



Title: A class UML diagram for read-and-sort application
Function: Read scanned integers from user and sort them in a Stack from smallest to largest
Author: Huihua Huang
Date: 06/12/2025
Description:

1. Dependency:

1) **SorterFactory** and **Message**:
SorterFactory depends on the Message class to display warnings when the input sorter type is invalid.

2) **ReadAndSortApp** and **Message**:
- ReadAndSortApp depends on the Message class to display warnings while user input is empty in askInput method.
- ReadAndSortApp depends on the Message class to display warnings, successes and sorted result lists in printResult method.

2. Implementation:
The LinkedListSorter class implements and specifies the sort method from the interface CollectionSorter.
The StackSorter class implements and specifies the sort method from the interface CollectionSorter. (newly added for project 5)

3. One-to-One Composition:
An instance of the LinkedListSorter class is created in the SorterFactory class when the sorter type is LinkedList or a fallback situation occurs.
An instance of the StackSorter class is created in the SorterFactory class when the sorter type is Stack. (newly added for project 5)

4. One-to-One Aggregation:

1) **SorterFactory** and **ReadAndSortApp**:
An object of the SorterFactory class is passed to call run method of a ReadAndSortApp object.

2) **StringToListParser** and **ReadAndSortApp**:
An object of the StringToListParser class is passed to call run method of a ReadAndSortApp object.

Legend for Relation	
	Composition
	Aggregation
	Implementation
	Dependency