

- Ensure that the classes are both necessary and sufficient to solve the underlying problem
  - no missing attributes or methods in each class
  - no extra or unused attributes or methods in each class
  - no missing or extra classes



# Discarding Unnecessary and Incorrect Classes

---

- ❑ Redundant Classes: Some potential classes differ only in name.
- ❑ Irrelevant Classes: Classes that have nothing to do with the system. Example: *computer connection*
- ❑ Vague Classes: Classes whose meaning is not clear at all. Examples: *system* and *software*



# Discarding Unnecessary and Incorrect Classes

---

- ❑ Attributes: Some nouns in the list above are likely to be modeled as attributes rather as classes. Examples: *author, title*
- ❑ Operations: Some nouns are likely to be operations rather than classes. Example: *book search*.
- ❑ Roles: Some nouns are roles of objects involved in associations rather than classes.
- ❑ Implementation Constructs: Anything that is not part of the real-world problem.



# Types of Classes

---

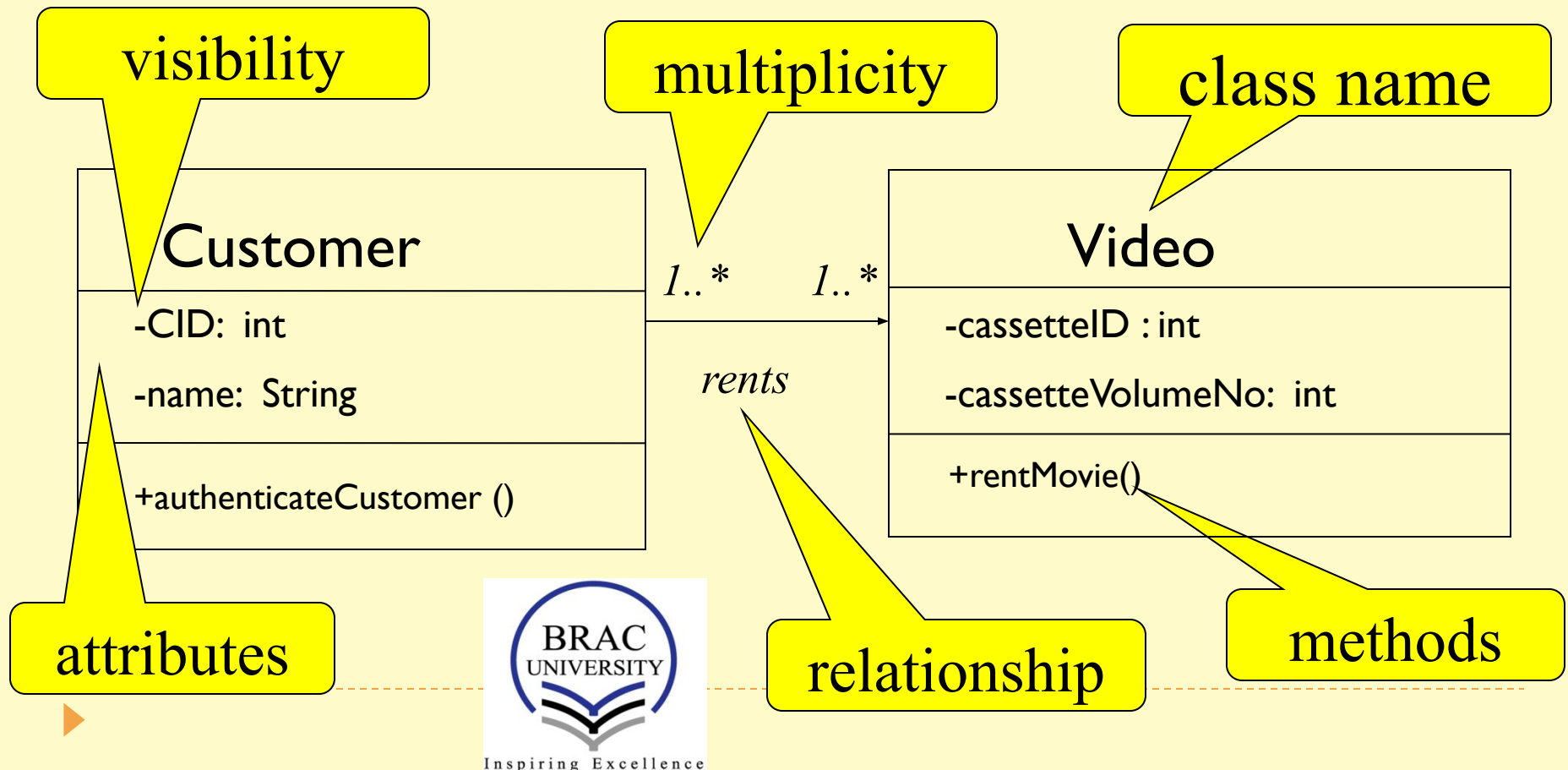
- Ones found during analysis:
  - people, places, events, and things about which the system will capture information
  - ones found in application domain
- Ones found during design
  - specific objects like windows and forms that are used to build the system



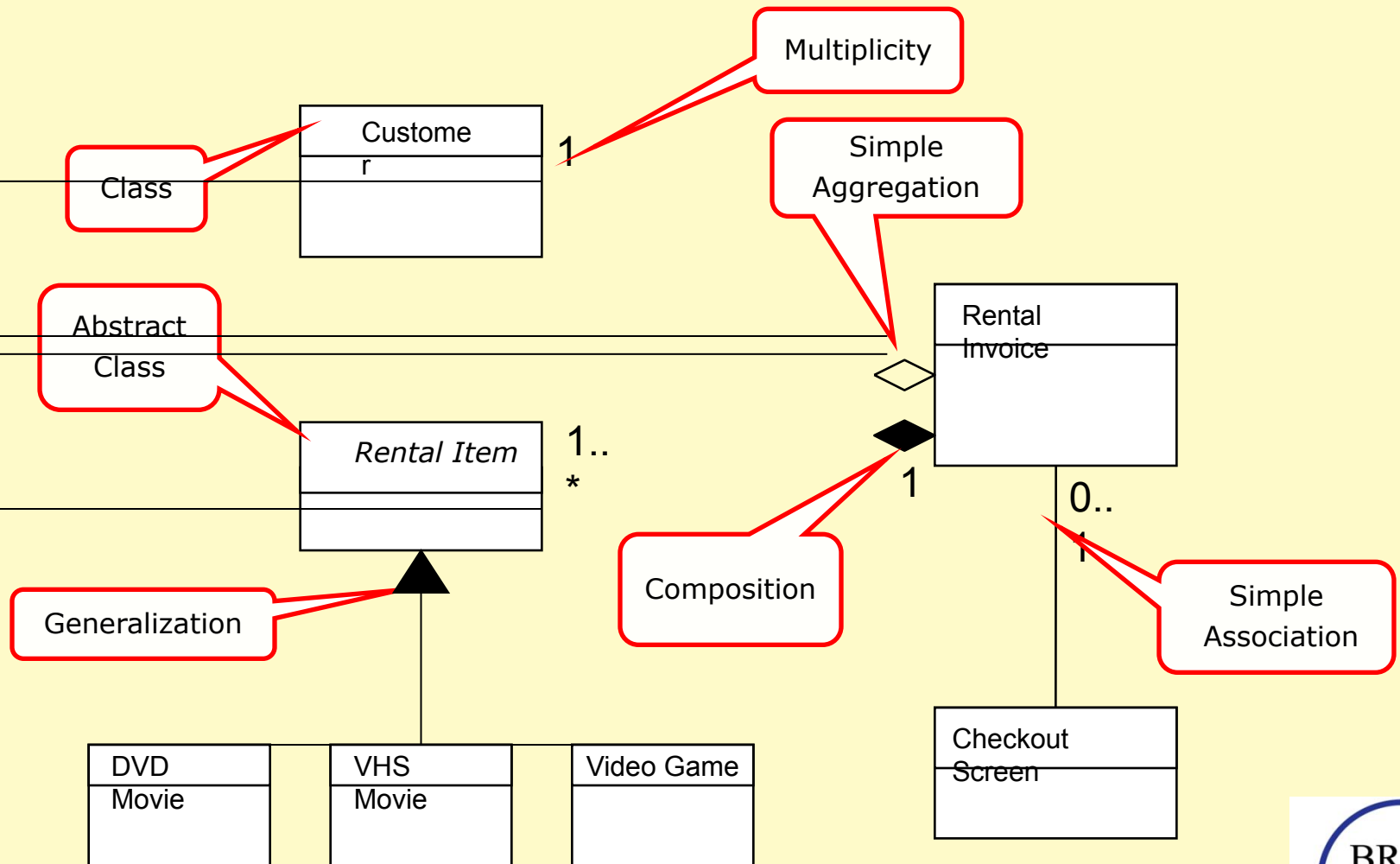


# Example of a Class Diagram

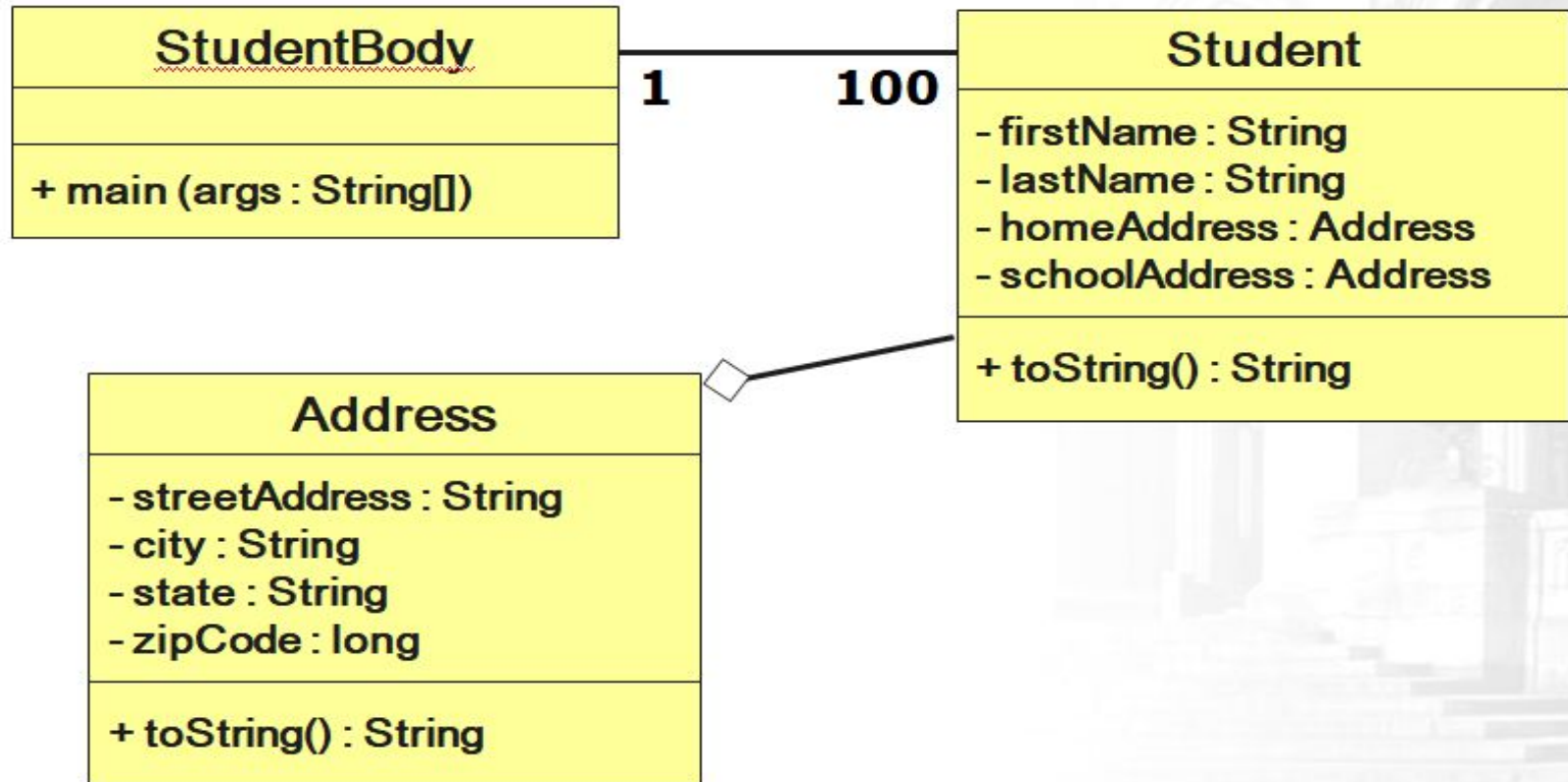
## Video Rental System



# Class diagram example 1



# Class diagram example 2



# Class Diagram Example 3

