



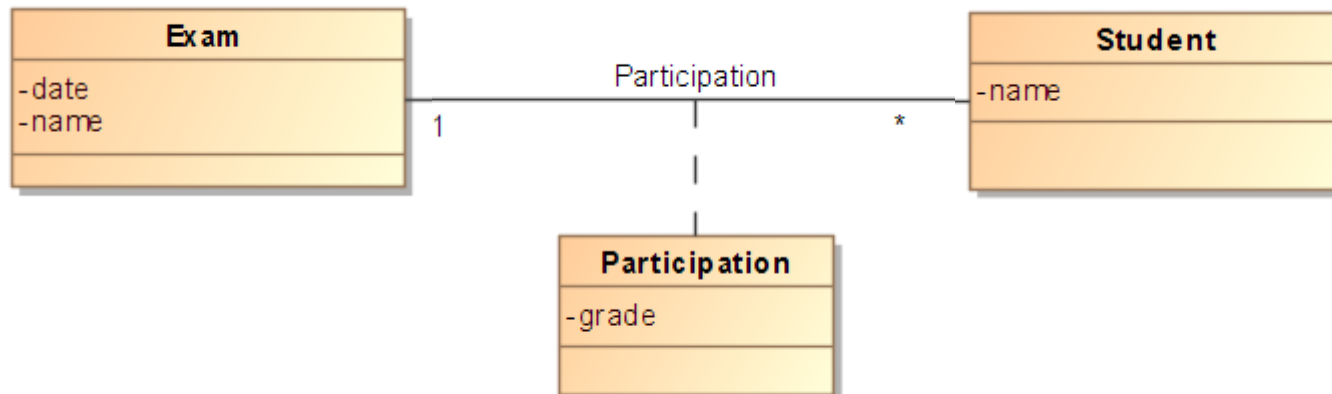
Class modelling (part 4)

Fabrizio Maria Maggi

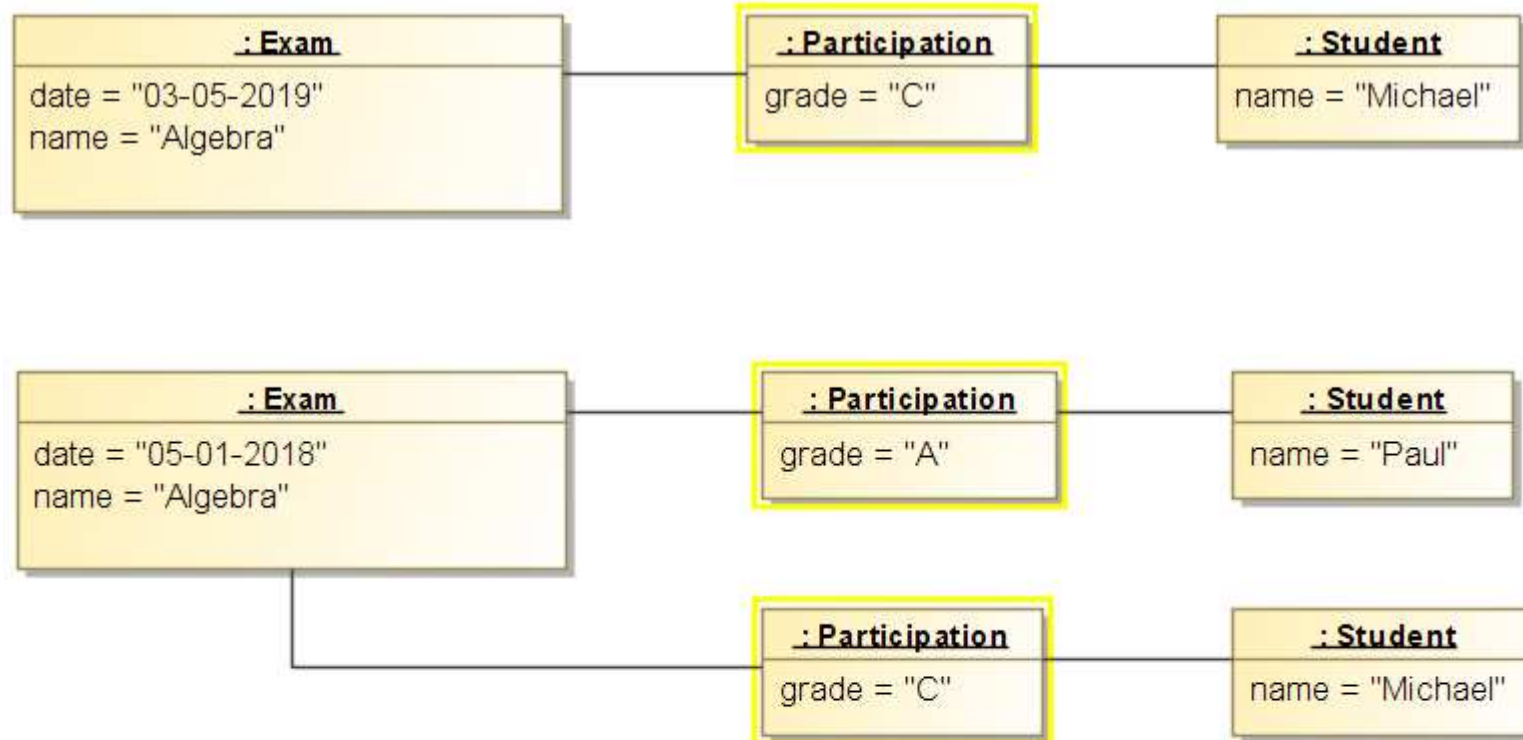
Institute of Computer Science

Association classes

- ▶ An association class is an association that is also a class
- ▶ Like a class, an association class can have attributes and operations and participate in associations
- ▶ UML notation: class box attached to the association by a dashed line

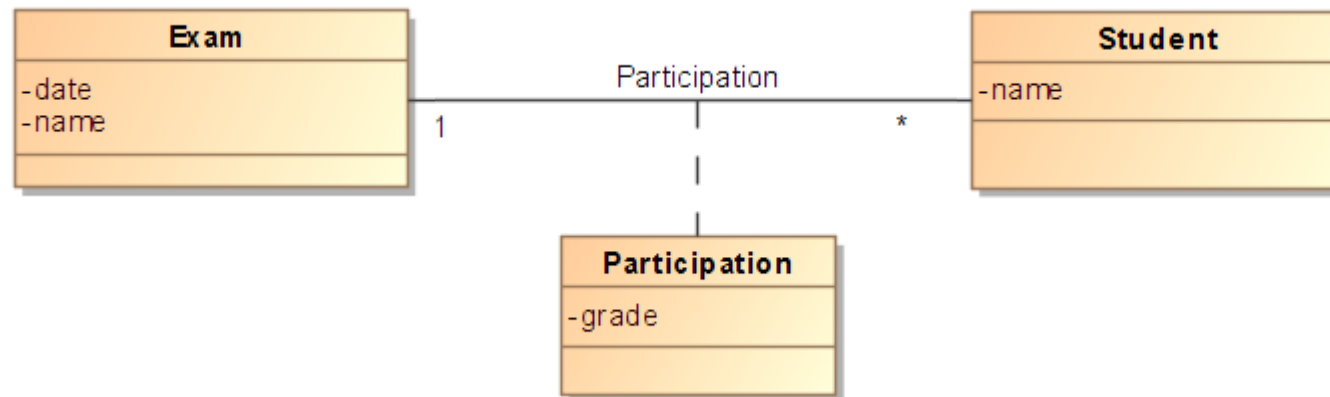


Association classes (instantiations)



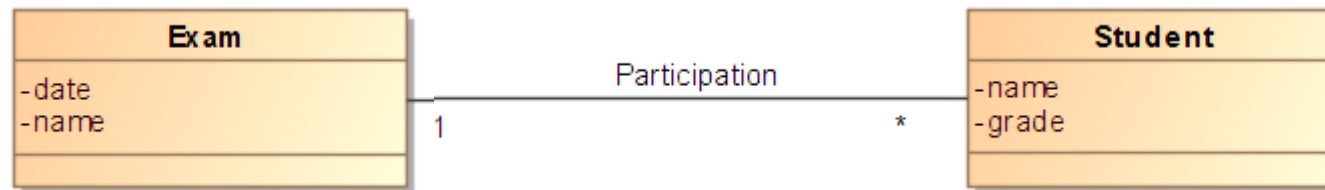
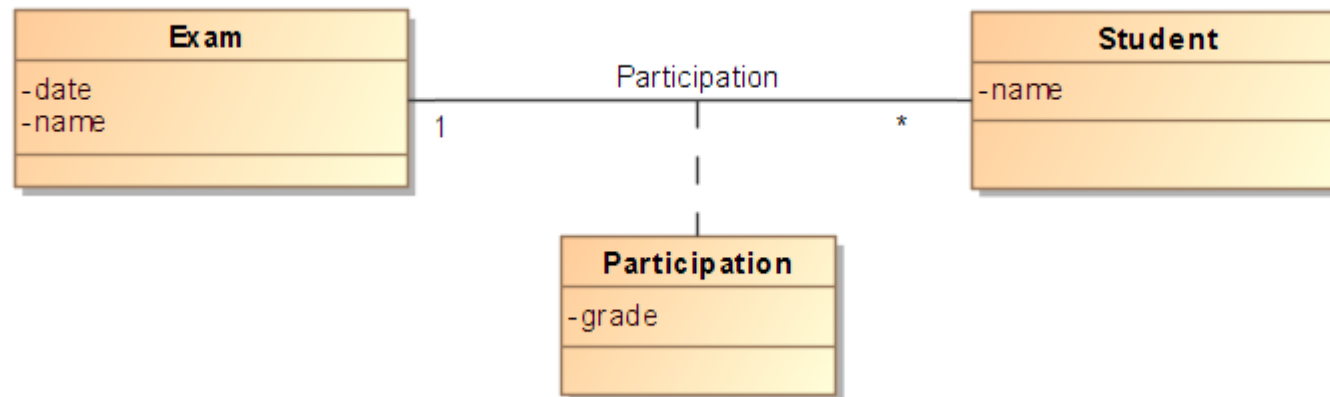
Association classes

- ▶ Alternative representations in case of multiplicity one-to-many



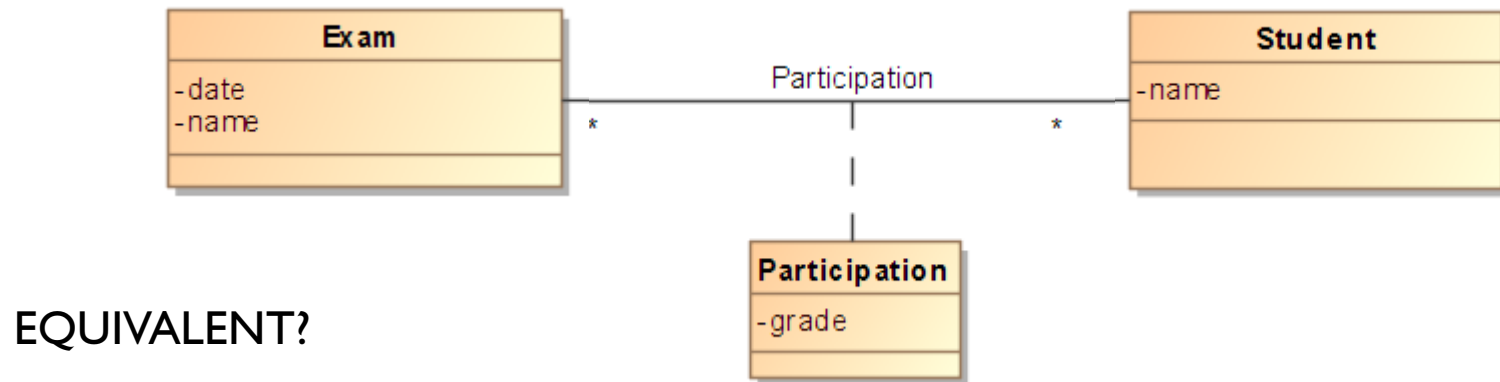
Association classes

- ▶ Alternative representations in case of multiplicity one-to-many



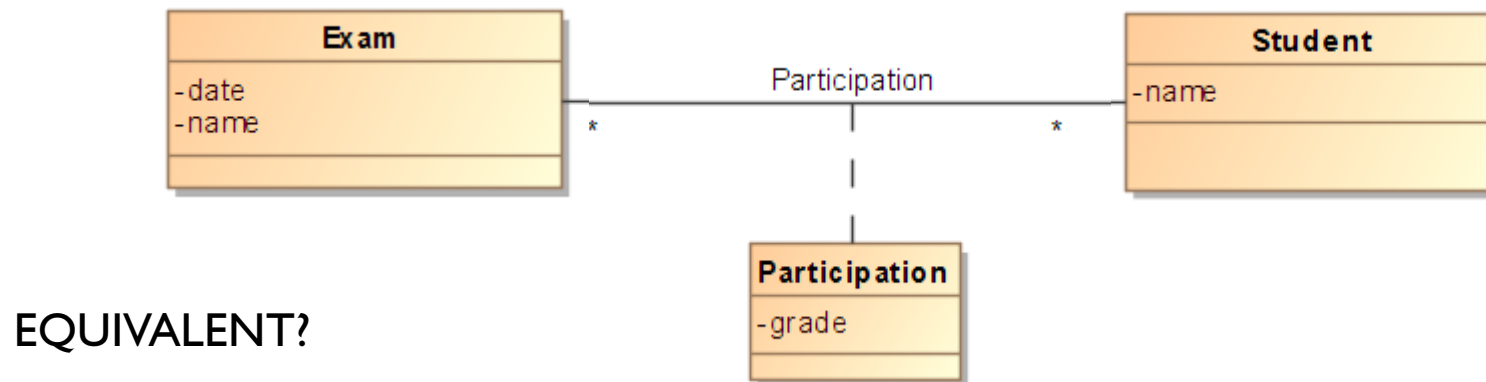
Association classes

- ▶ What about the case of many-to-many?



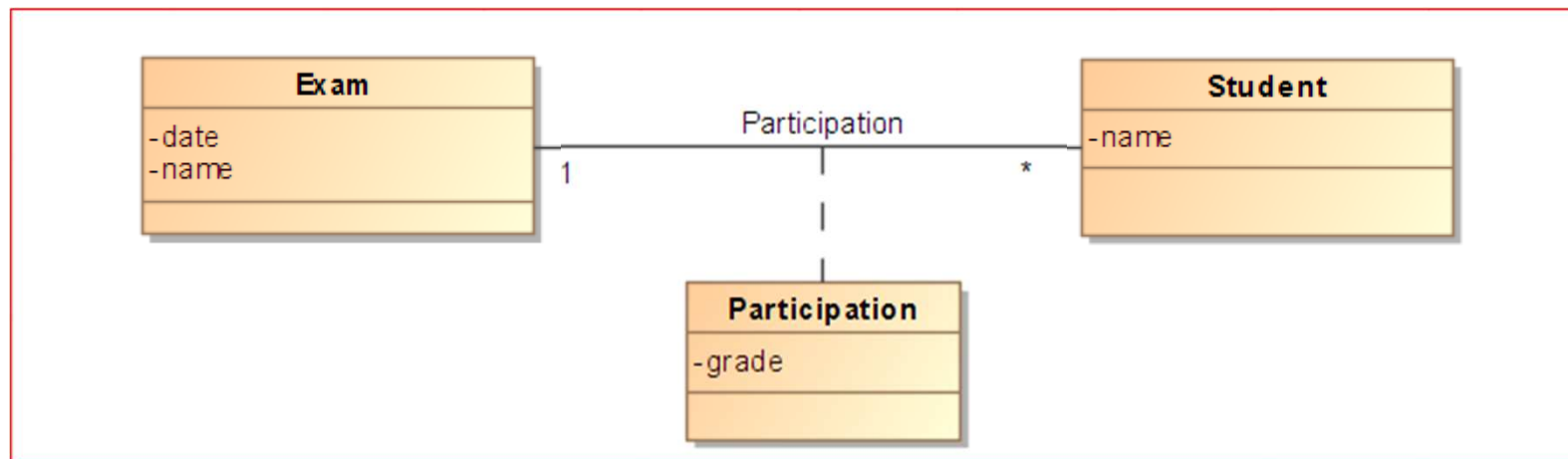
Association classes

- ▶ What about the case of many-to-many?



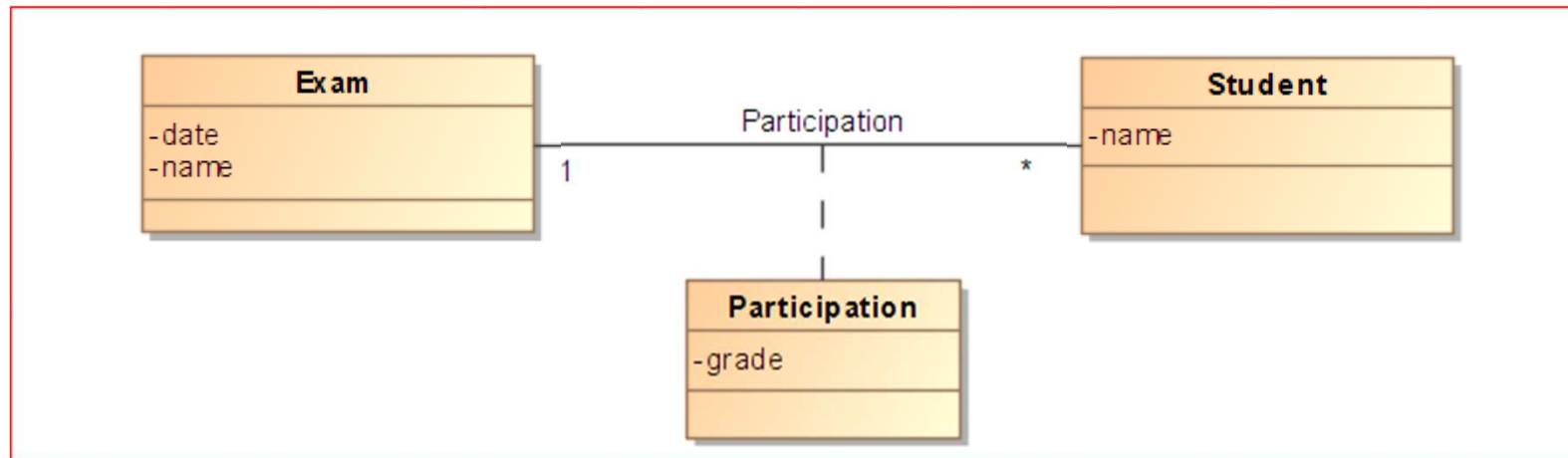
Association classes

► **PREFERABLE.**



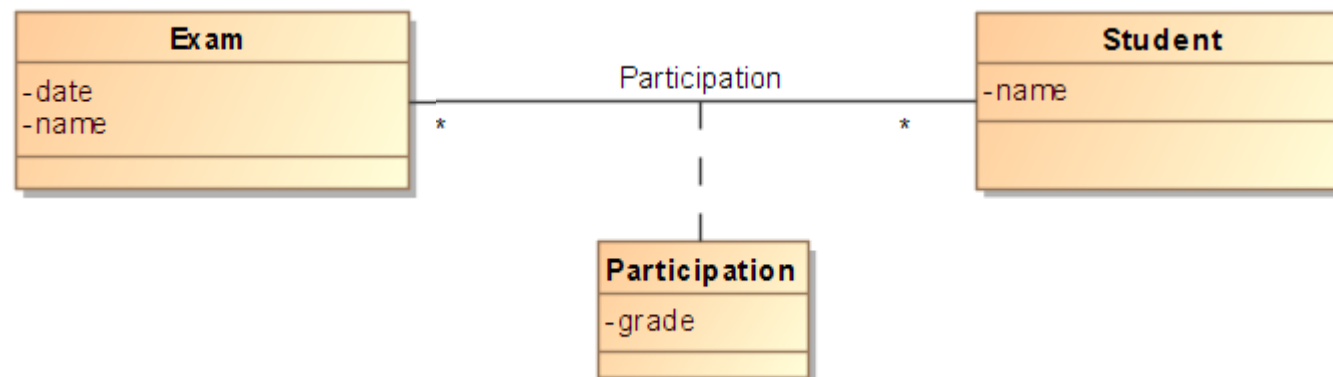
Association classes

- ▶ **PREFERABLE.** What if the multiplicity changes?



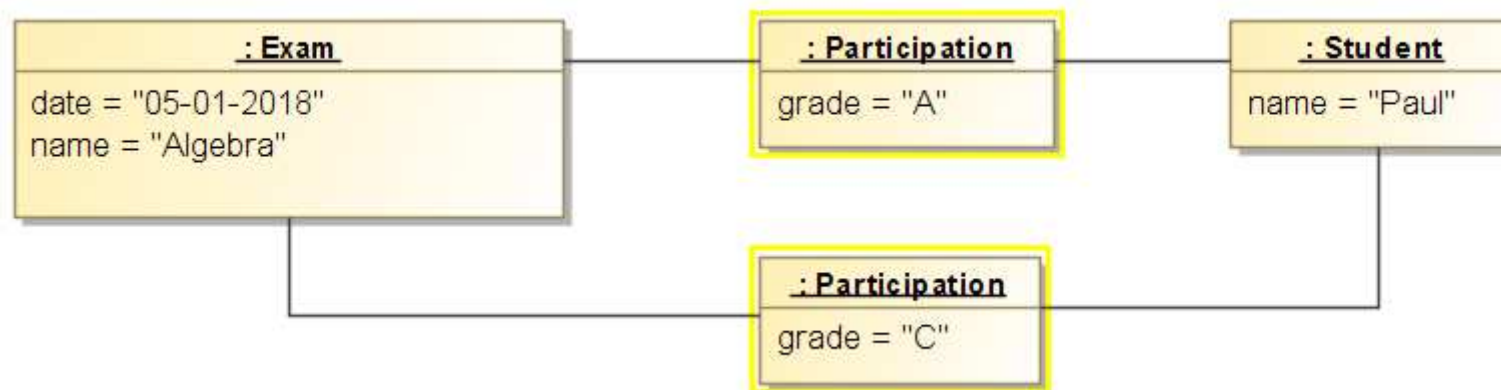
Association classes

- ▶ Since it is not possible to have two links of the same association between the same two objects, there can be only one instance of the association class between any two participating objects



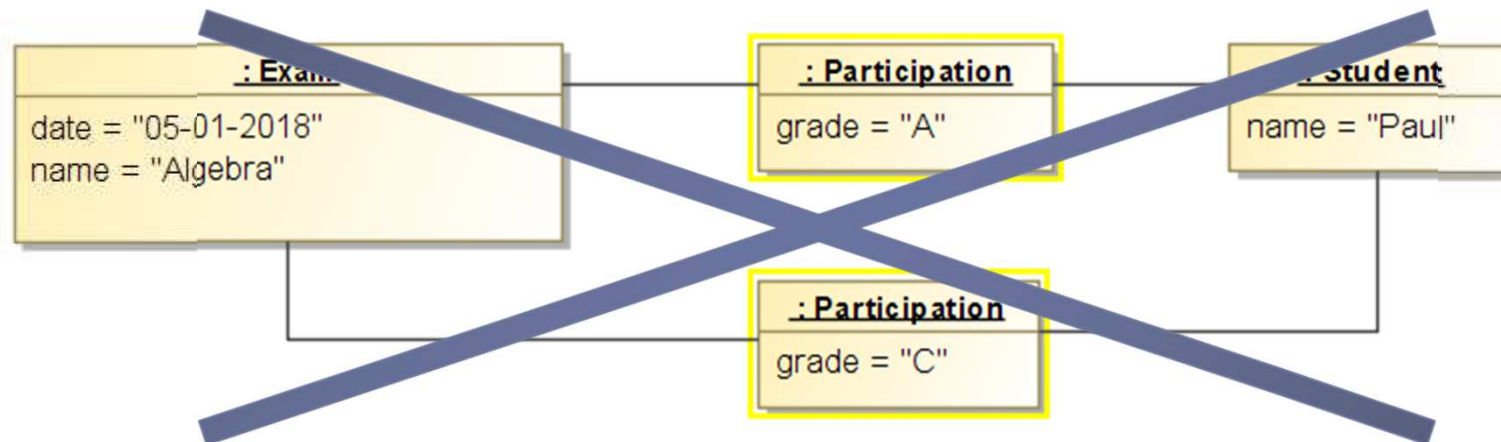
Association classes

- ▶ NOT POSSIBLE (Participation is a link and it is not possible to have two links of the same association between the same two objects)



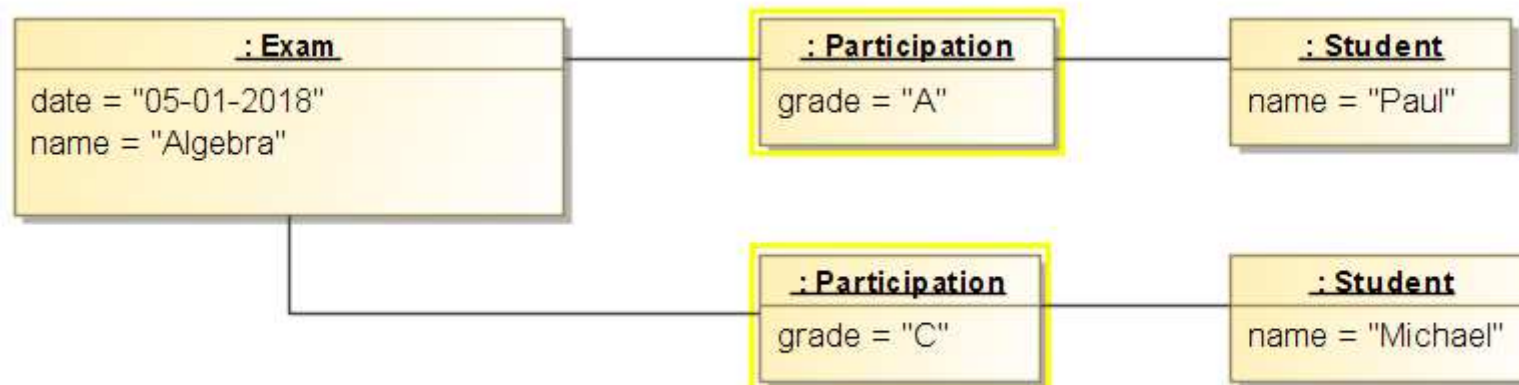
Association classes

- ▶ NOT POSSIBLE (Participation is a link and it is not possible to have two links of the same association between the same two objects)



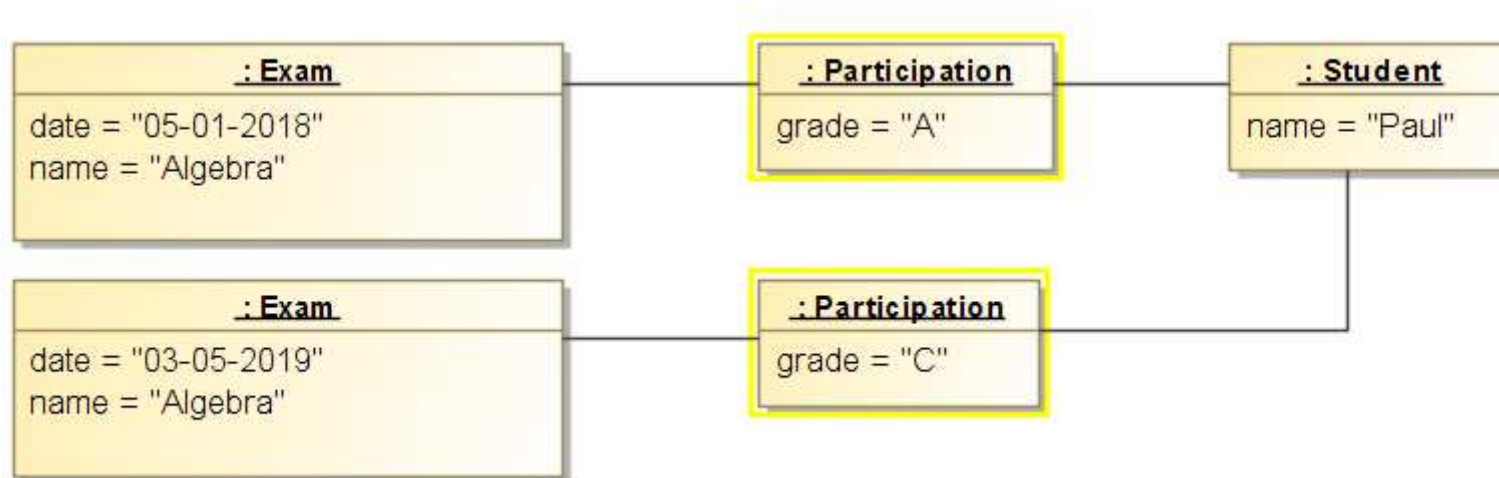
Association classes

▶ POSSIBLE



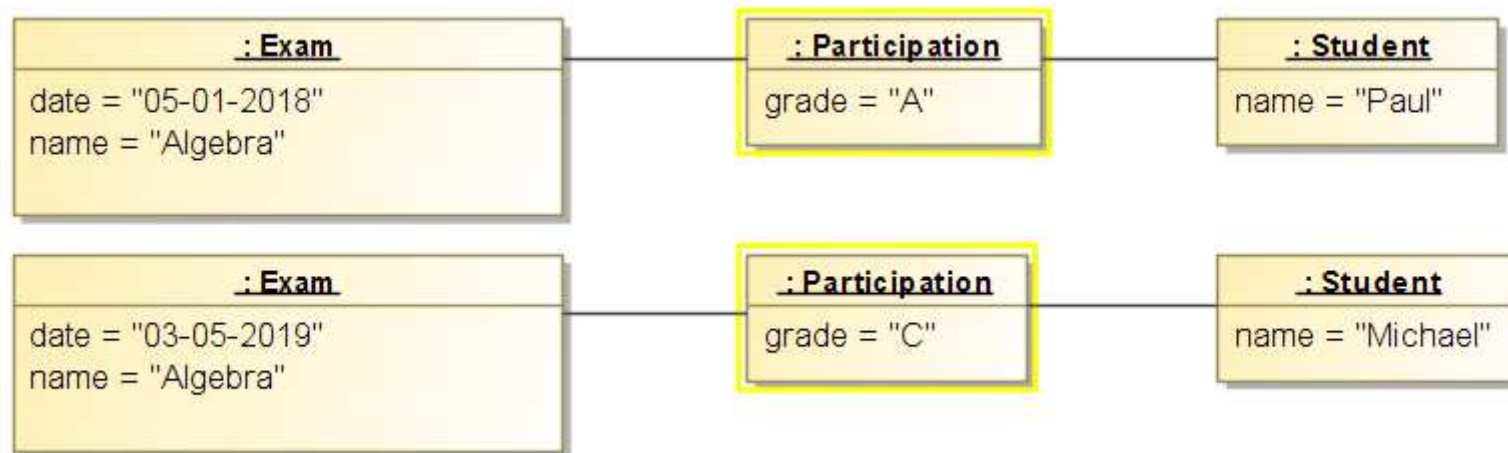
Association classes

▶ POSSIBLE



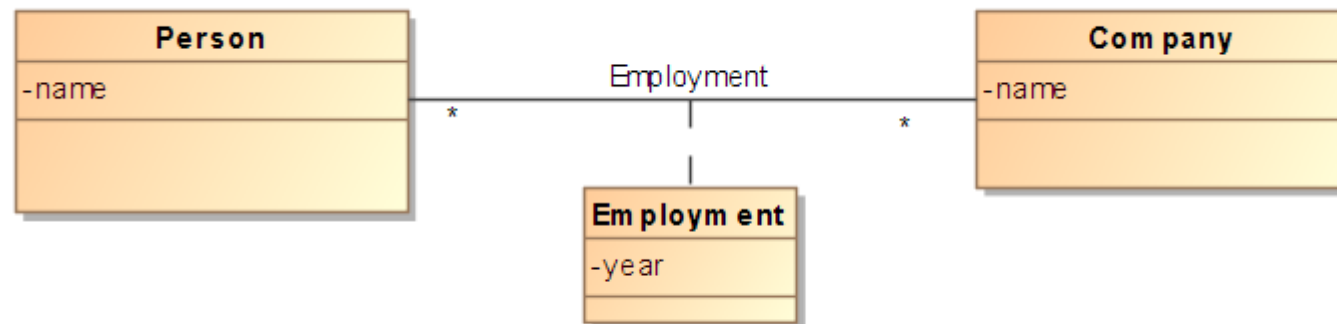
Association classes

► POSSIBLE



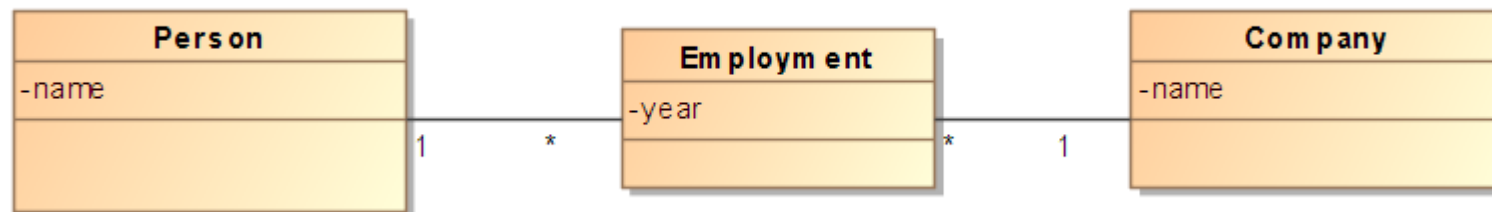
Class or Association Class?

- ▶ The same person **CANNOT** be employed by the same company in different years (Employment is a link and it is not possible to have two links of the same association between the same two objects)



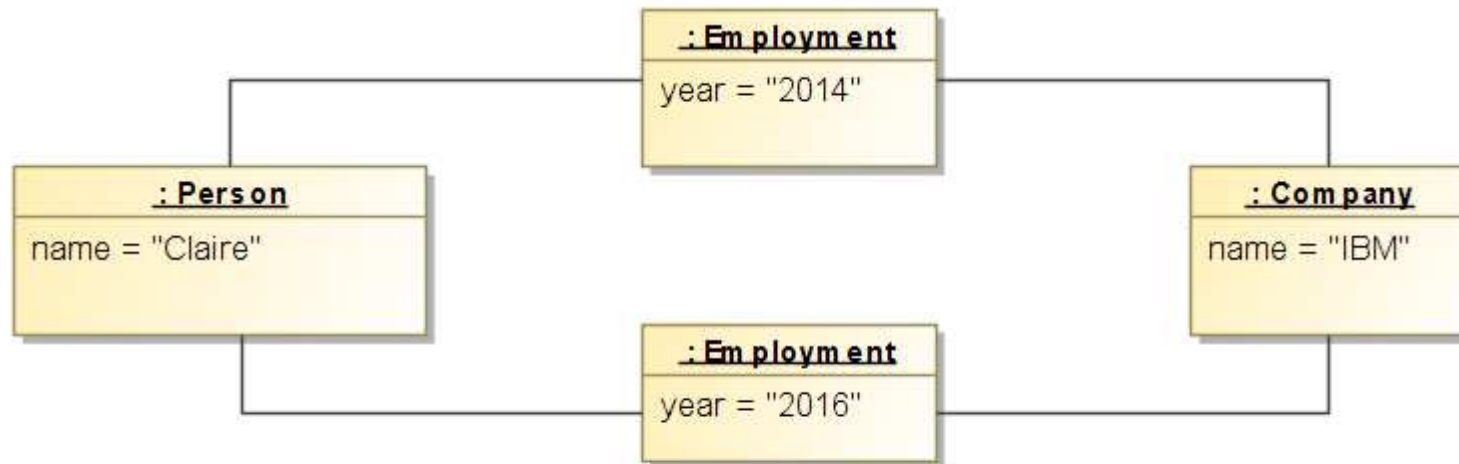
Class or Association Class?

- ▶ The same person **CAN** be employed by the same company in different years



Class or Association Class?

- ▶ POSSIBLE (Employment is a class now and not a link)



Exercise 1

- (a) In a system to manage employments of people in companies, a person can be employed in different companies, in each company only with one specific role.
- (b) In a system to manage employments of people in companies, a person can be employed in different companies. In each company the person can cover different roles.

Exercise 2

In an online system to deal with flight bookings, each flight has a departure airport and an arrival airport and it is characterized by a date and a flight number. Each airport has a name and is located in a city. A flight has a plane (characterized by a model and a serial number) and a list of passengers each occupying a seat in the plane. A flight is managed by only one airline and has 2 pilots.

Exercise 3

3.20 (6) Prepare a class model to describe undirected graphs. An undirected graph consists of a set of vertices and a set of edges. Edges connect pairs of vertices. Your model should capture only the structure of graphs (i.e., connectivity) and need not be concerned with layout such as location of vertices or lengths of edges. Figure E3.10 shows a typical undirected graph.

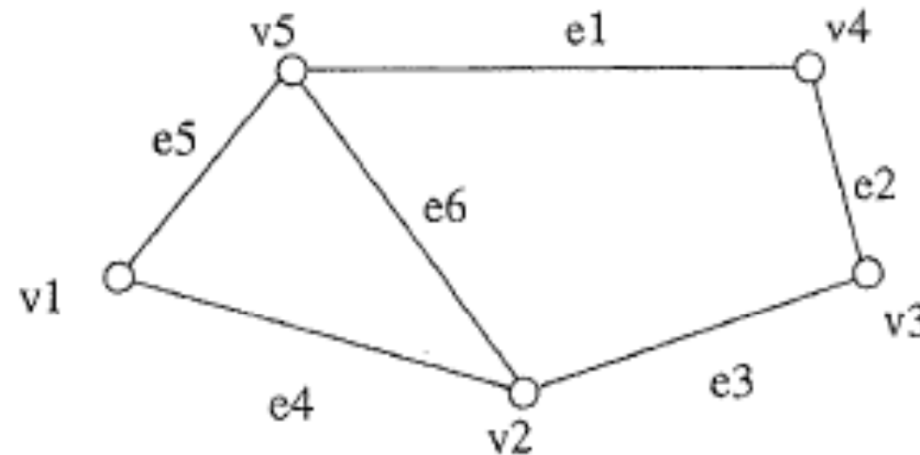


Figure E3.10 Sample undirected graph

Question 1

- ▶ In what type of diagram a connection between the same two nodes is not necessarily unique? How would you represent the connection in this case?

Association Classes: A possible implementation



```
public class Transcript {
    //Transcript's properties
}

public class Course {

    private Map<Student, Transcript> transcriptsByStudent;
}

public class Student {

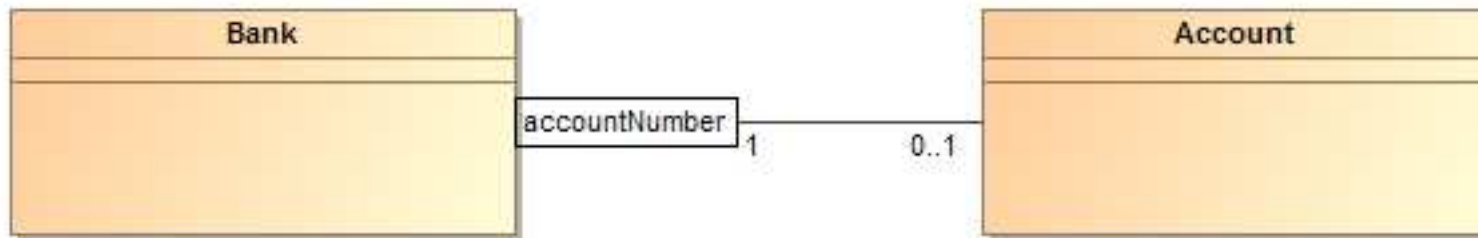
    private Map<Course, Transcript> transcriptsByCourse;
}
```

Qualified Associations



- ▶ What is the meaning of this association?
- ▶ How can we implement it?
- ▶ Is this a realistic representation?

Qualified Associations



- ▶ It is possible to define qualifiers for one-to-many and many-to-many associations
- ▶ A qualifier selects among the target objects, reducing the multiplicity from *many* to *one*
- ▶ How can we implement a qualified association?

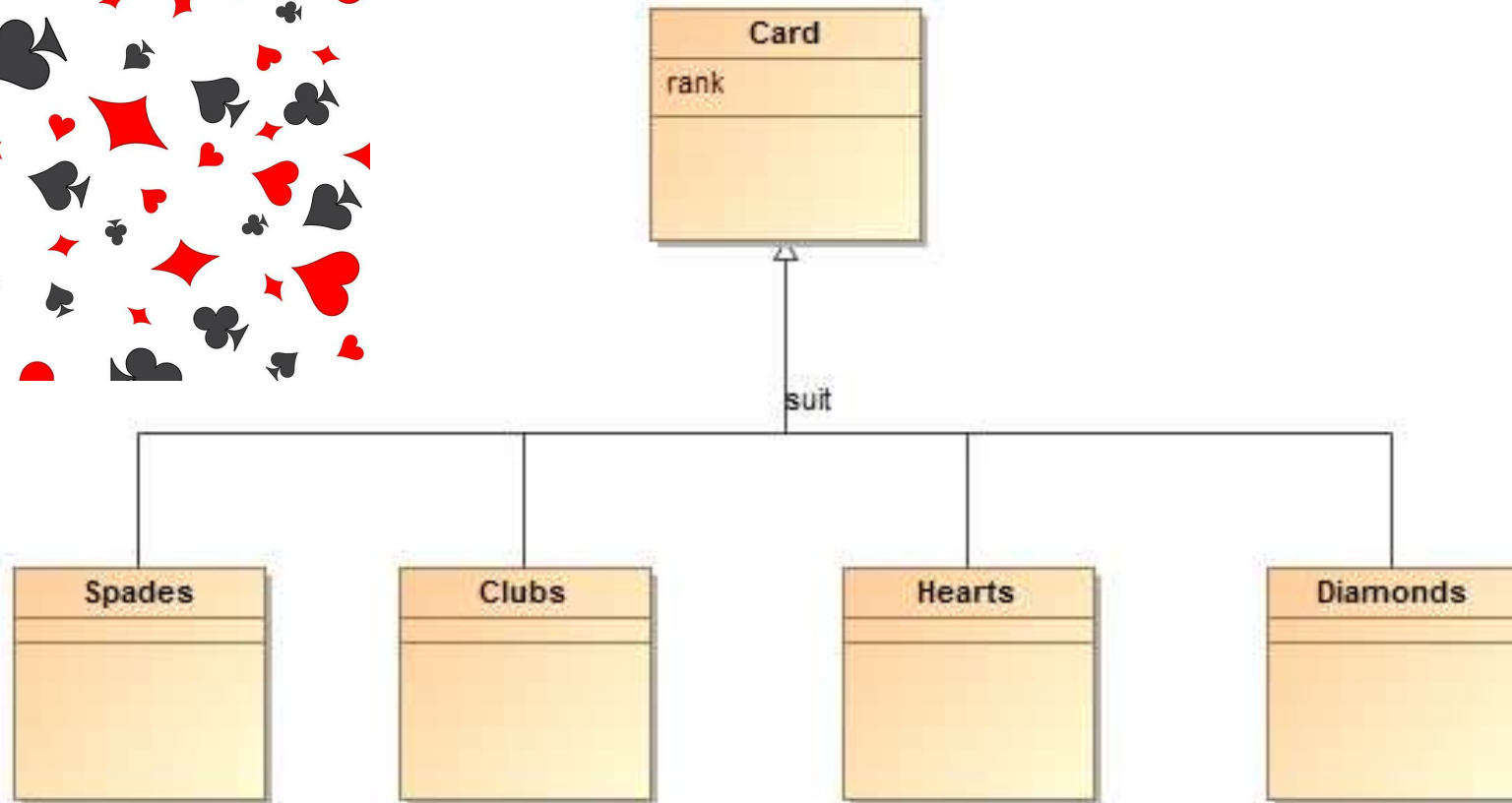
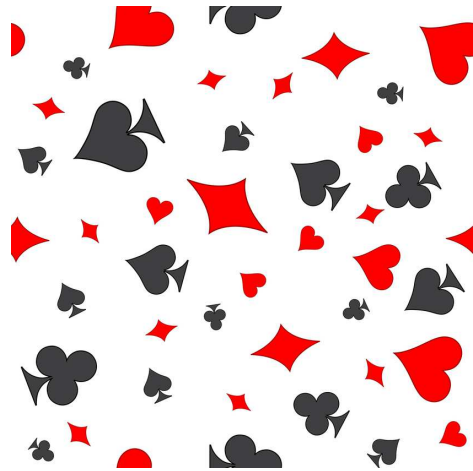
Exercise 4

In an online system to enroll students in University courses, each course has a unique course ID, a title, a start date and a duration. Each student has a unique student ID, a name and a birth date. A student can be enrolled in different courses and a course can be attended by several students. Each course is taught by only one professor who has a name and can teach several courses.

Enumerations

- ▶ An enumeration is a data type that has a finite set of values.
- ▶ Enumeration is a data type: you can declare an enumeration by listing the keyword enumeration in angle quotes (<< >>) above the enumeration name in the top section of a box. The second section lists the enumeration values.
- ▶ Do not use generalization to capture the values of an enumerated attribute:
 - ▶ An enumeration is a list of values.
 - ▶ Introduce generalization only when at least one subclass has significant attributes, operations, or associations that do not apply to the superclass.

Enumerations



Enumerations



Enumerations



Monday, Tuesday, Wednesday,...

January, February, March,...

Spring, Summer,...

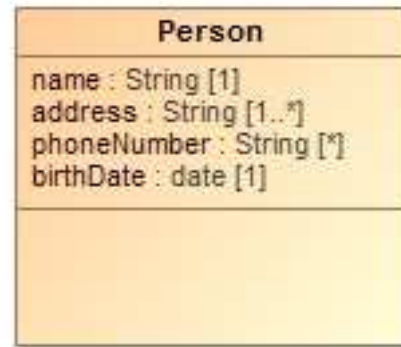
Question 2: Enumeration?

- ▶ A paper reviewing system has several conferences. Each conference has a title and a year and is managed by a chair and a list of committee members. Committee members and chairs must be assigned to one, but possibly more conferences. They have a name and an affiliation. A conference has several submitted papers, but a paper can be submitted to only one conference. A paper is assigned to 3 reviewers taken from the committee members. A paper can be accepted rejected or under review. We also know the paper titles and list of authors with their names and affiliations.

Exercise 5

In a system for handling shipments of products in an online bookshop, there can be three different types of items: book, greeting card, stationery item. A book has a title and a list of authors with names. A greeting card has a brand. A stationery item can be a pen, a pencil or a notebook. Each shipment has a priority that can be standard, high, or express.

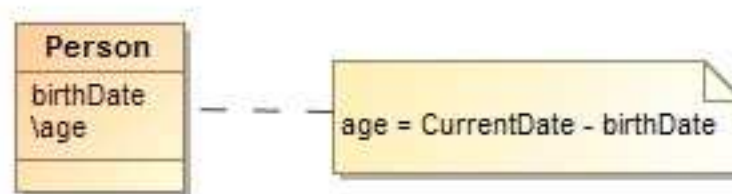
Multiplicity for attributes



- ▶ You can specify if an attribute is single or multivalued, mandatory or optional.

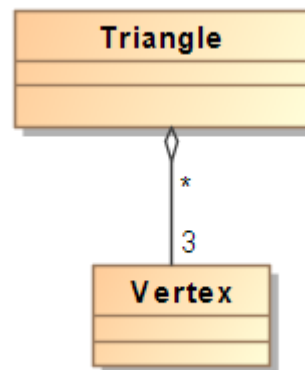
Derived data

- ▶ A derived element is a function of one or more elements, which in turn can be derived.
- ▶ The notation for a derived element is a slash in front of the element name.
- ▶ The constraint that determines the derivation must be shown.



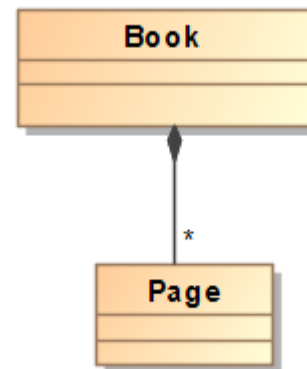
Aggregation

- ▶ Aggregation is a special form of association.
 - ▶ Underlines the fact that an object is made of constituent parts.
- ▶ The UML notation for aggregation is like the one for association with a small diamond indicating the assembly end.



Composition

- ▶ Composition is a more restrictive form of aggregation.
 - ▶ Two additional constraints:
 - ▶ A constituent part can belong to at most one assembly.
 - ▶ The part has a coincident lifetime as the assembly.
- ▶ The UML notation for composition is a small solid diamond next to the assembly class.



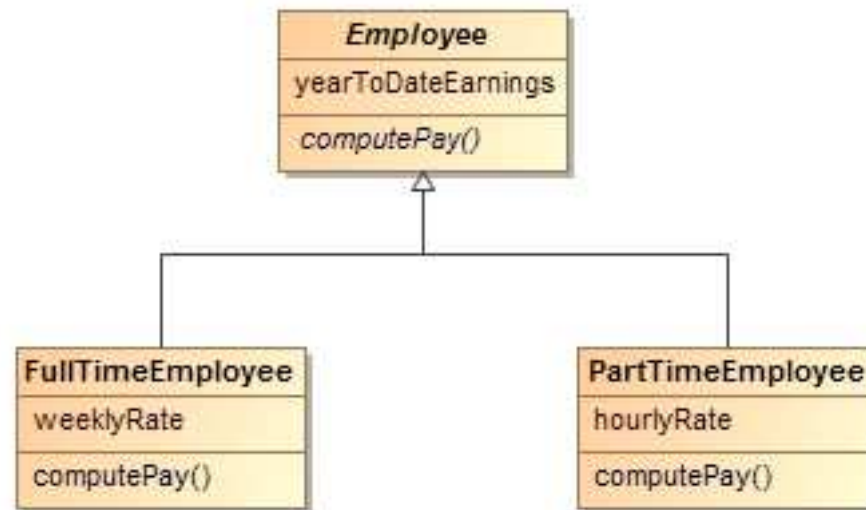
Exercise 6

In a system for managing salaries of people in companies, each company is divided in different divisions and each division in different departments. Each department is composed of a group of people and a location. Each person has a name a role and a salary. Each location has an address and a floor number.

Abstract classes

- ▶ An abstract class is a class that has no direct instances but whose descendants classes have direct instances.
- ▶ A concrete class is a class that is instantiable.
- ▶ A concrete class may have abstract subclasses, but they in turn must have concrete descendants: only concrete classes can be leaf classes in an inheritance tree.
- ▶ In the UML notation an abstract class name is listed in an italic font (or using {abstract} near the class name) .

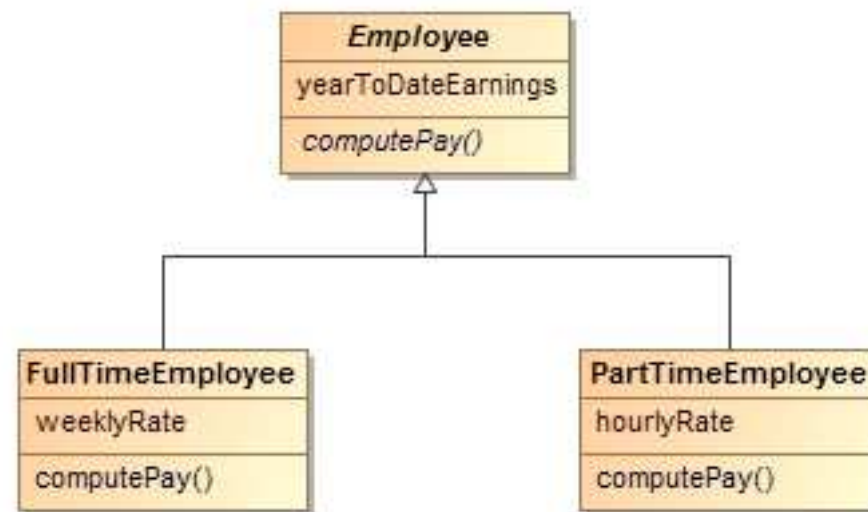
Abstract classes



Abstract operations

- ▶ Abstract classes can be used to define methods that can be inherited by subclasses.
- ▶ Abstract classes can define the signature of an operation without supplying a corresponding method.
- ▶ Abstract operations:
 - ▶ An abstract operation defines the signature of an operation for which each concrete subclass must provide its own implementation.
 - ▶ An abstract operation is designated by italics or the keyword `{abstract}`.

Abstract operations

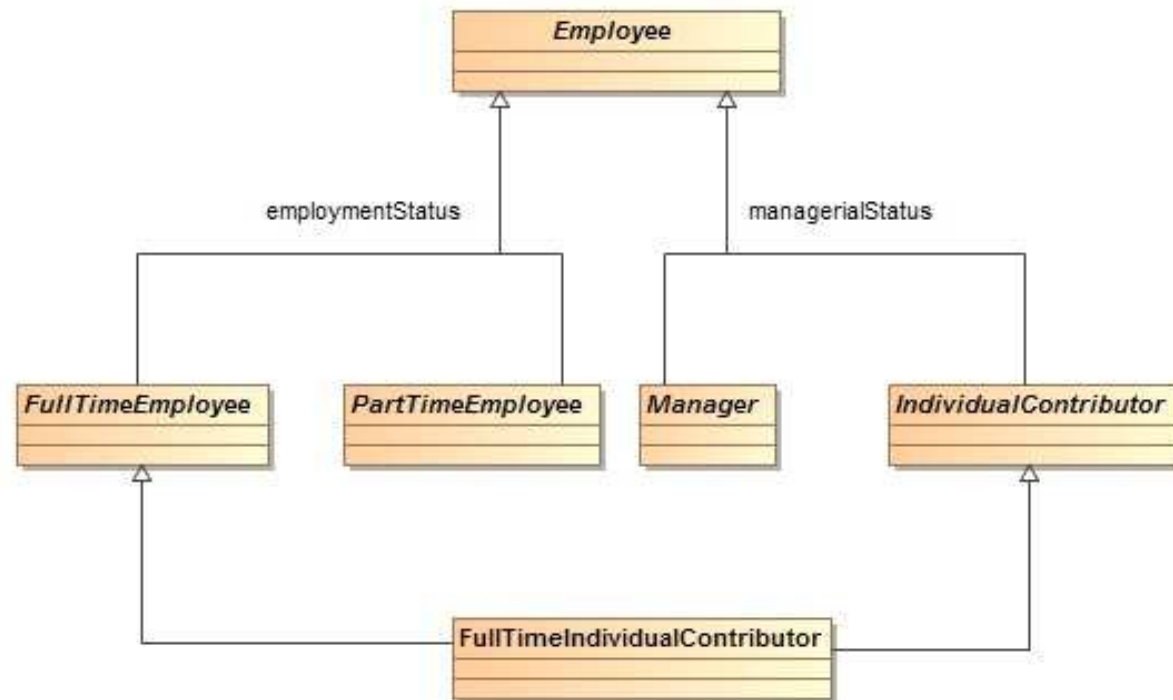


Exercise 7

Prepare a class diagram for a graphical document editor that supports grouping. Assume that a document consists of several sheets. Each sheet contains drawing objects, including text, geometrical objects, and groups. A group is simply a set of drawing objects, possibly including other groups. A group must contain at least 2 drawing objects. A drawing object can be direct member of at most one group. Geometrical objects include circles, ellipses, rectangles, lines and squares.

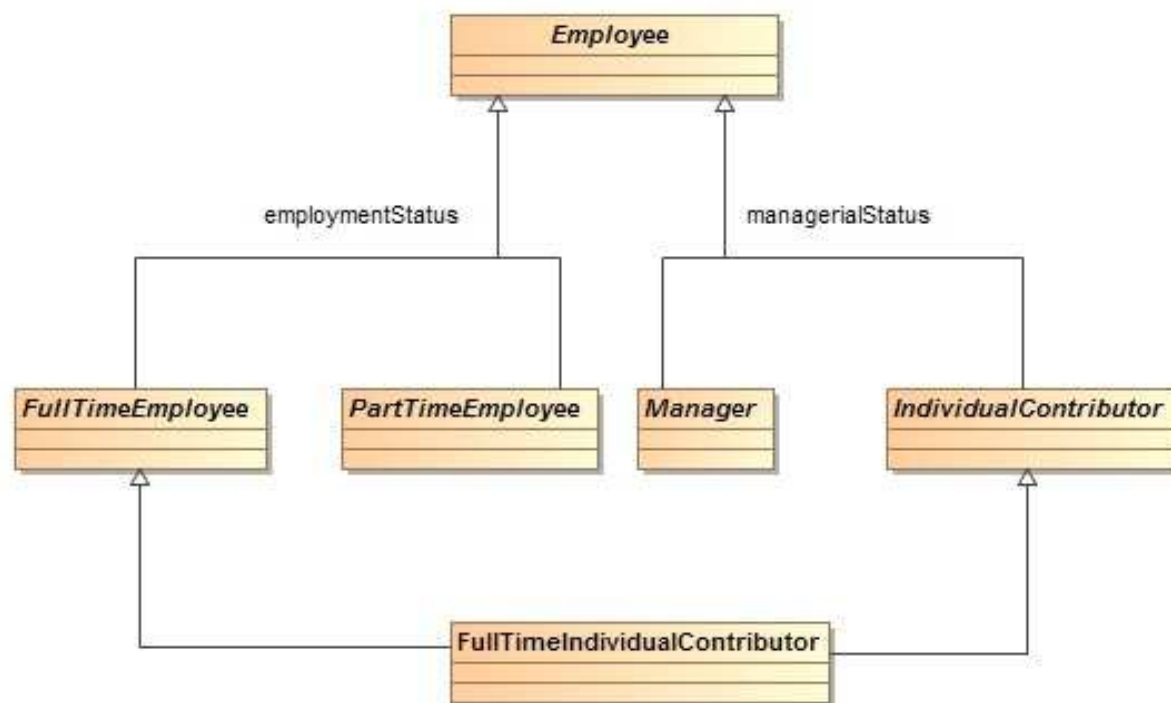
Multiple Inheritance

- ▶ Multiple inheritance permits a class to have more than one superclass and to inherit features from all parents.
- ▶ The most common form of multiple inheritance is from sets of disjoint classes.



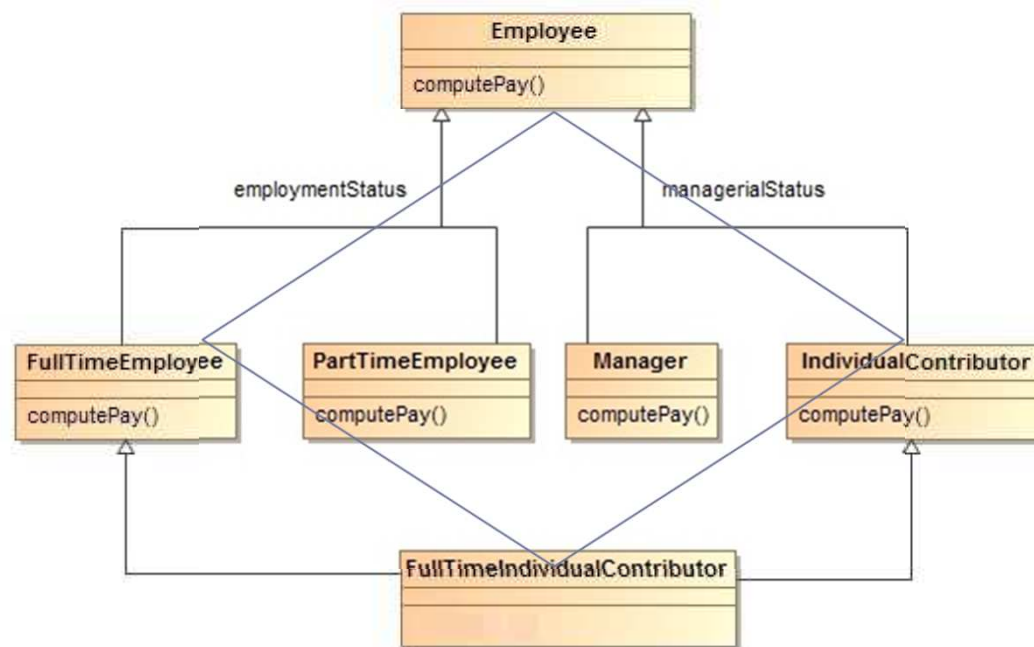
Multiple Inheritance

- ▶ A subclass inherits a feature from the same ancestor class found along more than one path only once.
- ▶ *FullTimeIndividualContributor* inherits *Employee* features along two paths but it has only a single copy of them.



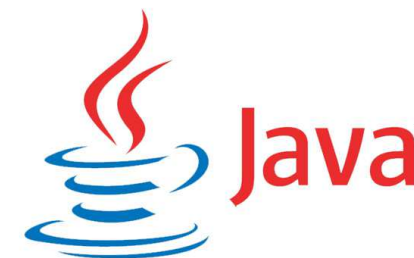
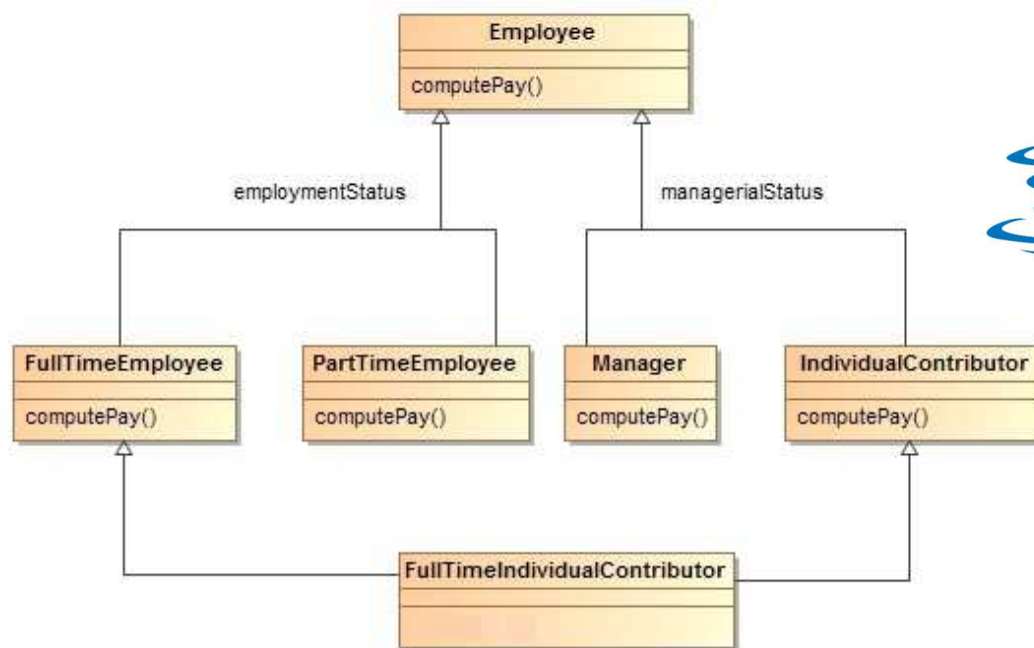
Multiple Inheritance

- ▶ Conflicts among parallel definitions create ambiguities that implementations must resolve.
 - ▶ Diamond problem: which version of *computePay()* should be used in *FullTimeIndividualContributor*?



Multiple Inheritance

- ▶ Conflicts among parallel definitions create ambiguities that implementations must resolve.
 - ▶ Diamond problem: which version of *computePay()* should be used in *FullTimeIndividualContributor*?



Question 3

In MagicDraw a package contains a set of classes. Look at how the editor works. Would you represent a package as an aggregation of classes or as a composition of classes?

How to model

- I. A country has a capital.

How to model

1. A country has a capital.
2. A file is an ordinary file or a directory file.

How to model

1. A country has a capital.
2. A file is an ordinary file or a directory file.
3. A file is an ordinary file or a directory file. A directory file contains 0 or more files.

How to model

1. A country has a capital.
2. A file is an ordinary file or a directory file.
3. A file is an ordinary file or a directory file. A directory file contains 0 or more files.
4. A programmer participates in a project and can program using only one programming language

How to model

1. A country has a capital.
2. A file is an ordinary file or a directory file.
3. A file is an ordinary file or a directory file. A directory file contains 0 or more files.
4. A programmer participates in a project and can program using only one programming language
5. A programmer participates in a project and can program using different programming languages

How to model

1. A country has a capital.
2. A file is an ordinary file or a directory file.
3. A file is an ordinary file or a directory file. A directory file contains 0 or more files.
4. A programmer participates in a project and can program using only one programming language
5. A programmer participates in a project and can program using different programming languages
6. A file contains records and metadata

Exercise 8

The customers use banking application of a bank to manage current and saving accounts of customers. Each account has an account number and an opening date. We assume that a customer at least has a current account. No customer can have more than one account of the same type. The bank also has a manager who has administrator privileges. In the application, a customer or a manager is represented by a user profile that contains the name, date of birth, username, password and role. The role can be either customer or manager. A manager cannot have any account.

Exercise 9 (Domain Modelling)

Chapter 12 of the text book



An institution may issue many credit card accounts, each identified by an account number. Each account has a minimum credit limit and a current balance. The account serves one or more customers. Each month, the institution issues a statement for each account. The statement indicates the reference month (and year) and the total amount available at the end of the month. The statement also itemizes various transactions that have occurred throughout the billing interval: cash advances, interest charges, fees of type regional or national, and adjustments. The name of the merchant is printed for each purchase.

Exercise 10 (Domain Modelling)

Chapter 12 of the text book



A scheduling software must support the following functions: arranges meetings, schedules appointments, plans tasks, and tracks holidays. The scheduler runs on a network that many users share. Each user may have a schedule. A schedule contains multiple entries. Most entries belong to a single schedule (the schedule of the owner of the entry). However, a meeting entry also appears in the schedules of all the attendees. There are 4 types of entries: meetings, appointments, tasks and holidays. Meetings and appointments both occur in a single day and have a start time and an end time. In contrast, tasks and holidays may extend over several days and just have a start date and an end date. Any entry has a description. The scheduling software facilitates meetings. When a user (the chairperson) arranges a meeting, the software places a meeting entry in the schedule of each attendee. The chairperson uses the scheduler to reserve a room for the meeting, to identify the attendees, and to find time on their schedules when everyone is available. The chairperson can indicate whether the attendance for each attendee is required or optional. The system tracks the acceptance status for each attendee – whether an attendee has accepted or declined.