

Class diagram

UML Class Diagram shows:

- **Classes** : the main building blocks.
- **Attributes** : data/properties of the class.
- **Methods/Operations** : methodes or functions the class can perform.
- **Relationships between classes** : association, inheritance, aggregation, ...etc.

Visibility Notation :

Symbol	Visibility	Meaning	Who can access	Used for
+	Public	Accessible everywhere	Everyone	Interface (visible to all)
#	Protected	Accessible in class and subclasses	Class + child classes	Inheritance (visible to subclasses)
-	Private	Accessible only inside the class	Class only	Encapsulation (data hiding)
~	Package	is a container used to group related classes ,	visible to classes in the same package	

N.B : A package in UML is a container used to group related classes, interfaces, use cases, or even other packages. It's like a folder in your computer — it helps keep the model organized and modular.

Relationships notations :

Relationship Type	UML Symbol	Meaning
Association	————	Simple connection
Multiplicity	1, 0..*, *, ...etc	Number of instances
Generalization (Inheritance)	————>	subclass inherits superclass
Aggregation	◊————	“has-a”: weak relationship
Composition	◆————	“contains” ,Strong “part-of” relationship (part dies with whole)
Dependence	- - - - ->	temporary use (one class changes due to the change in another class)

Inheritance constraints :

Constraint	Description
{Disjoint}	An object (instance) can belong to only one subclass.
{Overlapping}	An object (instance) can belong to more than one subclass.
{Complete}	All subclasses are known. All possible subclasses are defined in the diagram (no more can exist).
{Incomplete}	More subclasses may exist. More subclasses may exist that are not shown in the diagram

Aggregation vs Composition :

- ✓ **Aggregation** (a “has-a” relationship) : it is a weak relationship, it means one class contains or uses another class, but the contained object can exist independently. (Lifetime Dependency: Independent).

Example : A **Department** has **Professors**. :If the department is deleted, the professors still exist (they can belong to another department).

- ✓ **Composition** (“part-of” relationship) : is a strong form of aggregation (strong relationship). It means one class owns another class, the part cannot exist without the whole(Lifetime Dependency: dependent).

Example: A **House** has **Rooms**. If the house is destroyed, the rooms also disappear.

Relation	Example	What happens if the whole is destroyed?
Association	Driver – Car	Both still exist separately
Aggregation	Department – Professor	Professors still exist
Composition	House – Room	Rooms are destroyed too