UML introduction (class diagrams)

UML - unified modeling language applicable in the field of software engineering at various stages such as concept, specification and implementation.

It provides a few types of diagrams: static, dynamic and physical.

In this course, however, we will talk a little bit about static diagrams, namely class diagrams. It should help us in explaining and understanding SOLID principles.

UML introduction (class diagrams)

Why learning a bit of UML?

It is the standardized notation that:

- Specifies software systems
- Documents design and architecture decisions
- ***Helps communicate clearly in different contexts***

In practice, only subset is actually needed and not really often, which depends on a specific job performed.

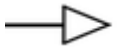
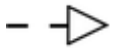
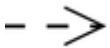
UML introduction (class diagrams)

Class diagrams

Basically these diagrams consist of rectangles, representing classes, and arrows, representing relationship between classes.

Classes have attributes (data members) and operations (member functions) with access identifiers such as public (+), private (-) and protected (#).

Relationships are depicted as follows:

- Association 
- Inheritance 
- Realization / implementation 
- Dependency 
- Aggregation 
- Composition