

# Диаграми на взаимодействието

---

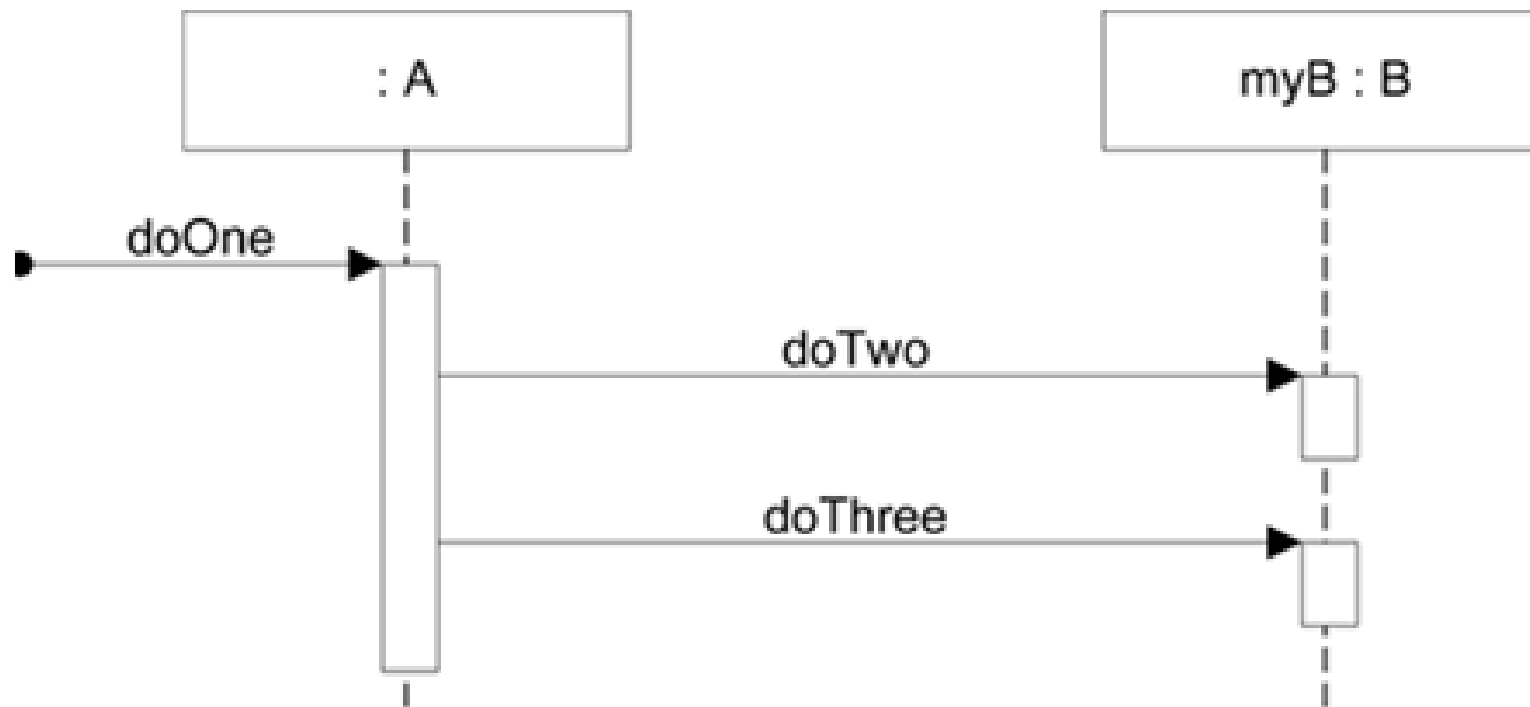
# ОСНОВНИ ТЕМИ

---

- Диаграми на последователност  
(Sequence diagrams)
  - Диаграми на комуникация  
(Communication diagrams)
-

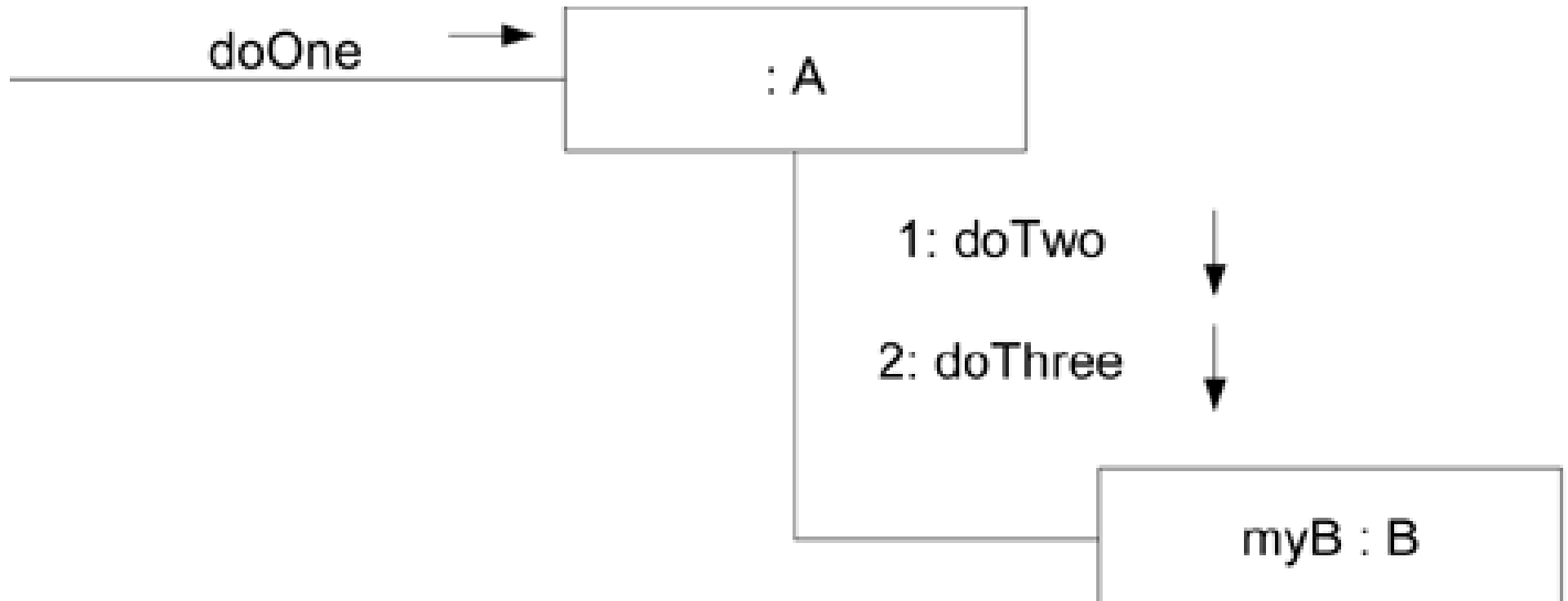
# Sequence diagram

---



# Communication diagram

---



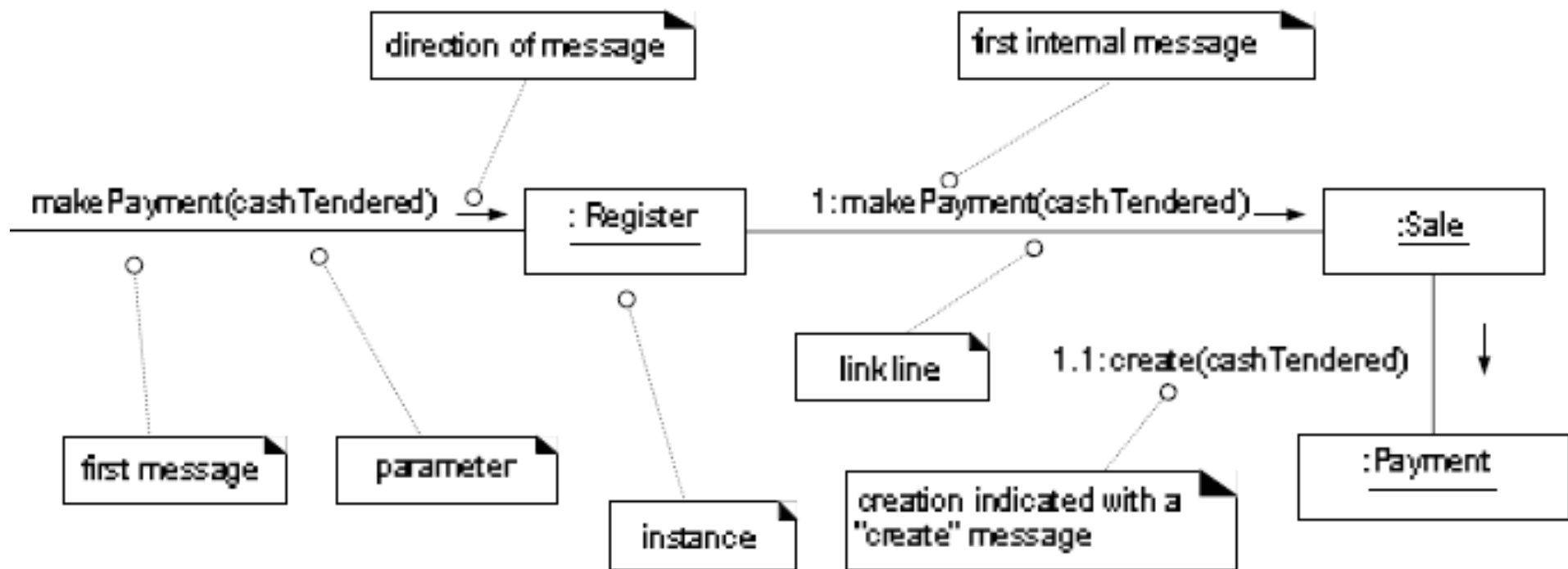
# Диаграми на комуникация или диаграми на последователност

---

Тип	Предимства	Недостатъци
Диаграми на комуникация	Икономични относно място Гъвкави при добавяне на нови обекти По-добри при изобразяването на сложна разклонена структура и конкуретна комуникация	Трудно е да се види последователността на съобщенията. По-сложна нотация.
Диаграми на последователност	Ясно показват последователността на съобщенията Опростена нотация	Нови обекти се добавят винаги отдясно Обемни

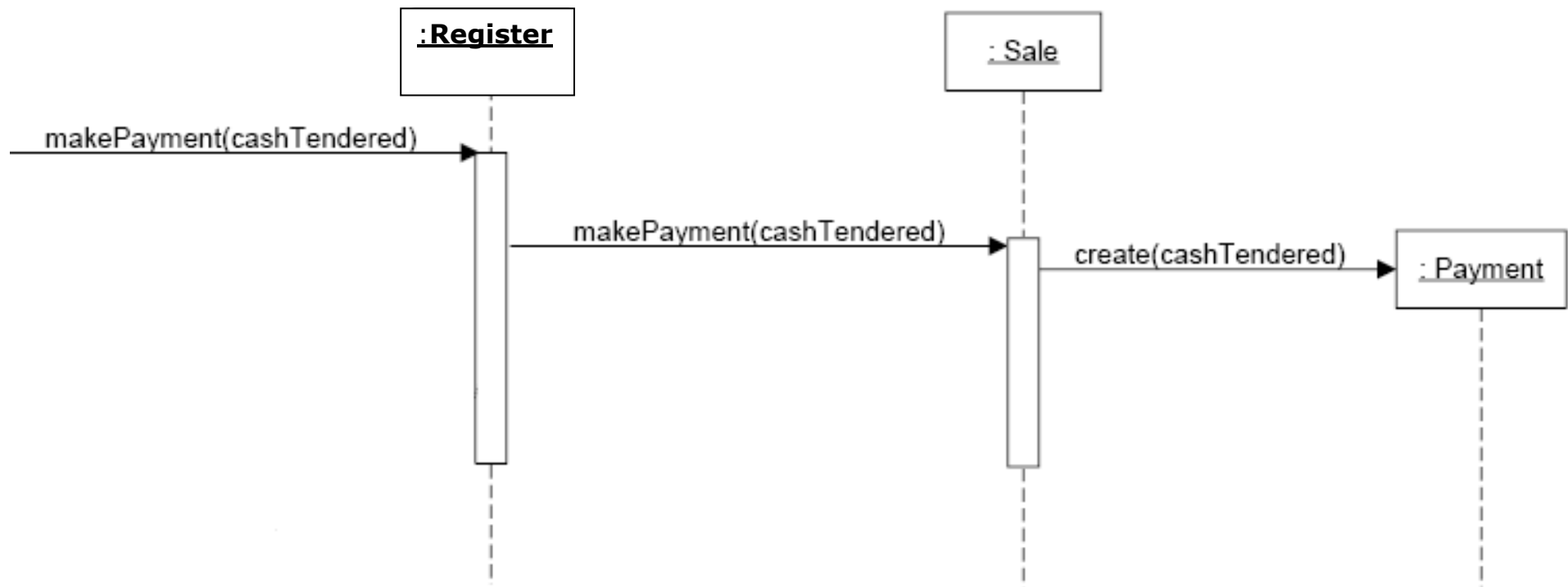
# Диаграма на комуникация - Пример

---



# Диаграма на последователност - Пример

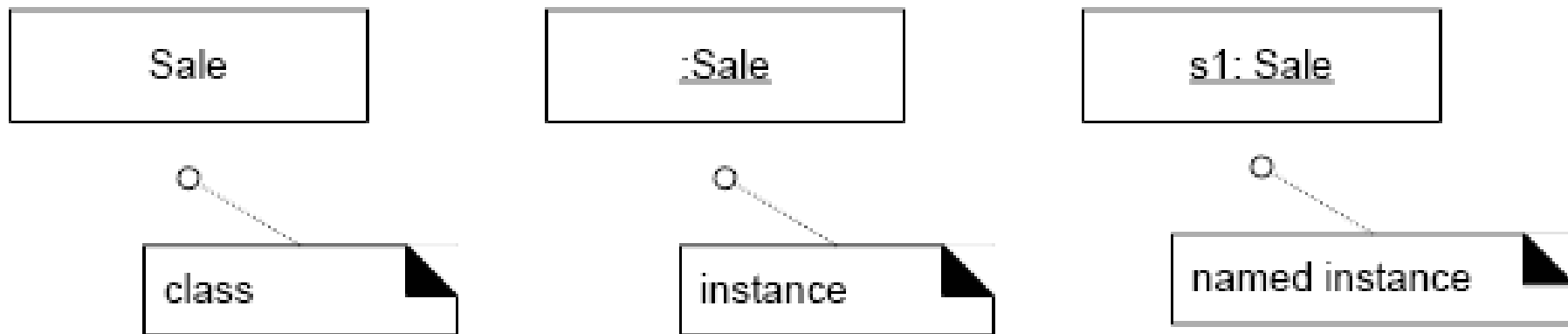
---



# Нотация - Основни символи

---

- Екземплярите използват същия символ като типа, но идентификаторът е подчертан.





# Sequence диаграми

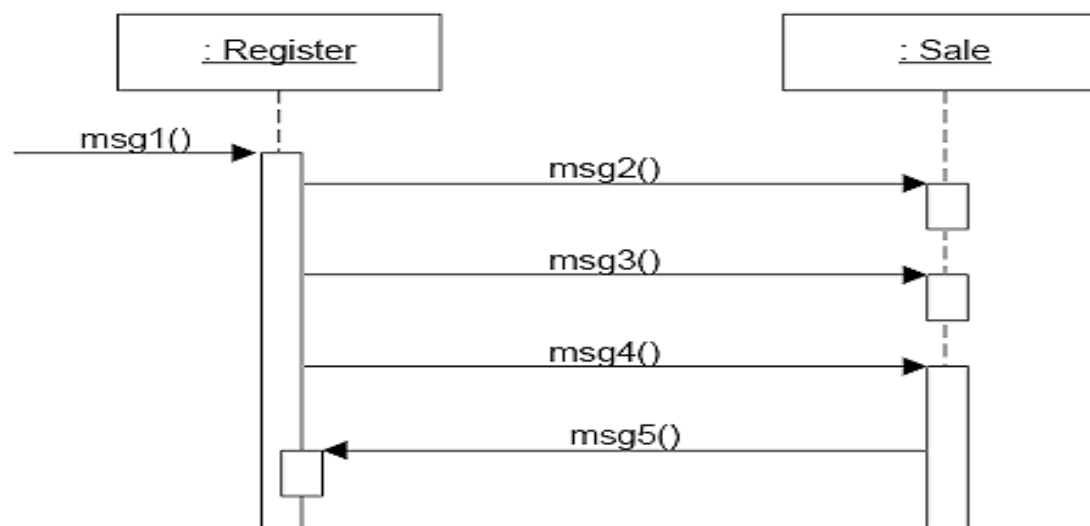
---

- Връзки и съобщения
  - Връщане на резултат
  - "Self" и "this" съобщения
  - Създаване на инстанции
  - Деструкция на инстанции
  - Условни съобщения
  - Итерация
  - Серия от съобщения
  - Multiobject итерации
  - Съобщения към класове
-

# Sequence диаграми

## - Връзки и съобщения

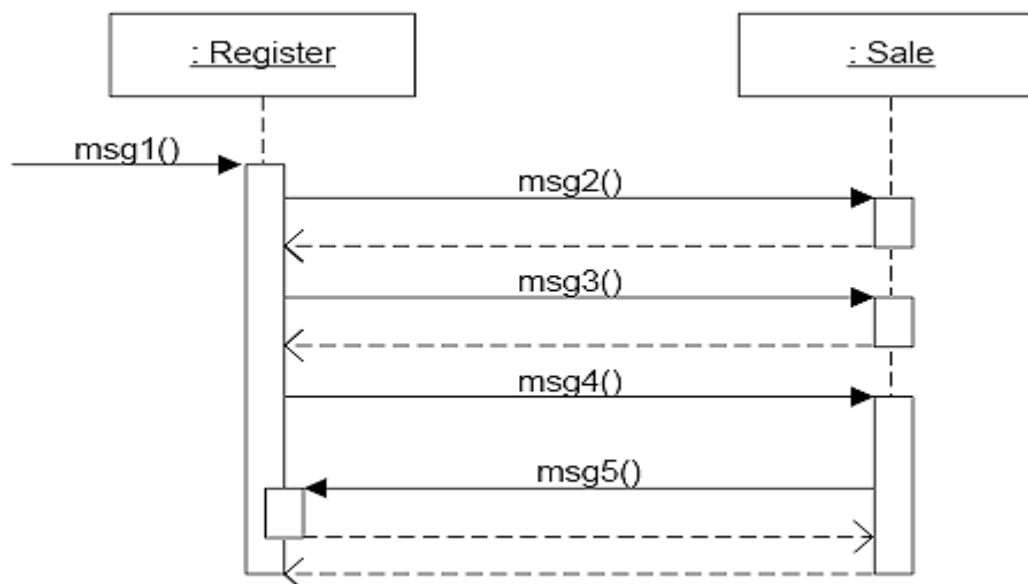
- Sequence диаграмите нямат връзки
- Съобщенията се представят със стрелка за посока и име
- Последователността на съобщенията визуално се представя отгоре надолу
- Диаграмите за последователност могат да показват също и фокуса на управление (не е задължително)



# Sequence диаграми

- Връщане на резултат

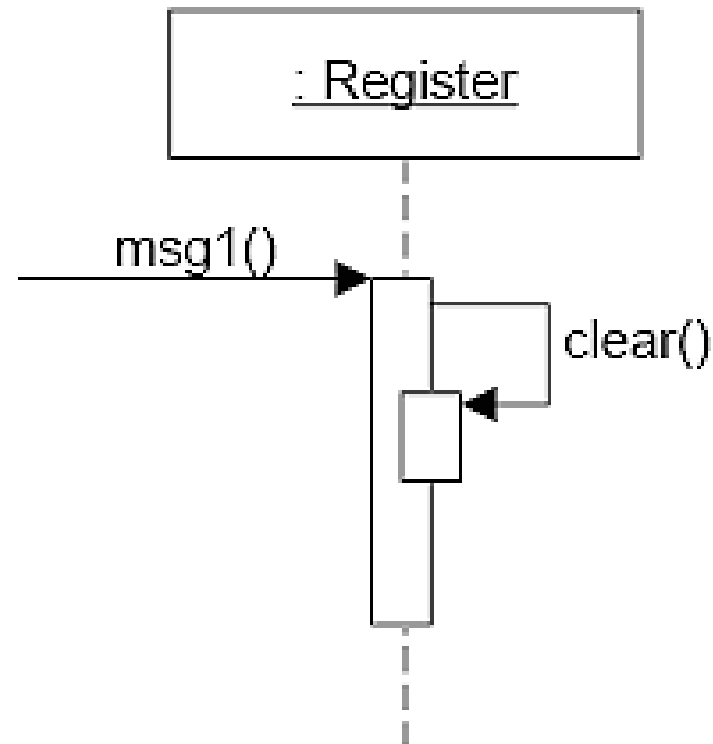
- Връщане на резултат от някое съобщение се представя със стрелка, обратна на стрелката на съобщението
- message syntax ***returnVar = message(parameter)***



# Sequence диаграми

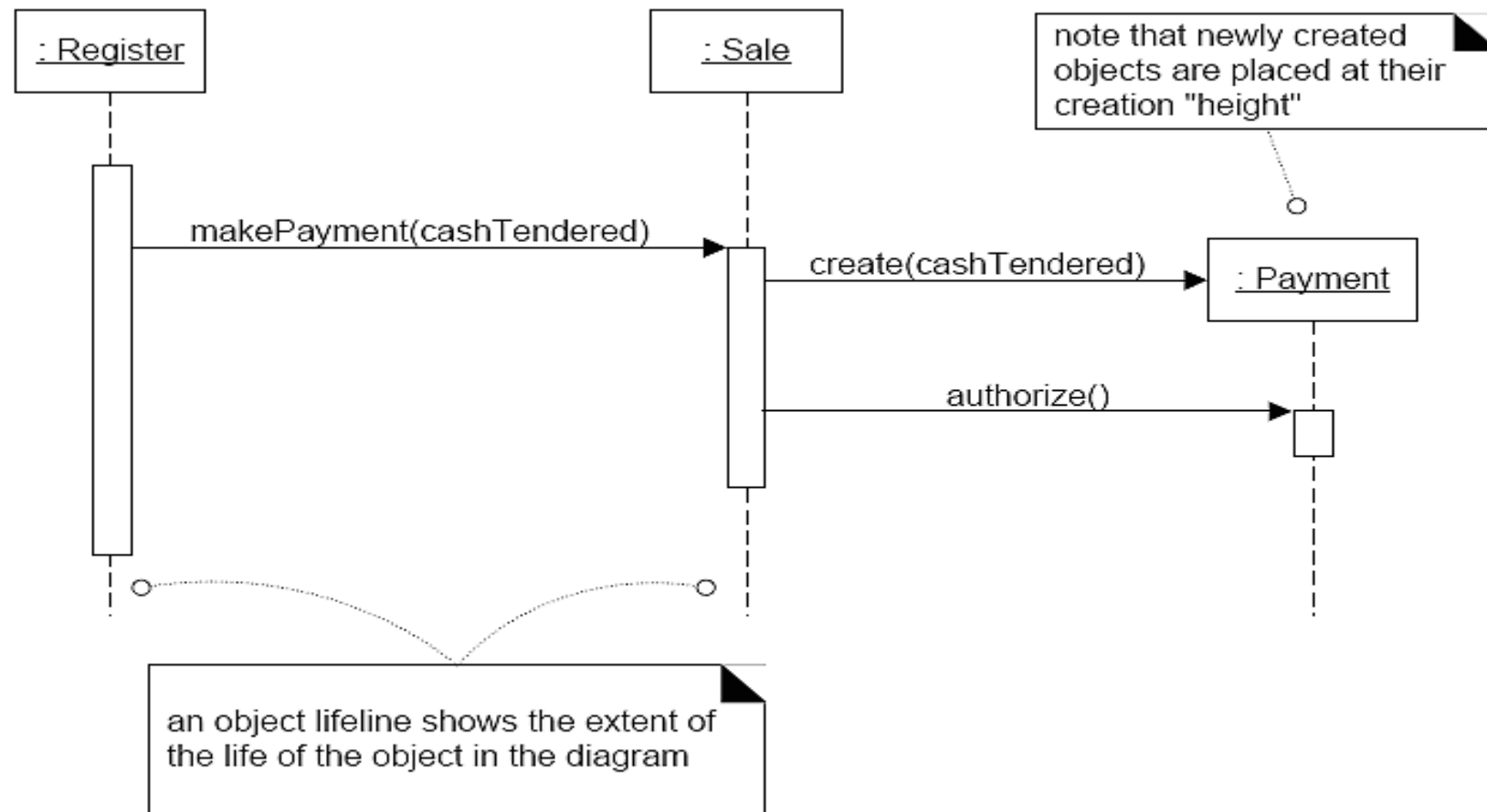
- "Self" и "this"

- Вложени *activation boxes*



# Sequence диаграми

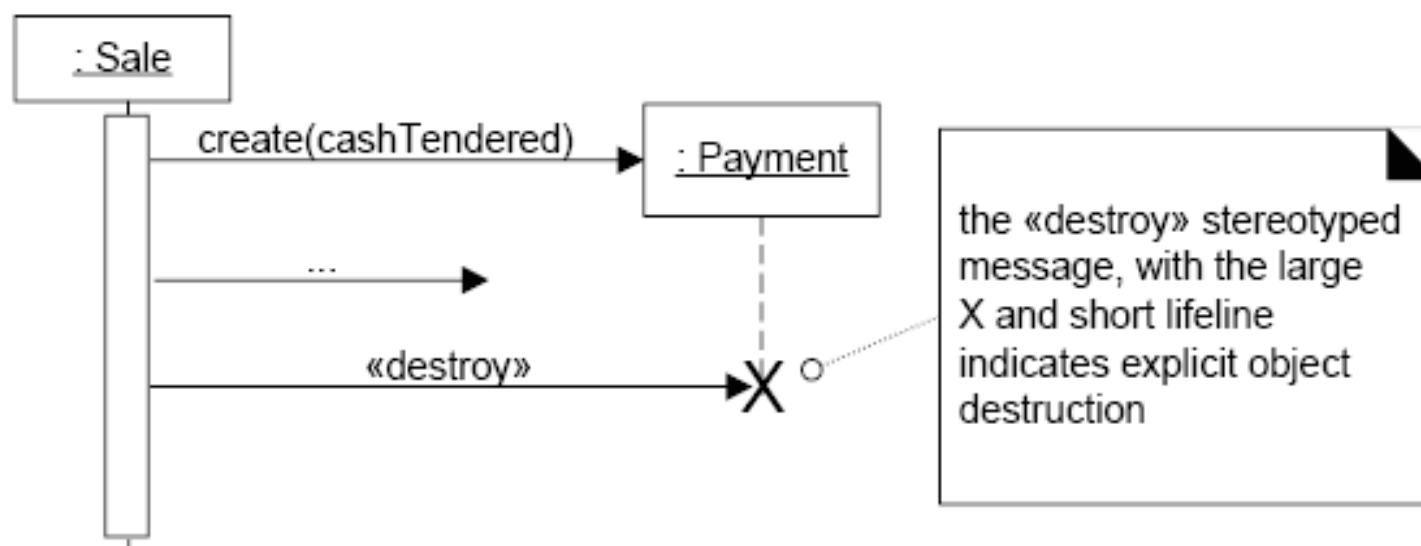
- Създаване на инстанции



# Sequence диаграми

## - Деструкция на инстанции

- Вертикалната линия под инстацията *Payment* означава, че в дадения момент обекта се унищожава
- Това е удобен начин на описание, когато експлицитно трябва да се унищожи някой обект (в случай, че нямаме garbage collector)



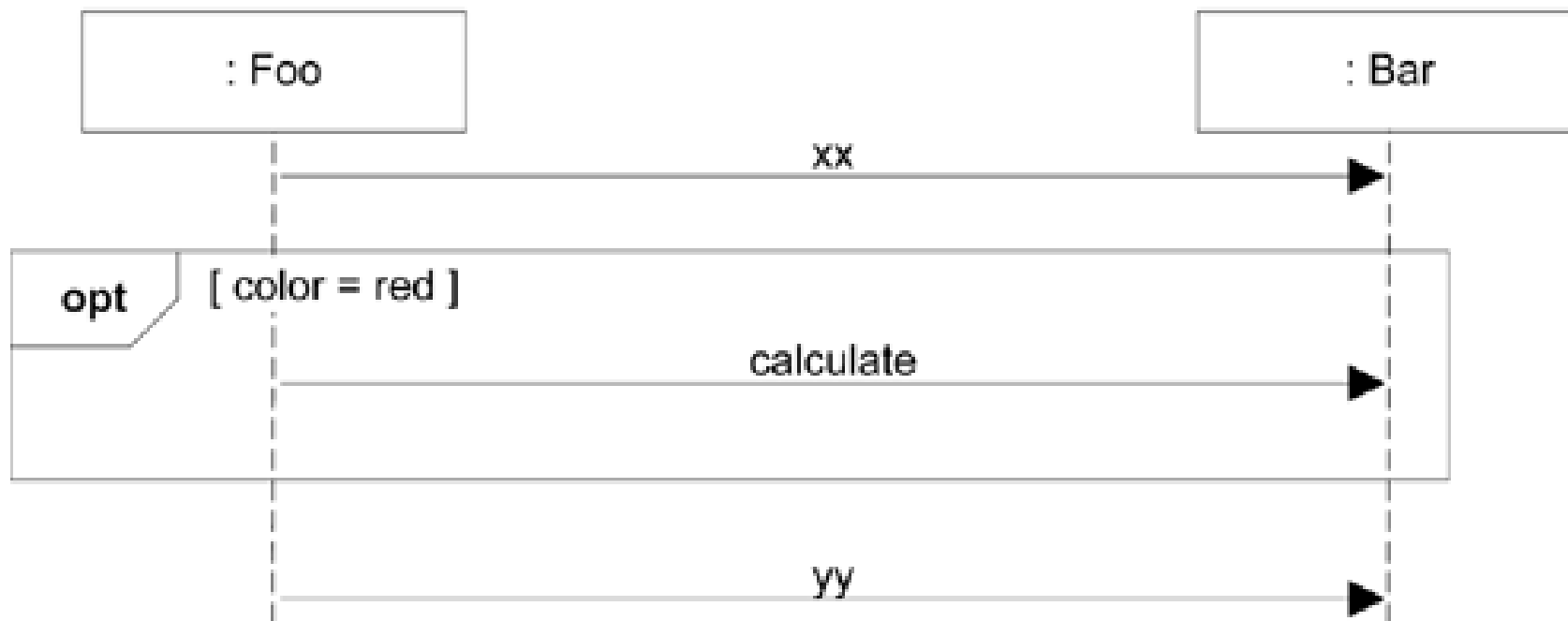
# Diagram Frames in UML Sequence Diagrams

---



# Conditional Messages in UML Sequence Diagrams

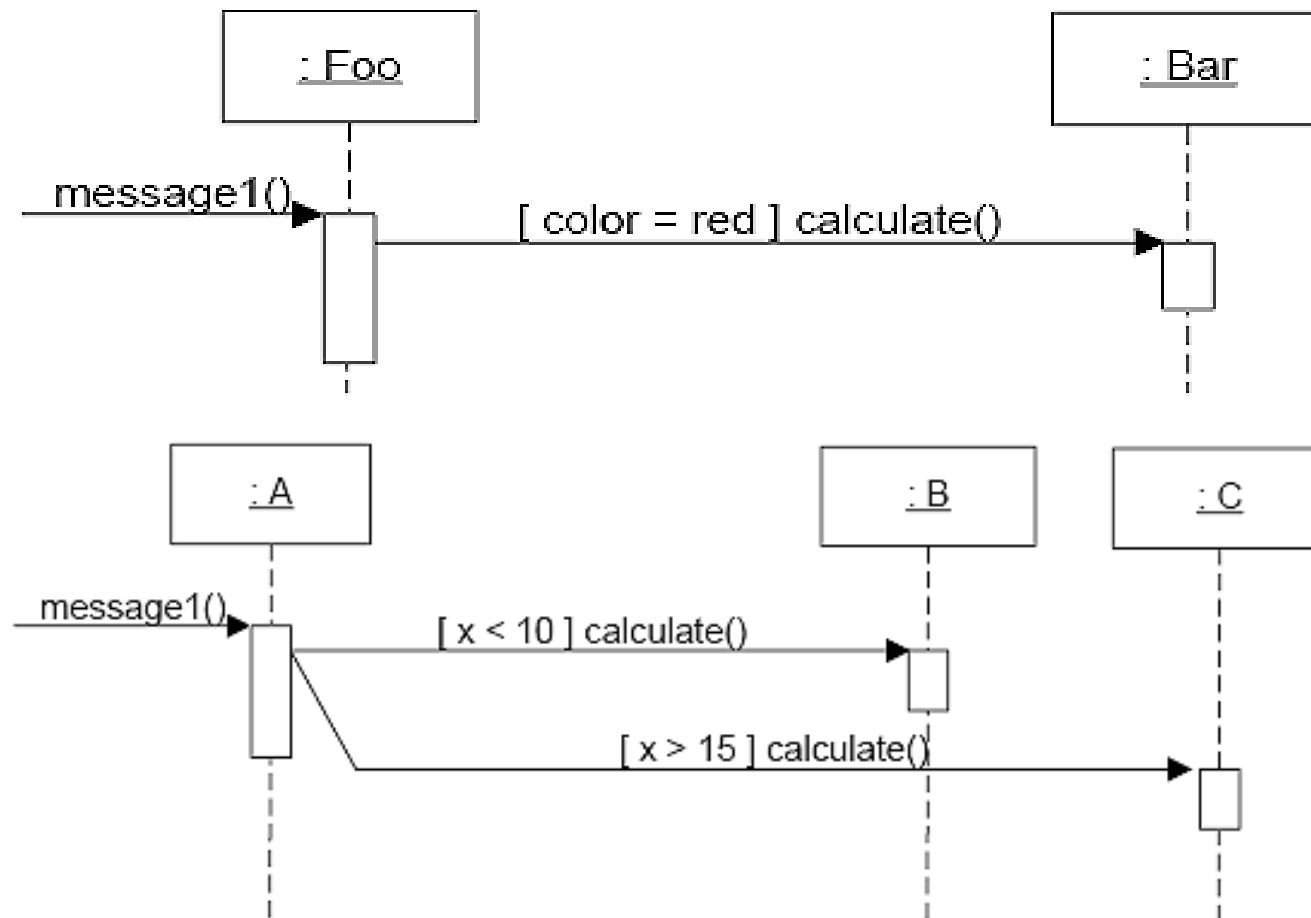
---





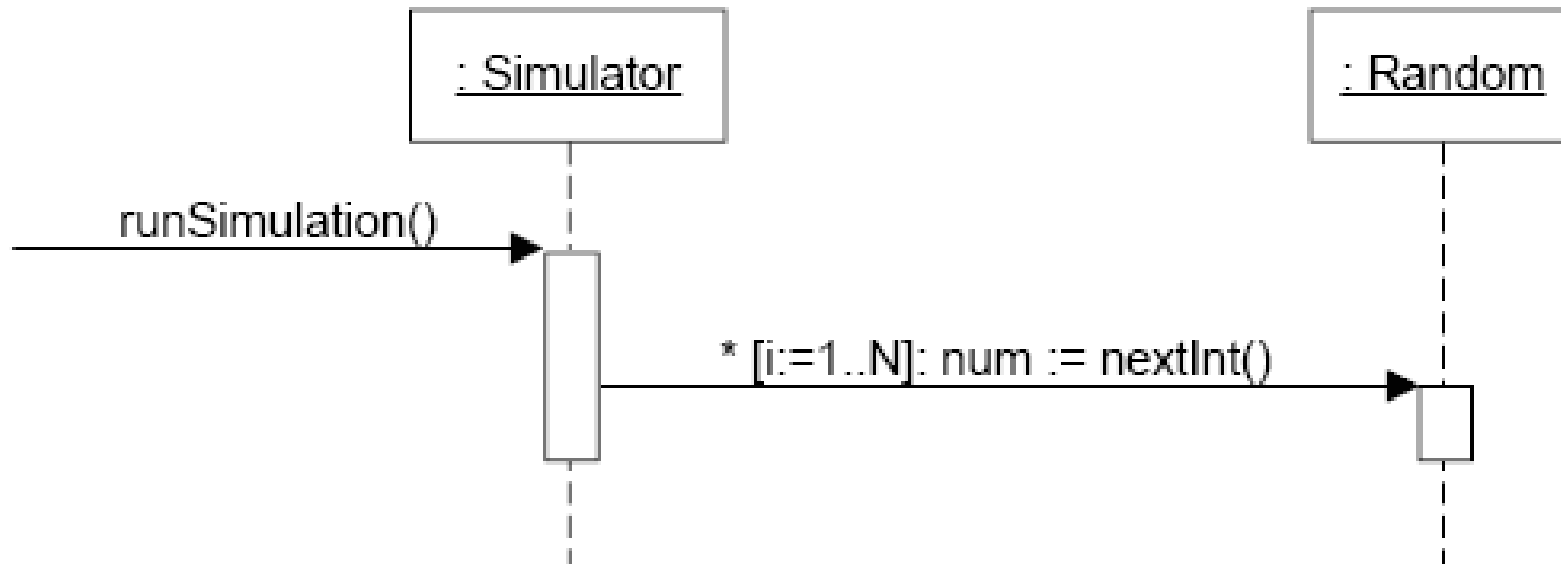
# Sequence диаграми

- Условни съобщения



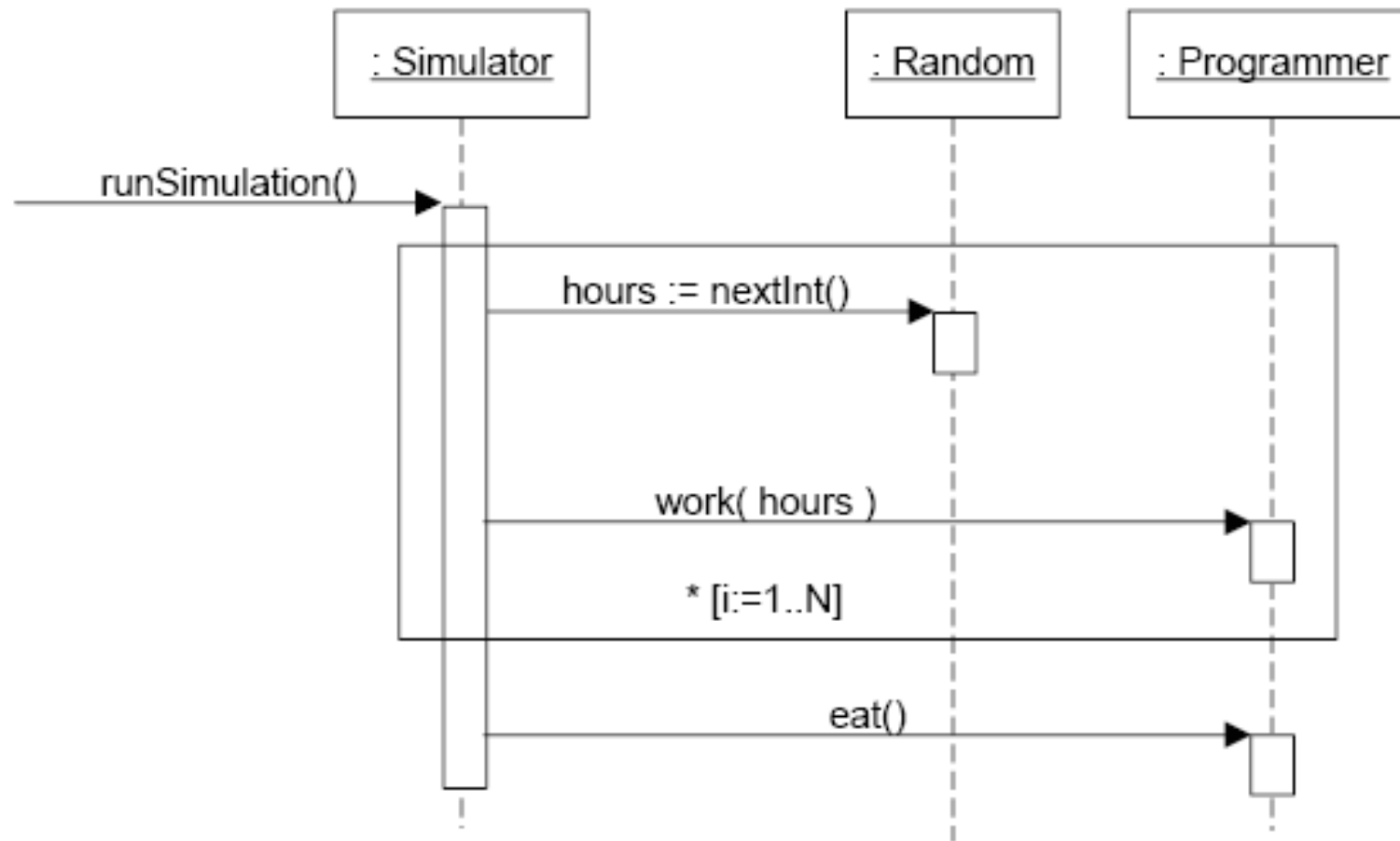
# Sequence диаграмми

- Итерация



# Sequence диаграми

- Серия от съобщения



# Communication diagrams

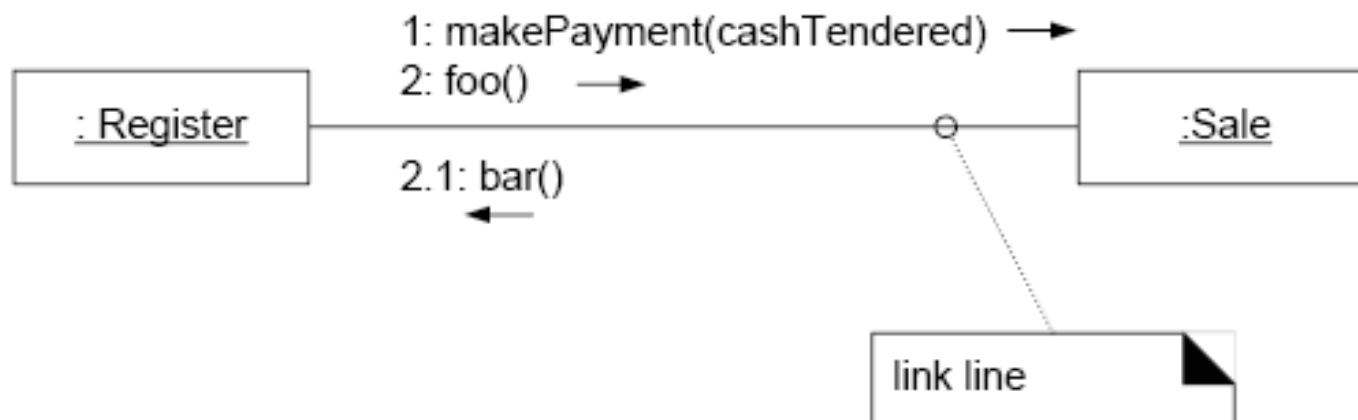
---

- Съобщения
  - "Self" и "this" връзки
  - Създаване на инстанции
  - Номерация
  - Условни съобщения
  - Взаимно изключващи се условия
  - Итерация
  - Съобщения към класове
-

# Communicaton диаграми

## - Връзки(links)

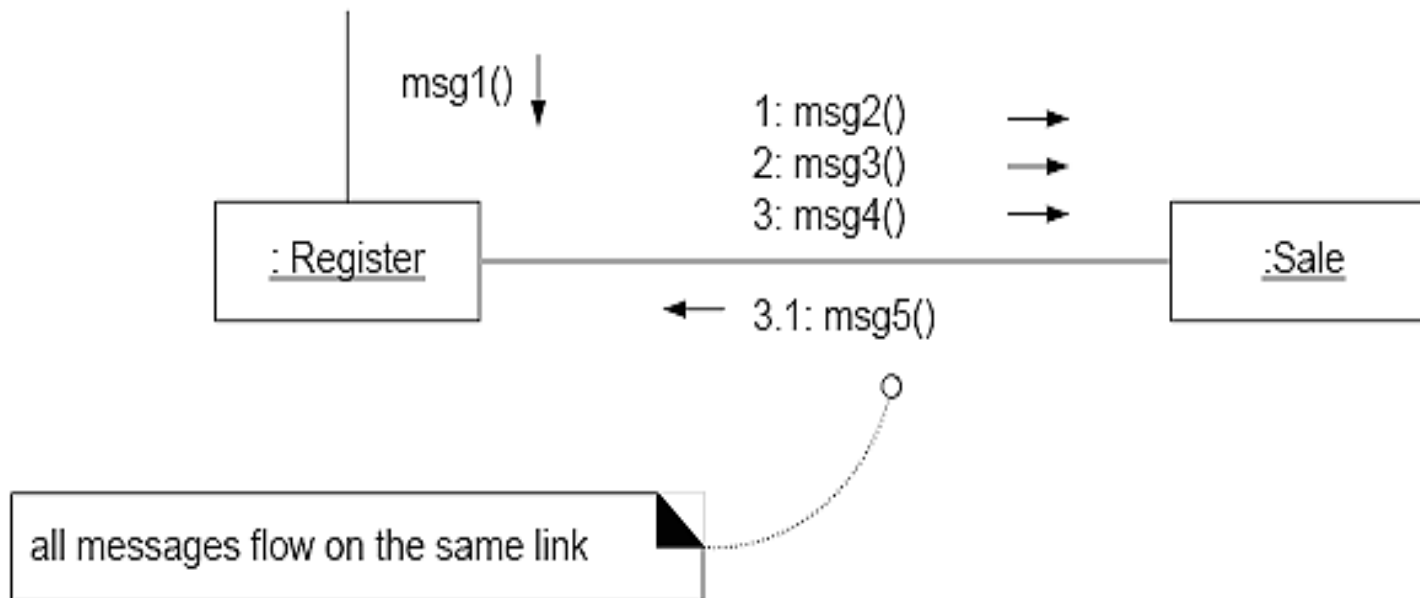
- В примера е показано, че има връзка от Register към Sale, като между тях могат да протичат съобщения
- По една връзка могат да се предават повече от 1 съобщения



# Communication diagrams

## - Съобщения

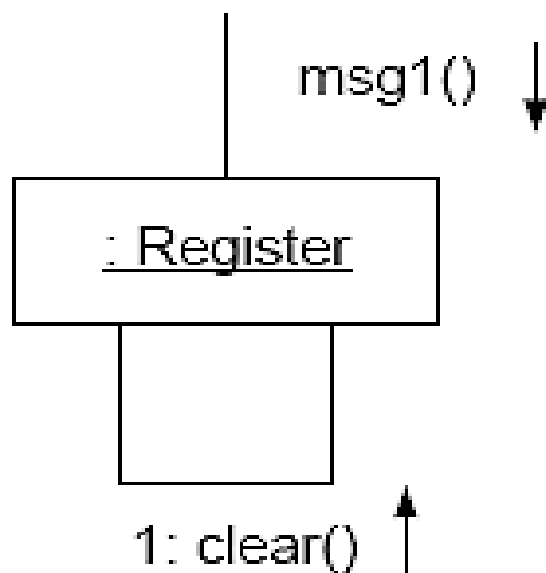
- Подредбата на съобщенията се указва чрез номера
- Стрелката означава посоката на съобщението
- Хубаво е да се измислят подходящи наименования за съобщенията



# Communicaton диаграми

- "Self" и "this"

- Един обект може да има връзка към себе си.
- Съобщенията, които се предават по такива връзки се представят по същия начин като обикновените.



# Communicaton диаграми

## - Създаване на инстанции

---

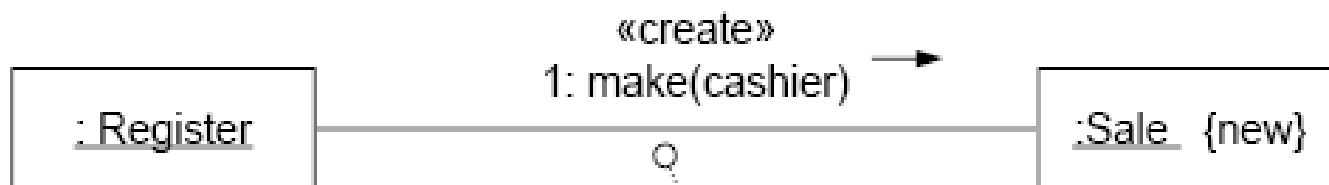
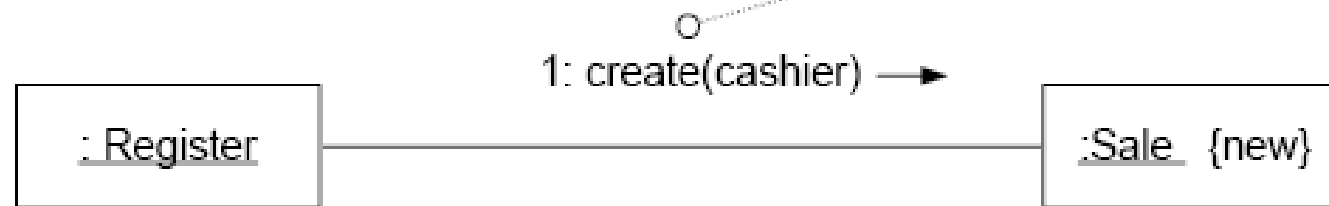
- Обект от даден клас е негова инстанция.
  - UML конвенция: съобщенията които създават инстанции имат име *create*
  - Ако има нужда името на съобщението да е по-специфично, трябва да се добави стереоип  
>> *create* <<
  - Както и при обикновените съобщения е позволено да се добавят параметри.
  - Свойството {new} е незадължително и указва типа на създавания обект.
-



# Communication diagrams

## - Създаване на инстанции

create message, with optional initializing parameters. This will normally be interpreted as a constructor call.

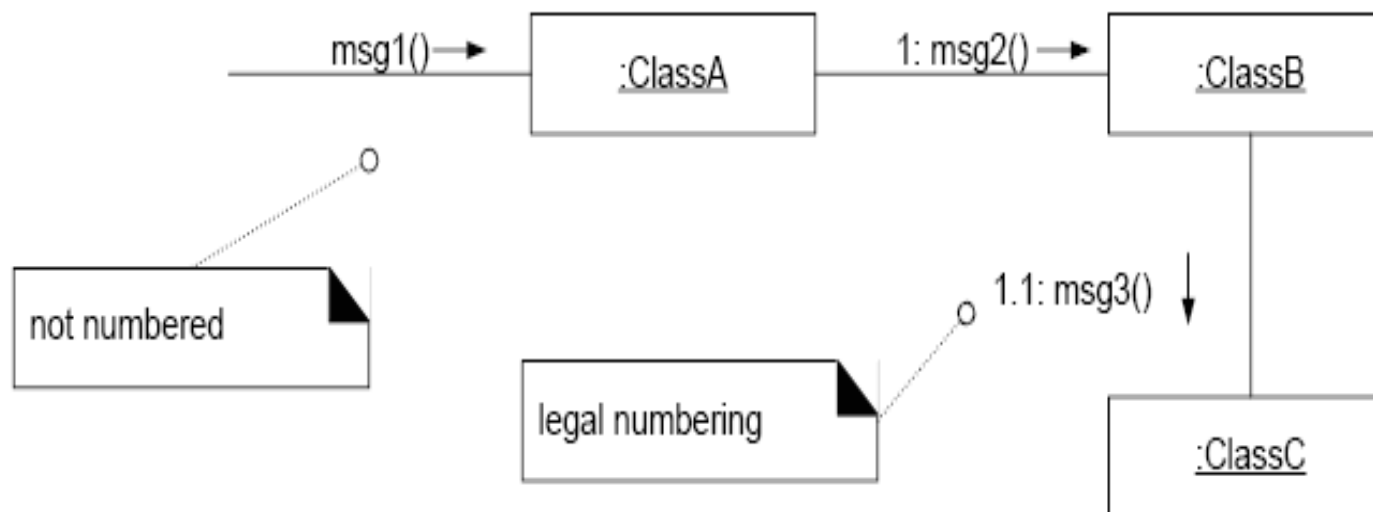


if an unobvious creation message name is used, the message may be stereotyped for clarity

# Communication diagrams

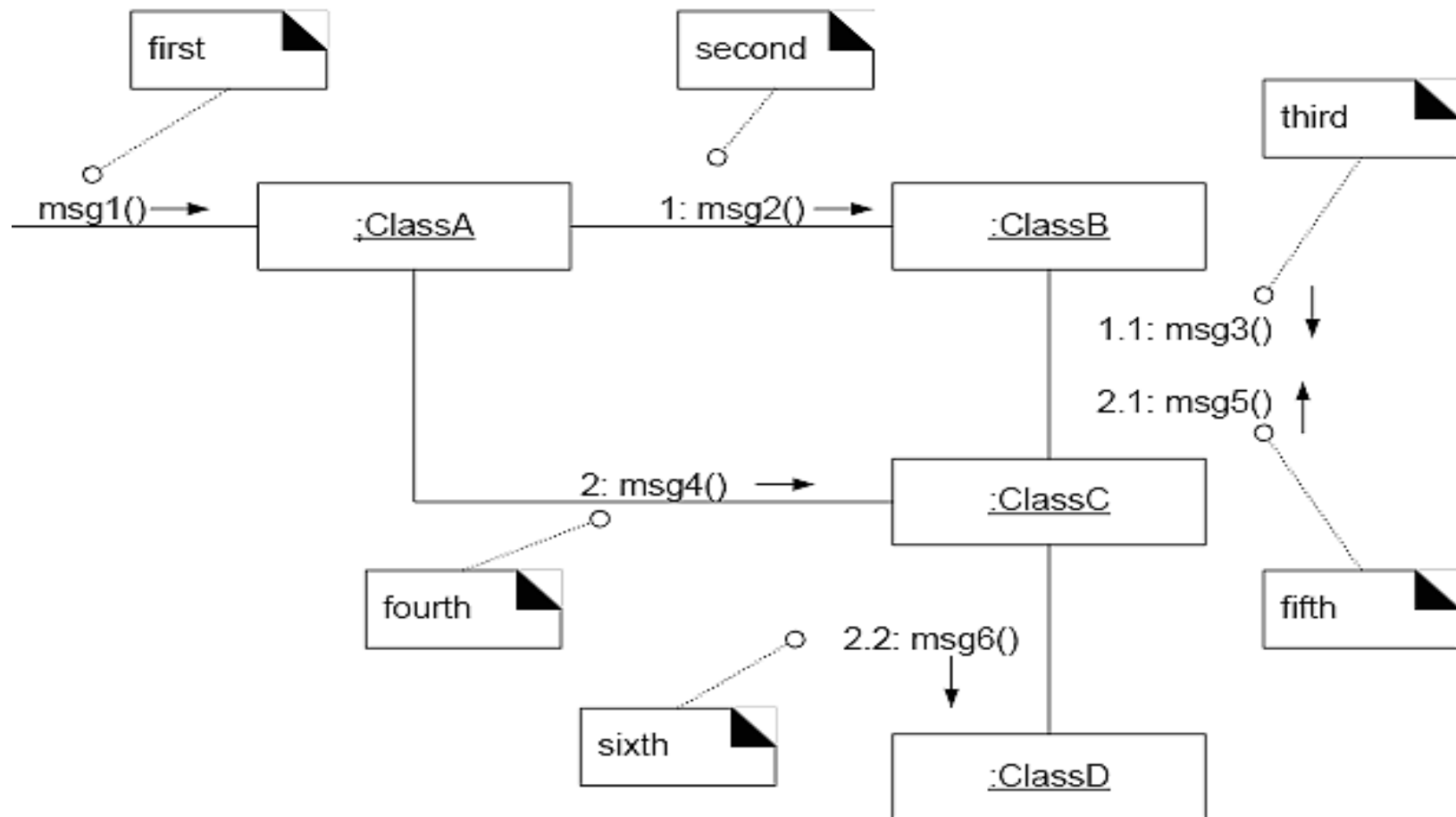
## - Numbering

- Правила при номерация:
  - Първото съобщение не се номерира
  - Основните стъпки се означават с последователни номера
  - Вложените съобщения имат допълнителен номер, който се добавя отпред и е номер на присигащото съобщение.



# Communication diagrams

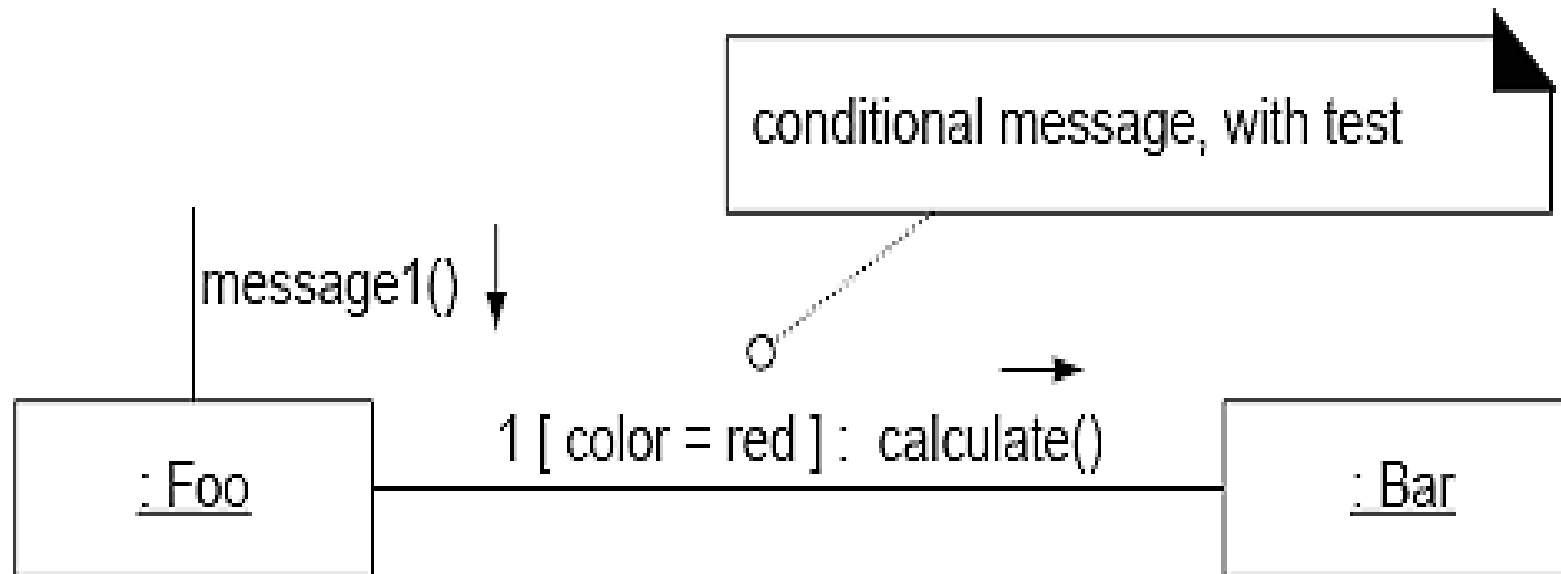
## - Numbering



# Communicaton диаграми

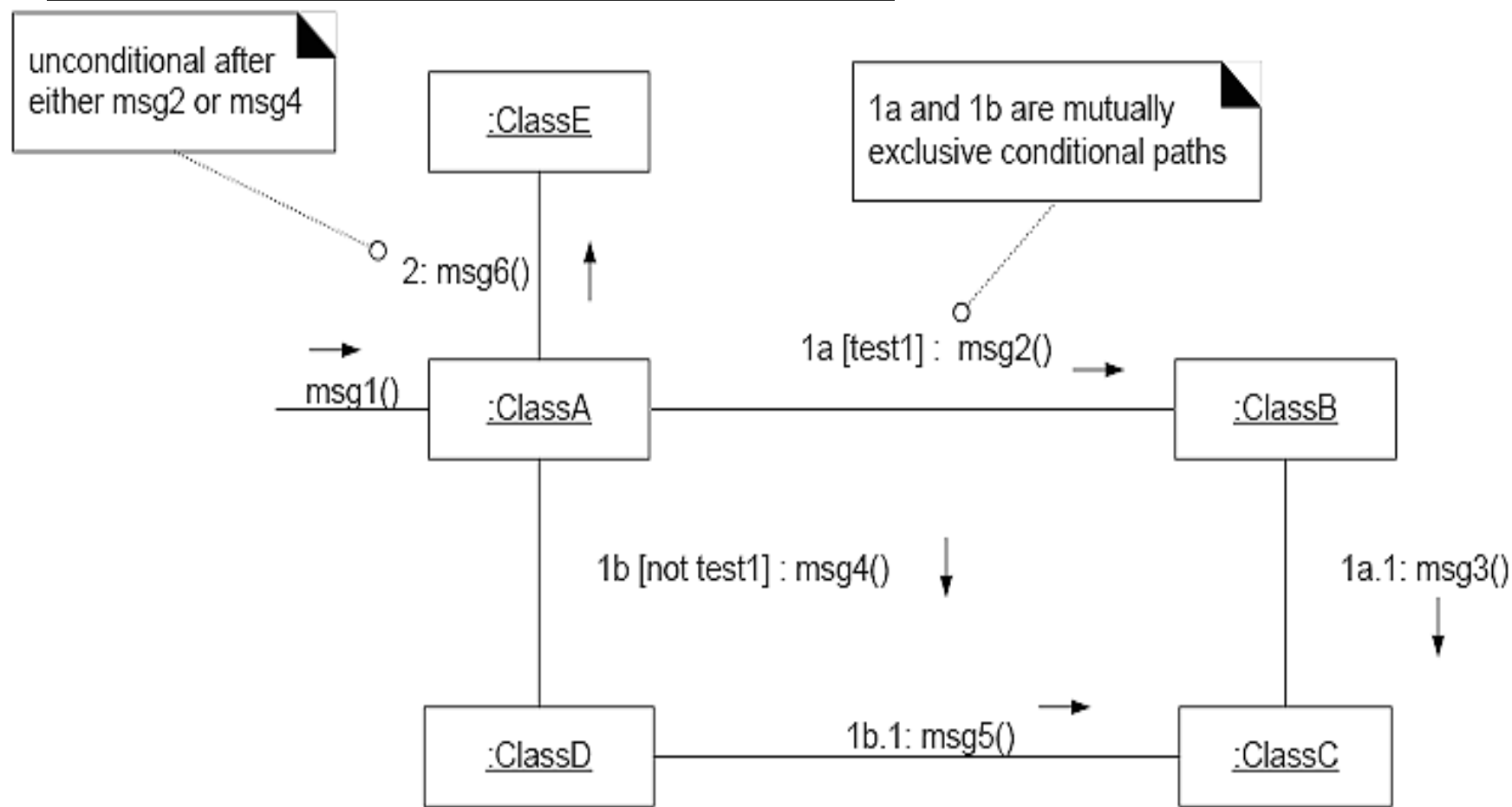
## - Условни съобщения

- Условните съобщения се използват при съобщения, които се изпращат само ако е изпълнено някакво условие.
- Условието се поставя в квадратни скоби след номера на съобщението.



# Communication diagrams

- Взаимно изключващи се условия



# Communicaton диаграми

- Взаимно изключващи се условия

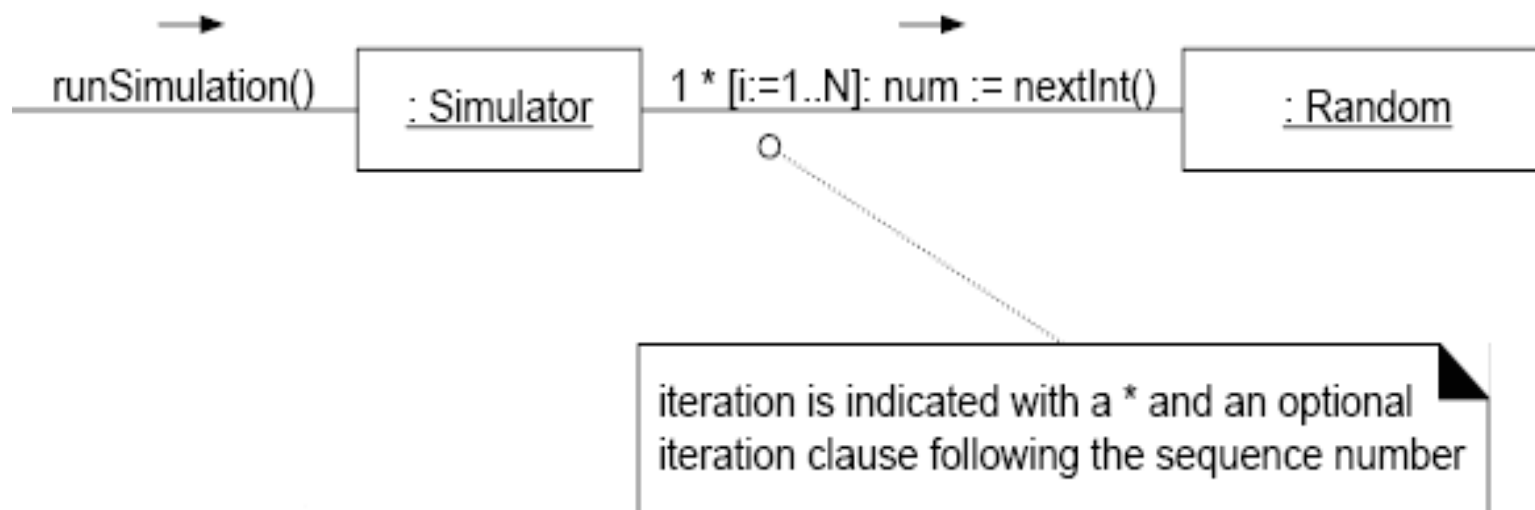
---

- По конвенция първата буква винаги е *a*
  - В примера или се изпълнява *1a* или *1b*
  - Понеже двете съобщения могат да са първи вътрешни съобщения, имат номер 1
-

# Communicaton диаграми

## - Итерация

- Ако условията на итерацията не са важни, може да се използва \*
- Клаузата за итерация е незадължителна
- Детайлите на итерацията се поставят след номера на съобщението



# Communication diagrams

## - Съобщения към класове

- Понякога съобщенията правят обръщение към самия клас, а не към инстанция (например, извикване на статични методи)
- За да се различават класовете от инстанциите, имената на инстанциите се подчитават

