

Class Diagram

NOUR



Contents



1

Introduction

2

Representing a class

3

Classes Relationships

Introduction



- Class diagram is a way of showing system classes and the relations among them.
- Static diagrams, pick only existence and relationships but not actions.
- Where to get your classes?

Representing a class



- Class name
- Attributes
- Operations(methods)



Attributes



Syntax:

attributeName : type

Ex:

firstName : String
salary : Dollars



Methods



Syntax:

methodName(parm) : return type

Ex:

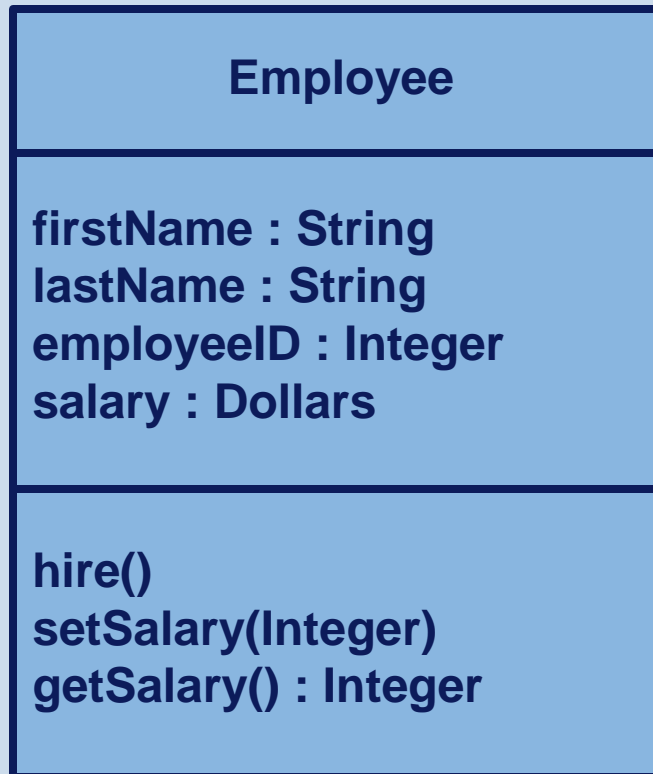
setName(String)
getSalary() : Dollars



Representing a class



- Complete Example

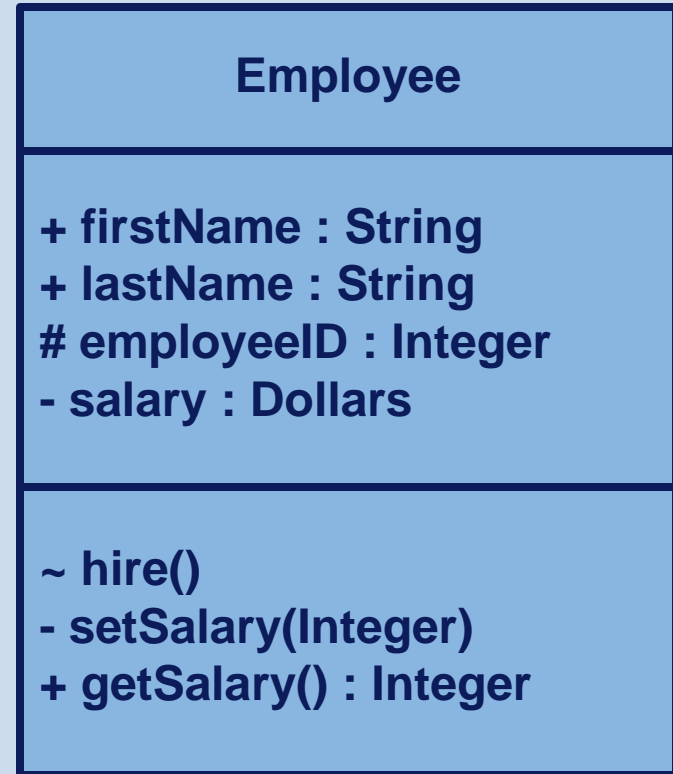


Visibility



❖ You can show the visibility of attributes and methods by these signs

+ : public
- : private
: protected
~ : package



Okay !



- ❖ We have learned how to create classes
- ❖ Now let's learn how to connect them !

Classes Relationships

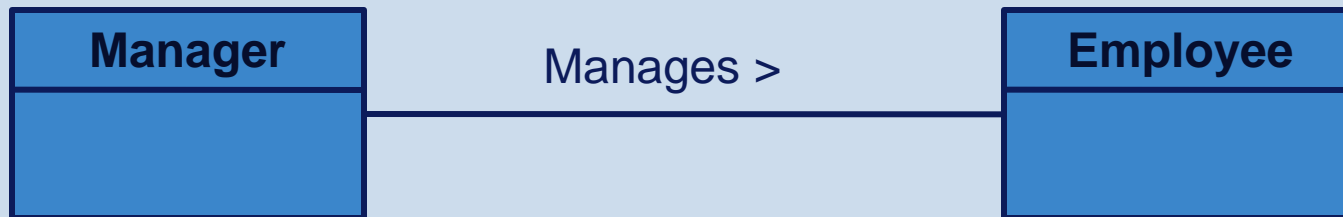


- ❖ Association
 - ❖ Aggregation
 - ❖ Composition
 - ❖ Generalization
- Multiplicity

Association



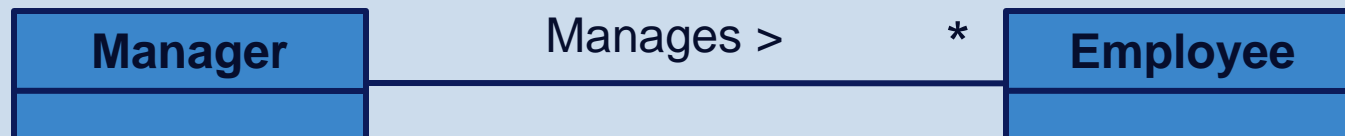
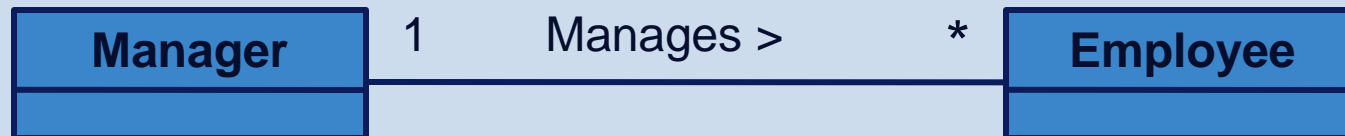
- If there is a conceptual connection between two classes we say that these classes are associated.



Multiplicity



❖ An indication of numerical limit of a relation.



Multiplicity



Aggregation & Composition



- Both describe the relation between a whole and its parts.

Band

Musician

Catalog

Product

Building

Room

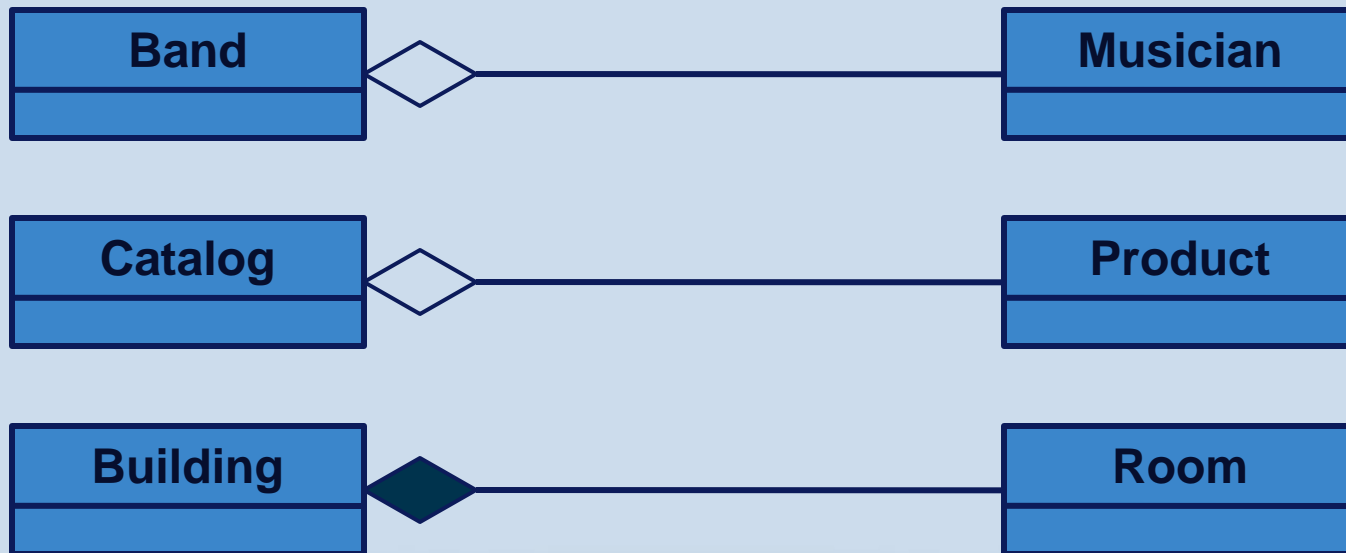
What is the difference ?!

Aggregation & Composition



- To test if you are dealing with aggregation or composition check :
- If the **part belong only to one whole** then its **composition**.

Classes Relationships



Aggregation & Composition



- Aggregation



- Composition

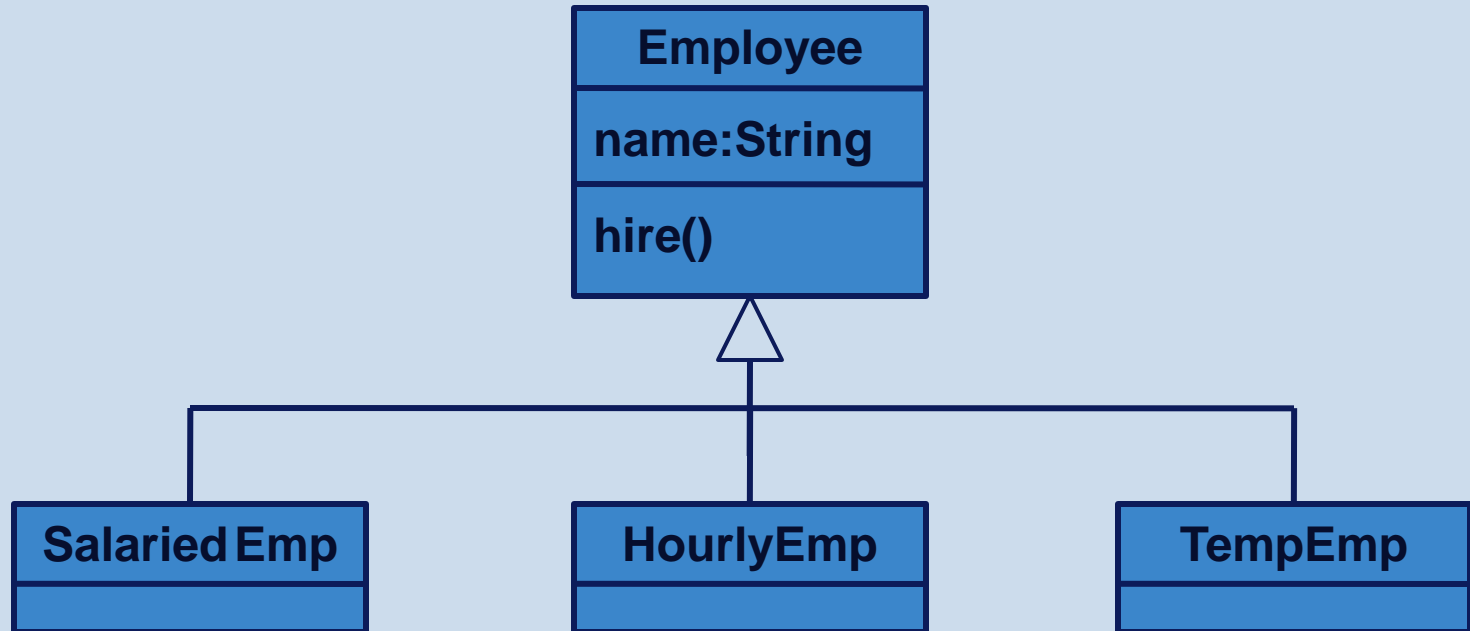


- The diamond sign is always at the side of the **whole**.

Generalization



- A way of describing inheritance in Class Diagram.



Thank You !

