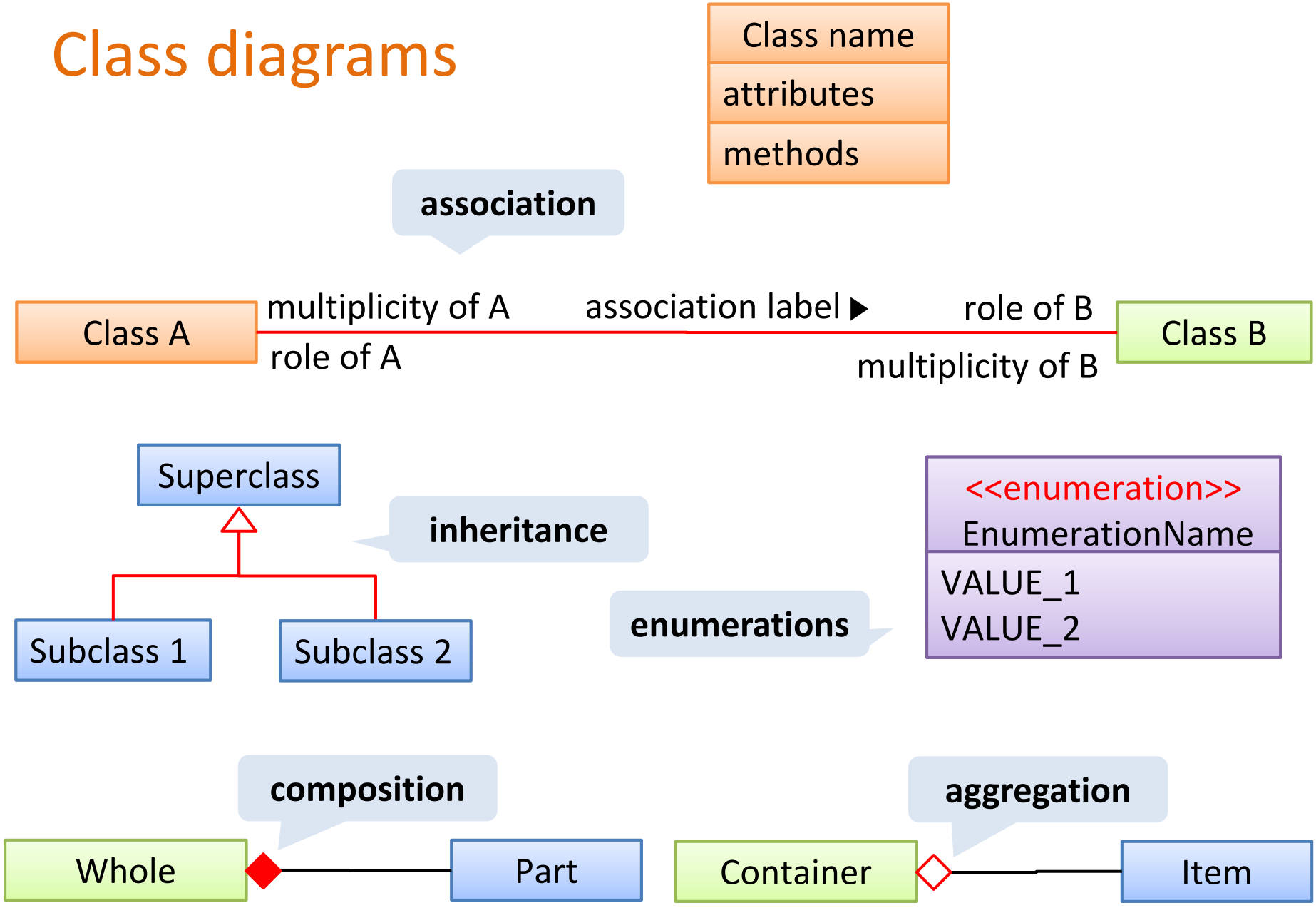
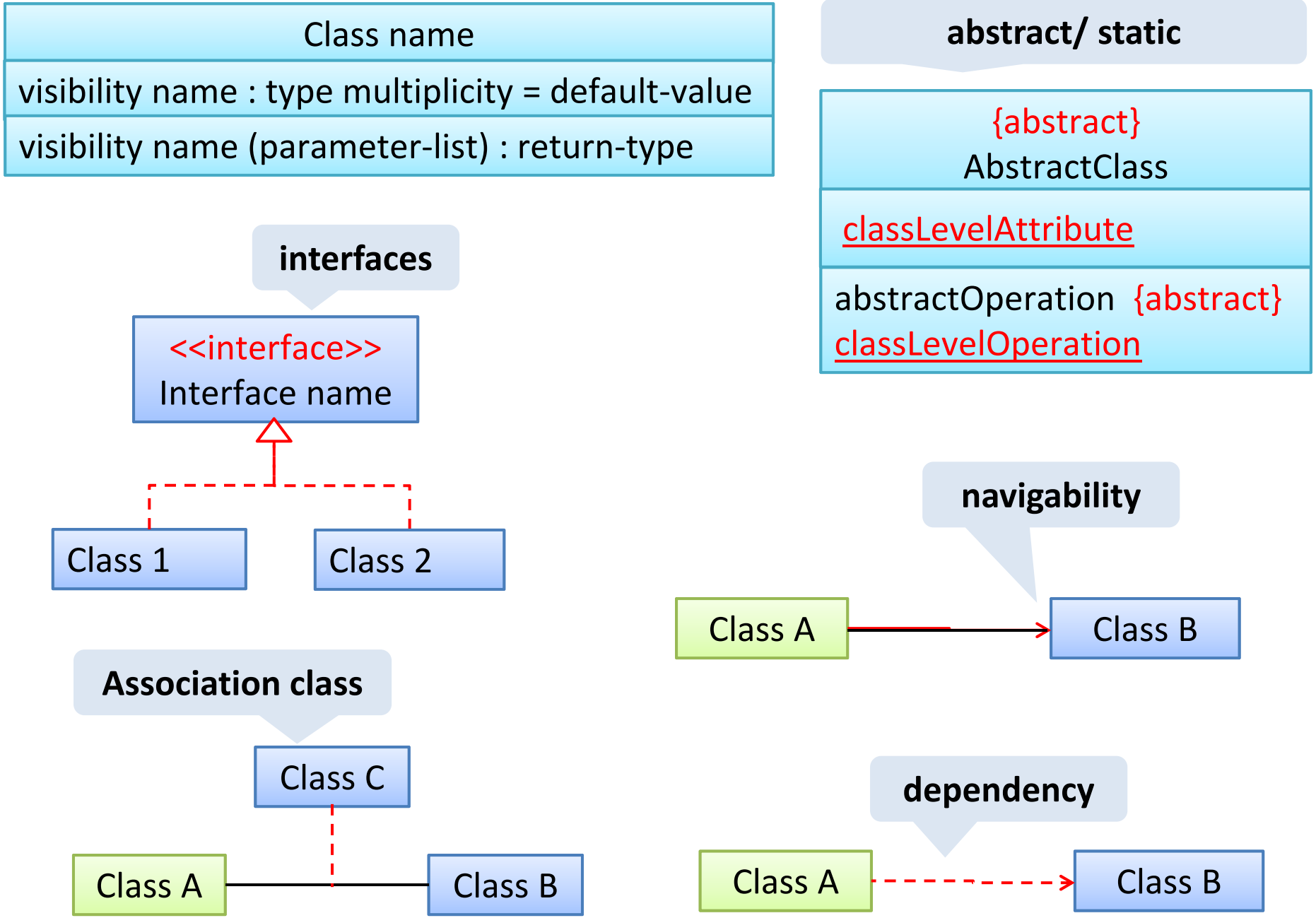
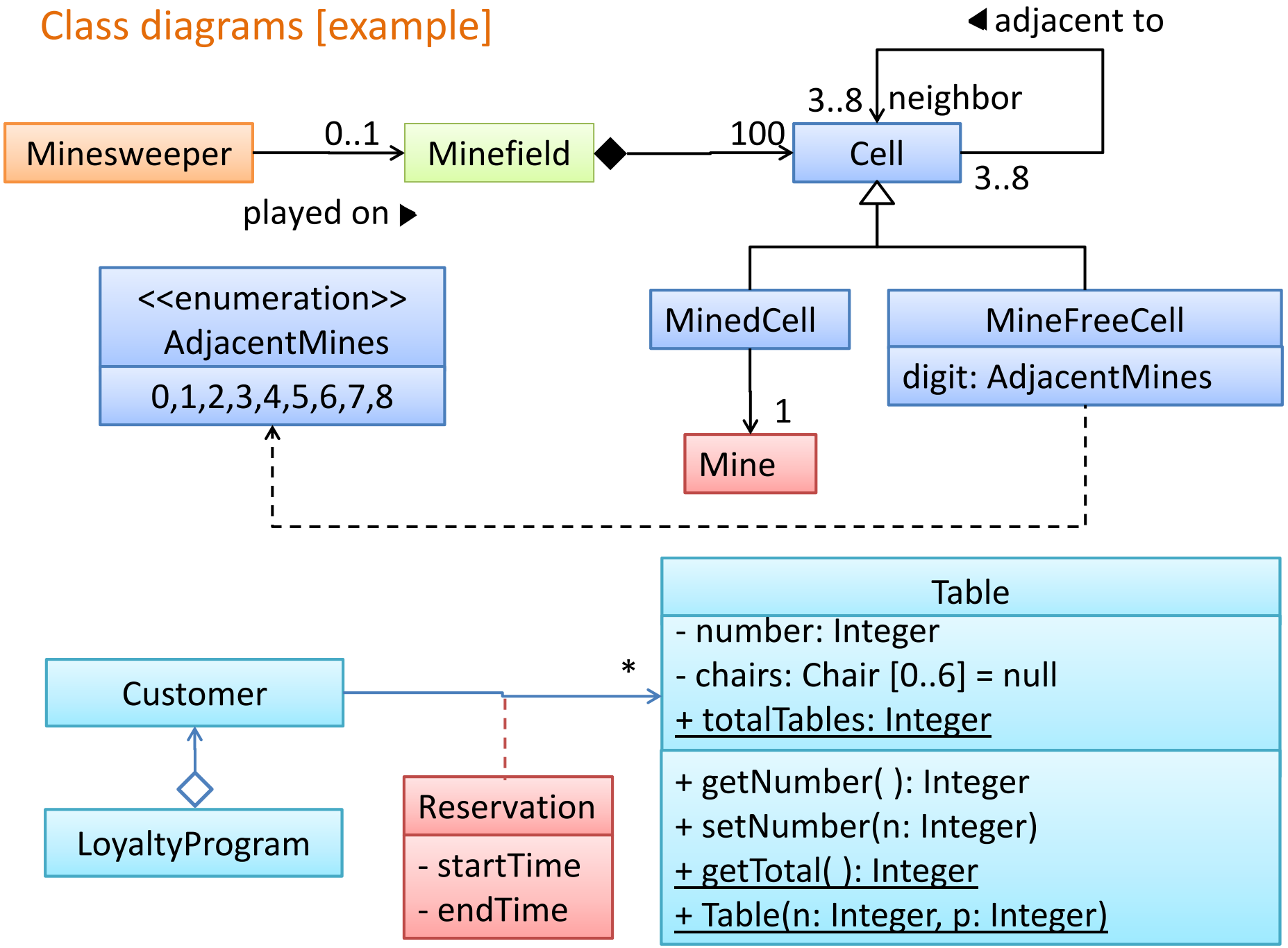
Class diagrams

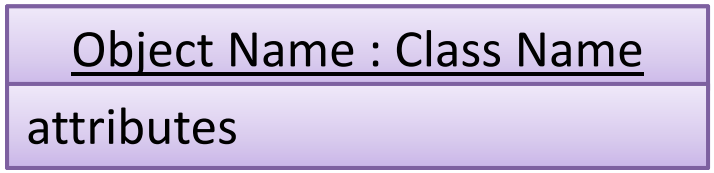



Class name
visibility name : type multiplicity = default-value
visibility name (parameter-list) : return-type
abstract/ static
{abstract}
AbstractClass
classLevelAttribute
abstractOperation {abstract}
classLevelOperation
interfaces
<<interface>>
Interface name
Class 1
Class 2
navigability
Class A
Class B
Association class
Class C
Class A
Class B
dependency
Class A
Class B

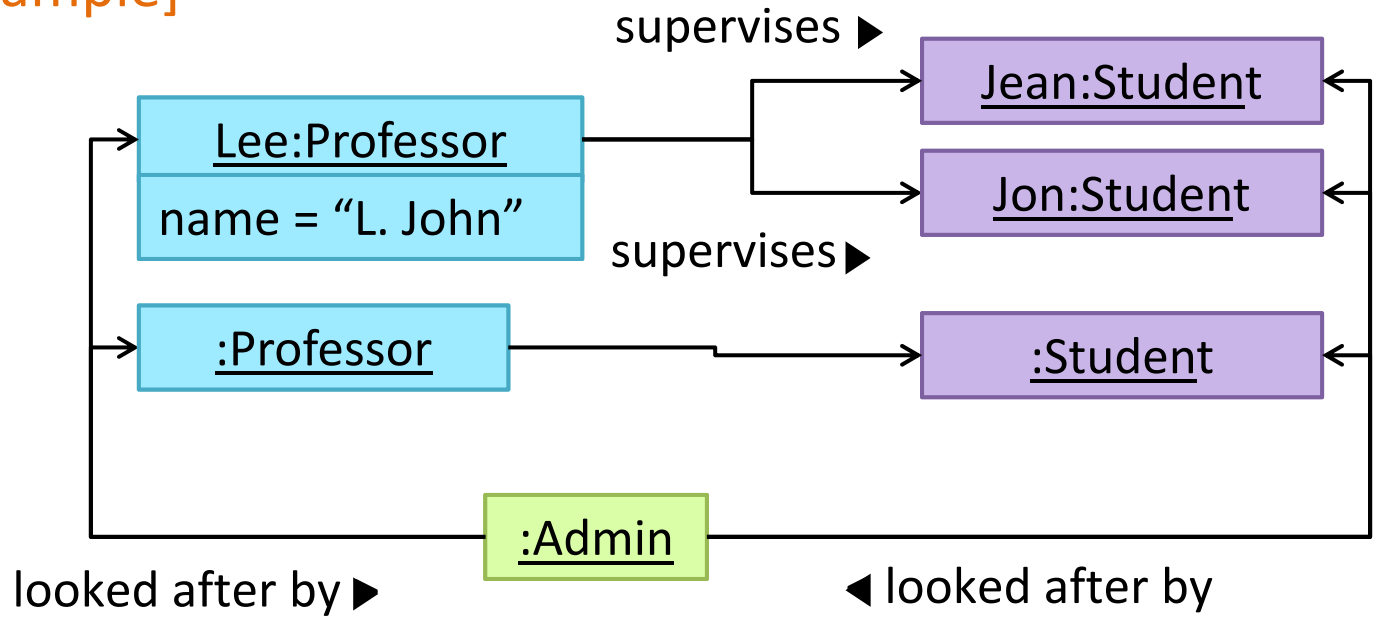
Class diagrams [example]



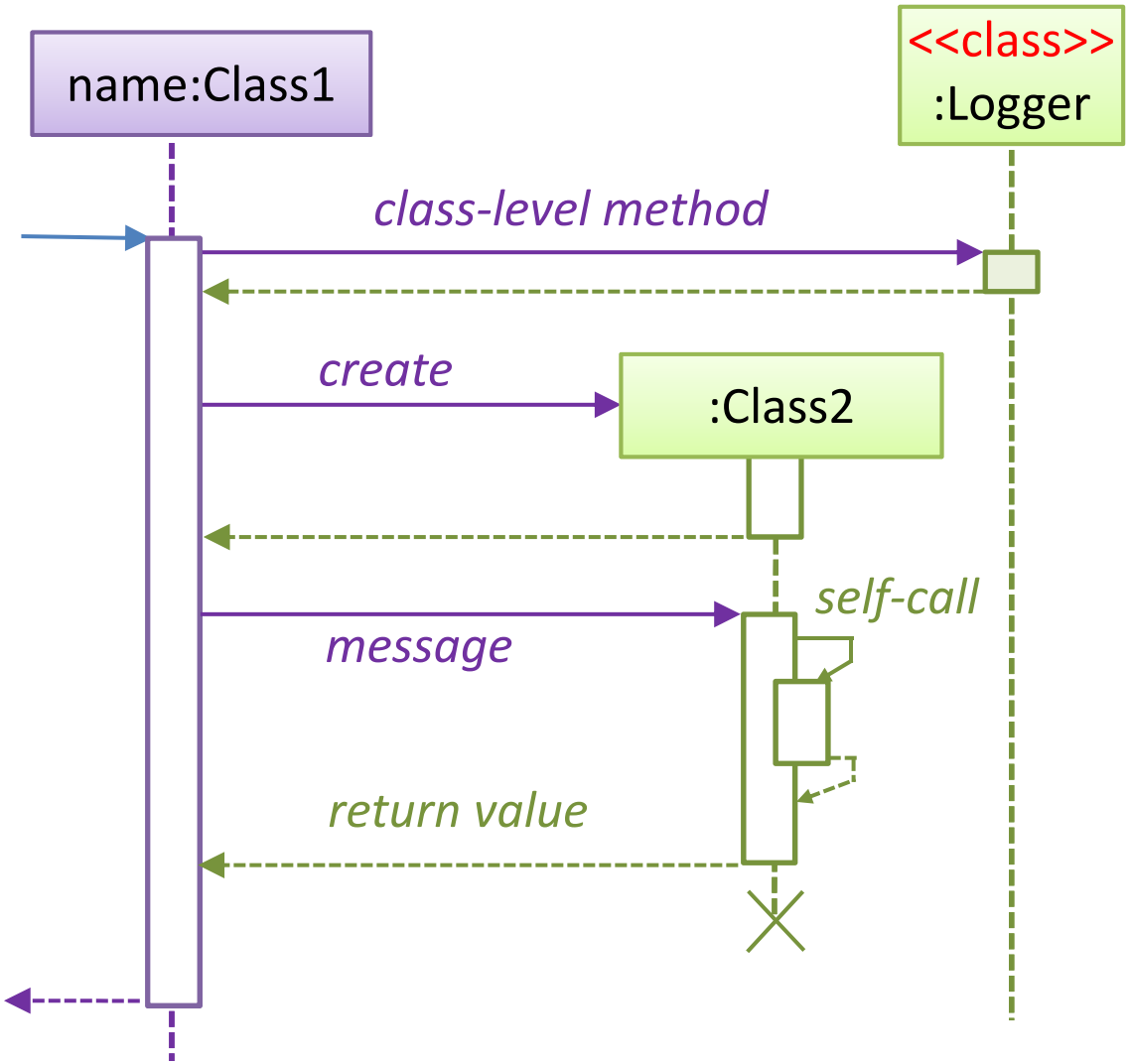
Object diagrams

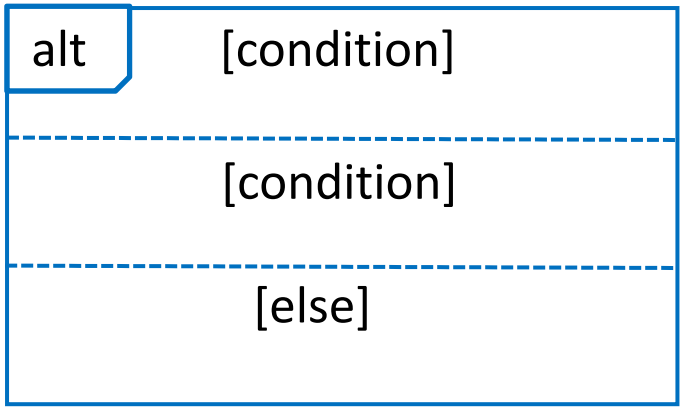


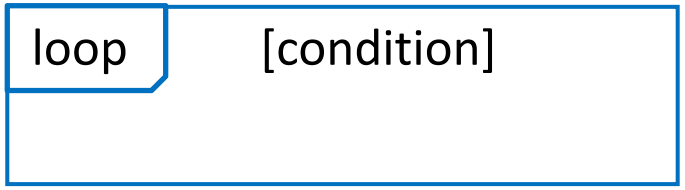
[example]

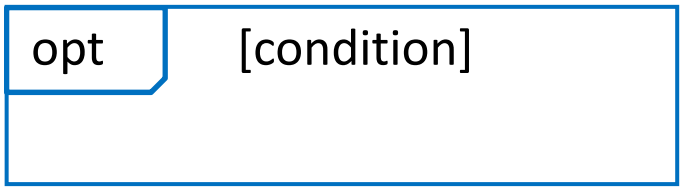


Sequence diagrams

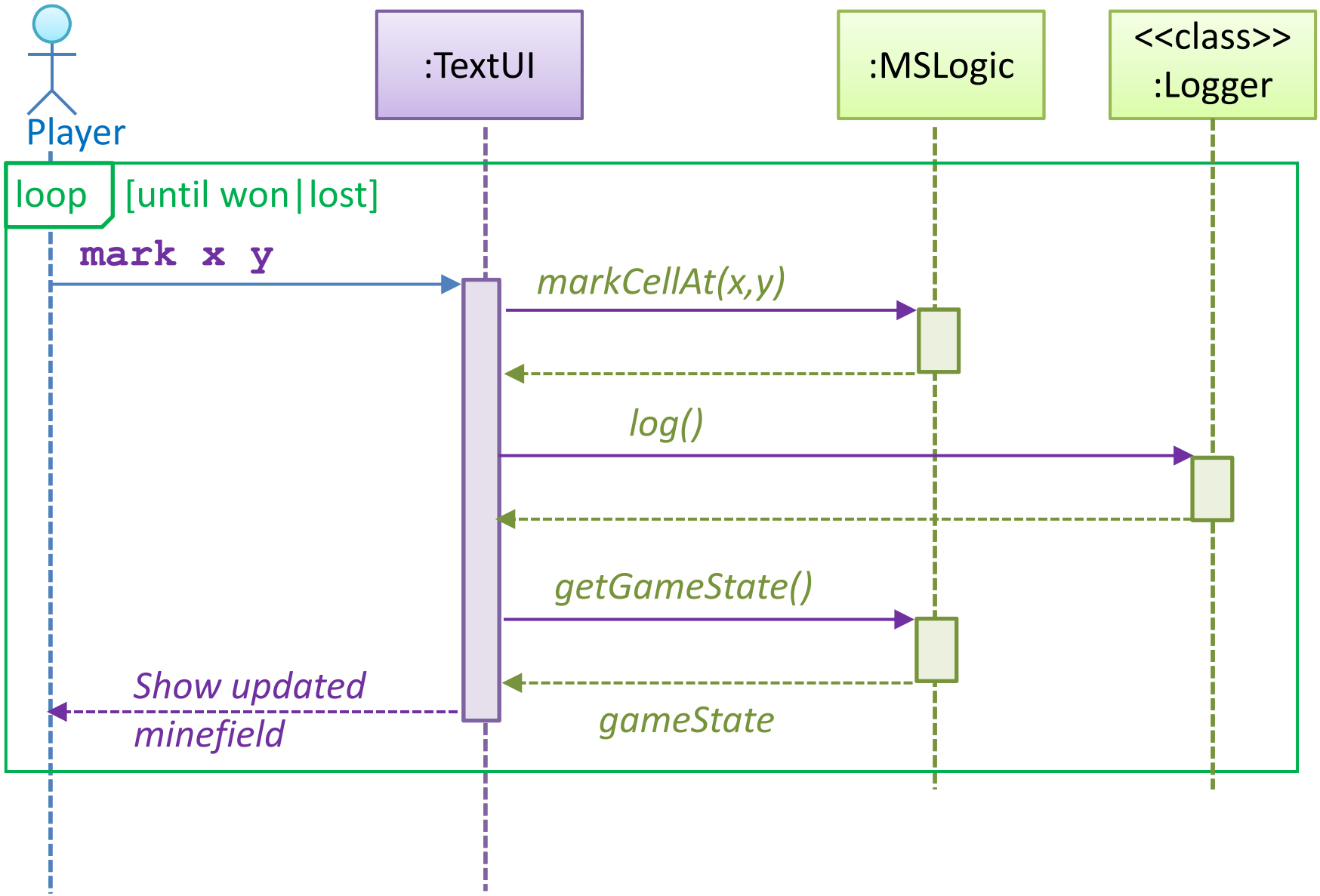




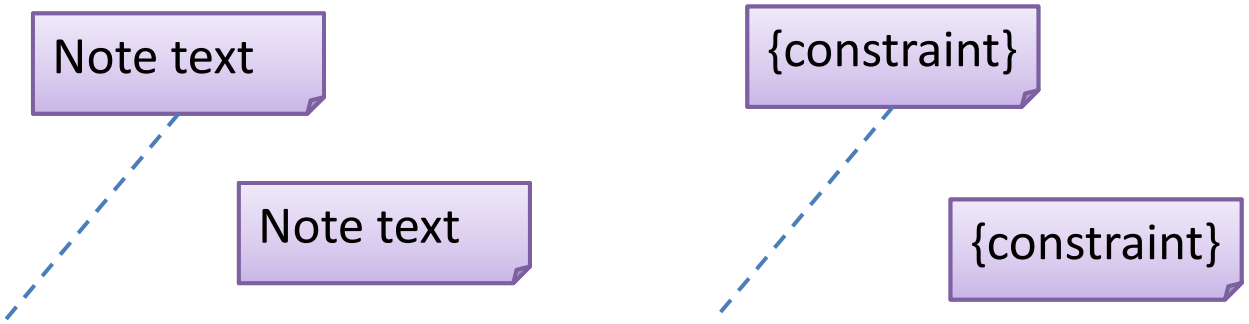




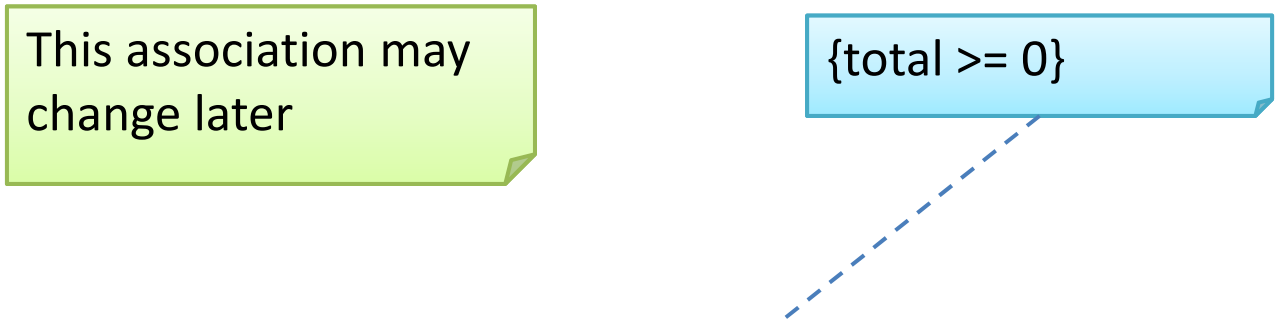
Sequence diagrams [example]



Notes and constraints

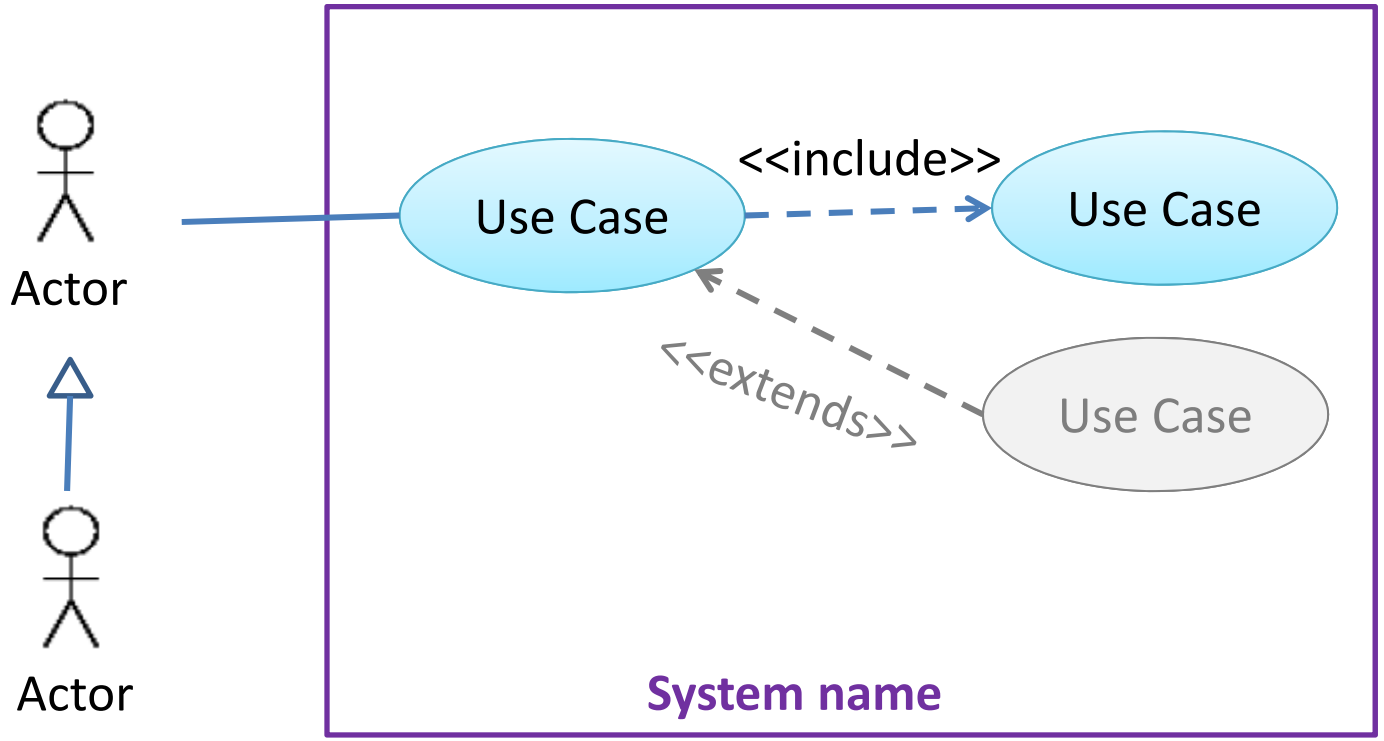


[examples]

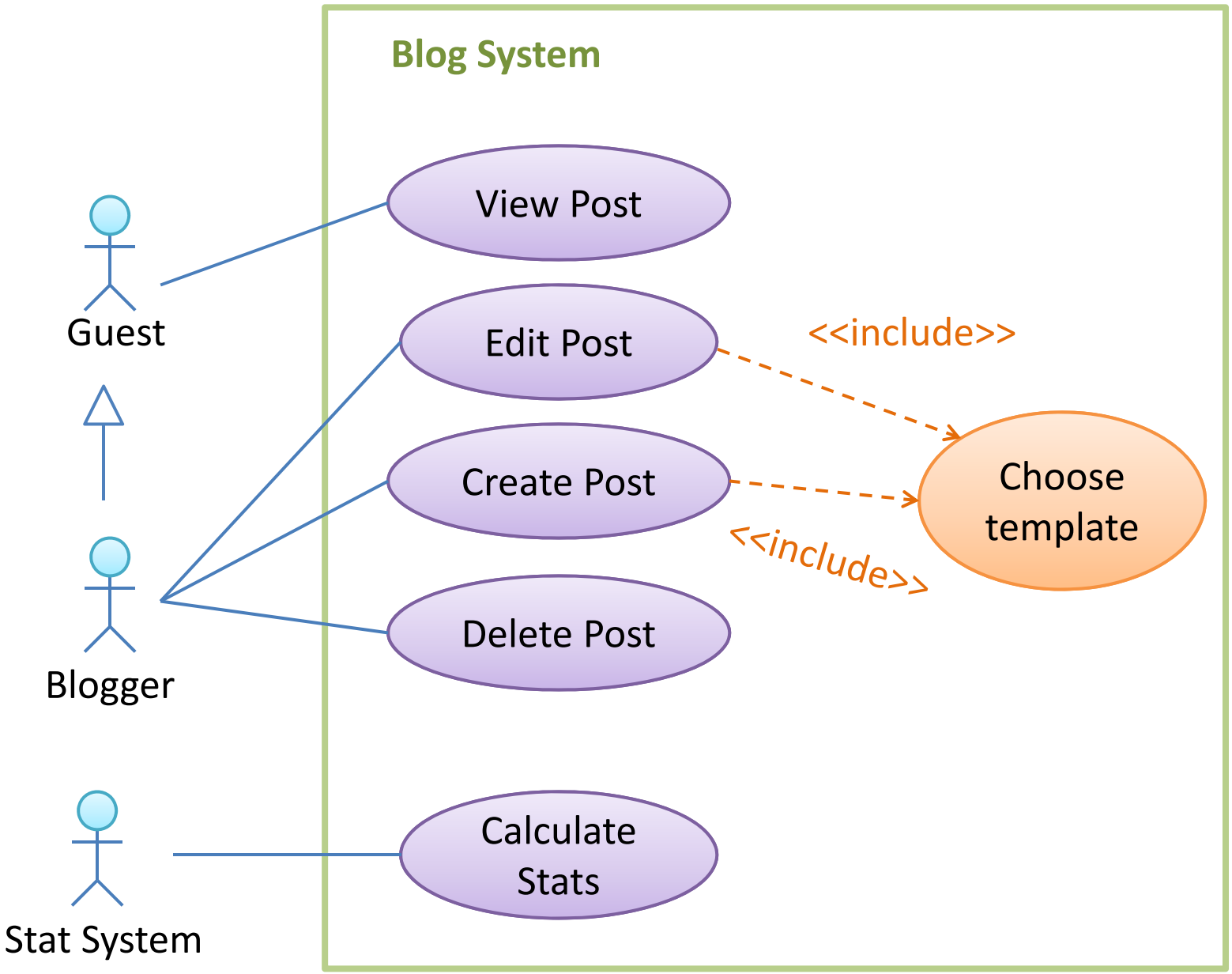


NOTE: Use case diagram and Activity diagrams are optional topics and for your own learning

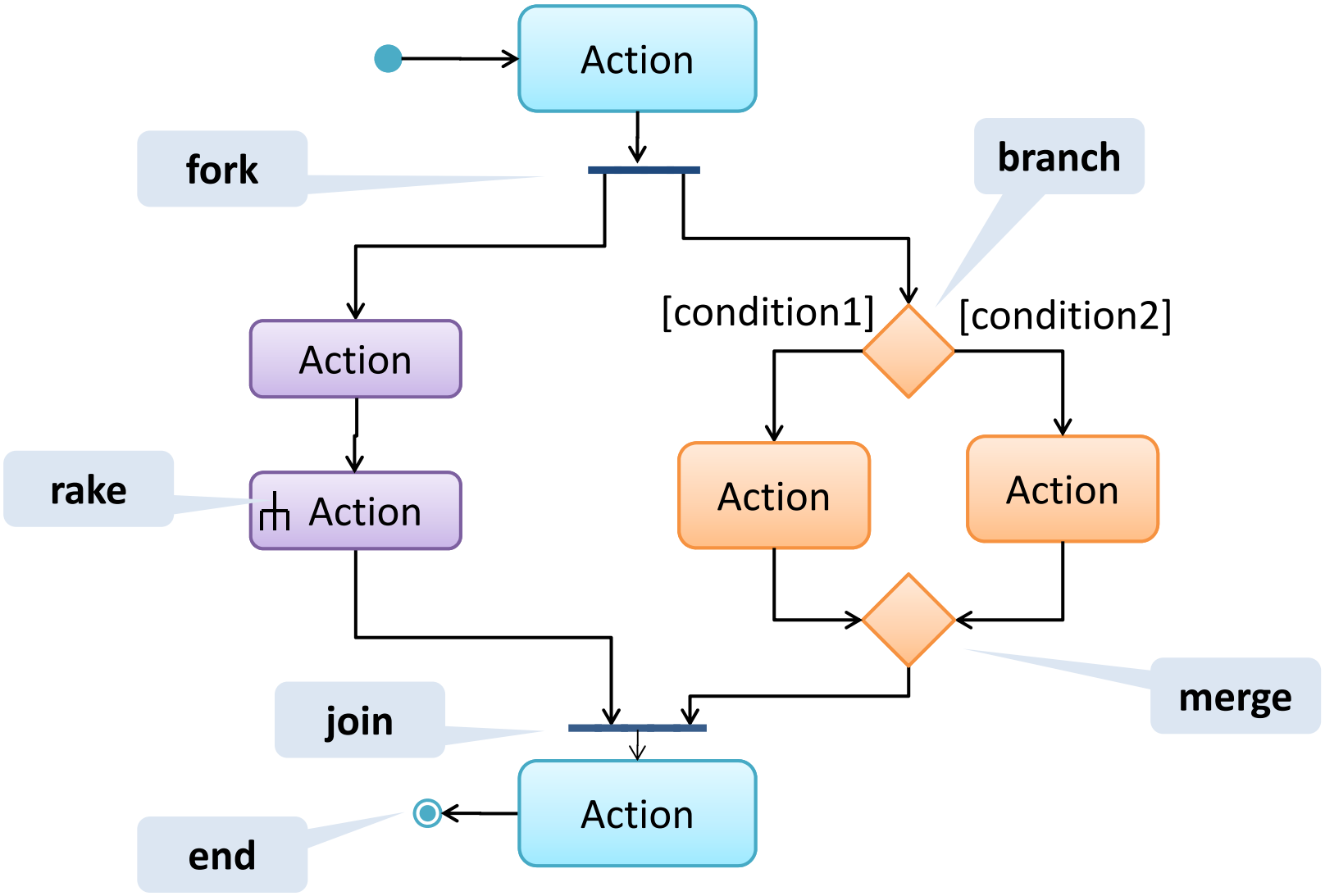
Use case diagrams



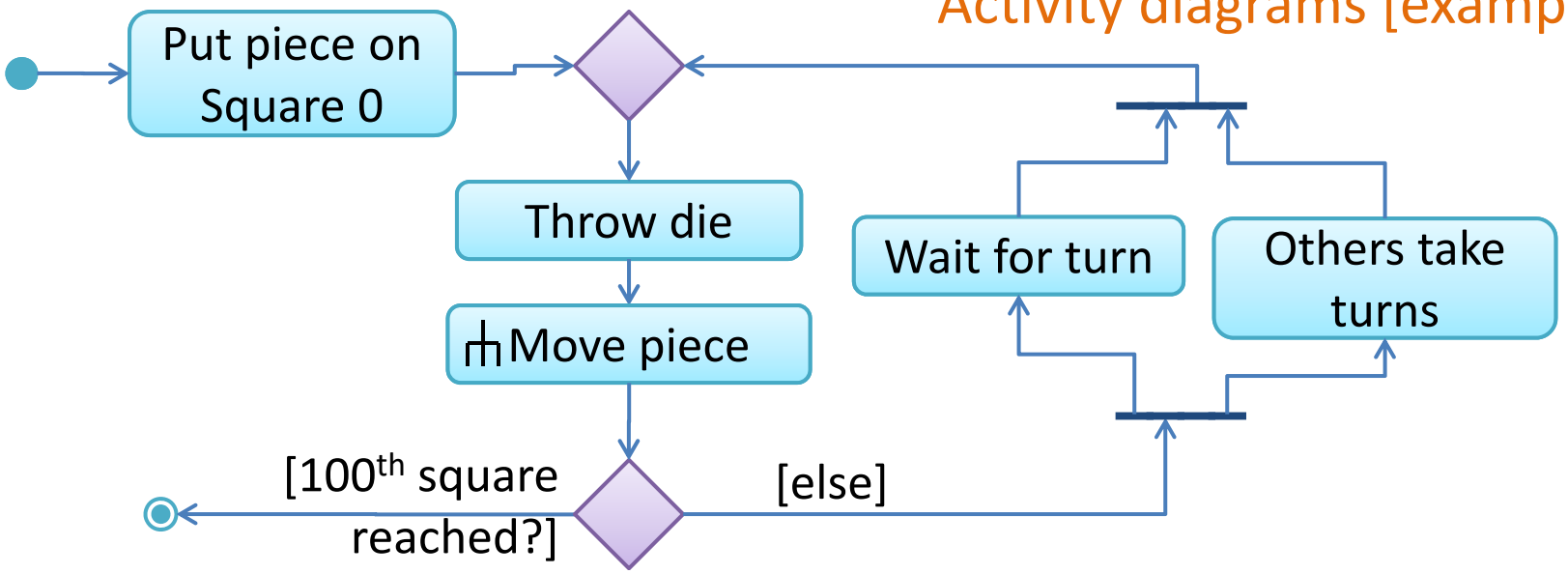
Use case diagrams [example]



Activity diagrams



Activity diagrams [example]



Activity: Move piece