

Об'єктно-орієнтовне моделювання та проектування складних систем

Лекція 5 – Діаграма послідовностей

Ніколаєнко Дмитро Володимирович



Зміст

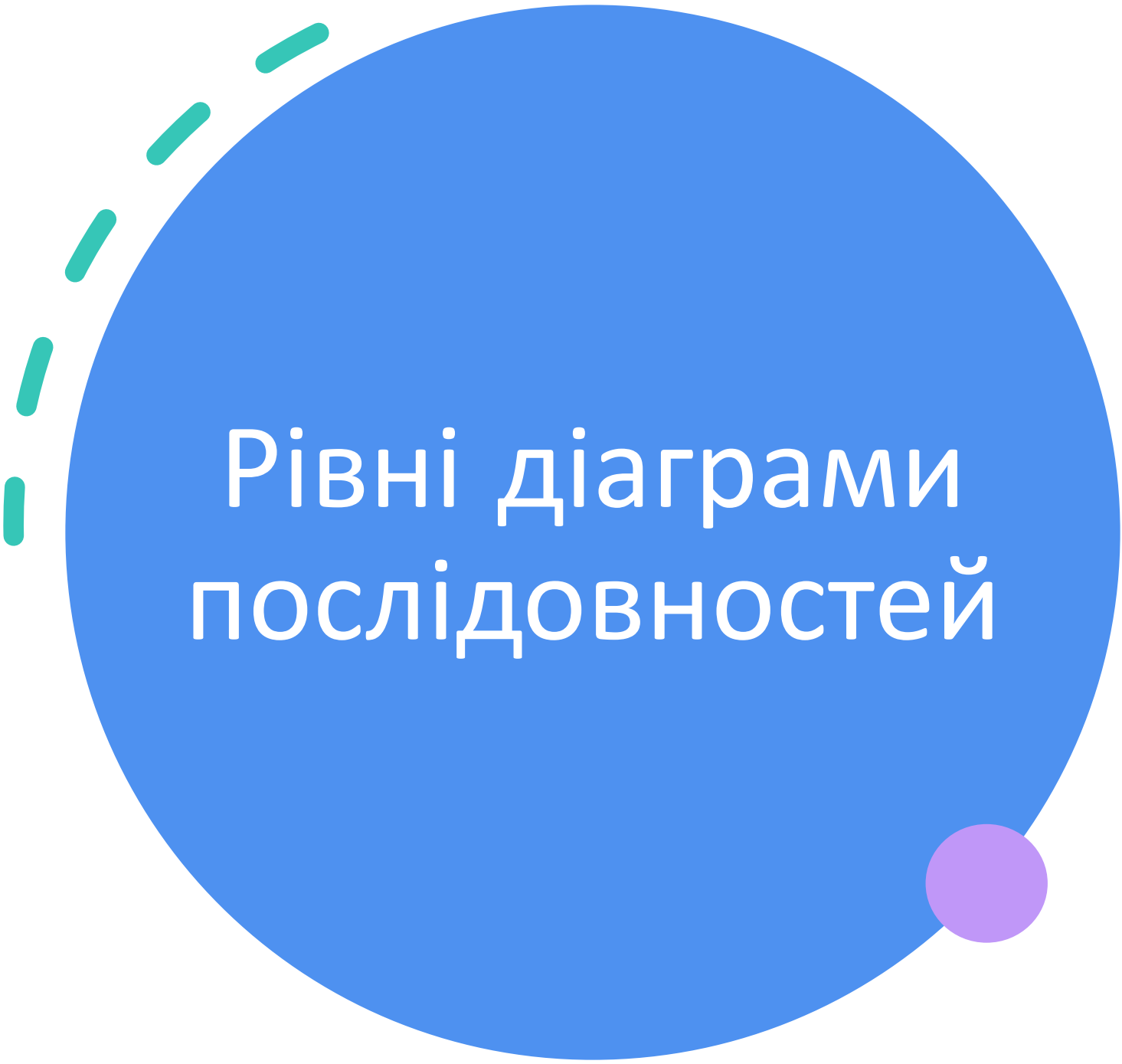
- Що таке послідовності
- Рівні діаграми класів
- Об'єкти (учасники)
- Фокус управління
- Повідомлення
- Створення та видалення об'єктів
- Використання фреймів

Вступ

Послідовності – опис взаємодії груп об'єктів в різних умовах їхньої поведінки.



Діаграма послідовностей



Рівні діаграми послідовностей

Рівні діаграми послідовностей

Концептуальна
(аналітична)
модель

Модель
проектування

Модель
реалізації


Діаграма послідовностей

- На діаграмі послідовності зображуються виключно ті об'єкти, які безпосередньо беруть участь у взаємодії, і не показуються можливі статичні асоціації з іншими об'єктами.
- Для діаграми послідовності ключовим моментом є саме динаміка взаємодії об'єктів у часі.
- Діаграма послідовностей показує саме відносини між об'єктами (учасниками), а не між класами.
- На діаграмі послідовностей не обов'язково представляються лише об'єкти, тому можна називати їх **учасники**

Виміри діаграми послідовностей

Діаграма послідовності має два виміри:

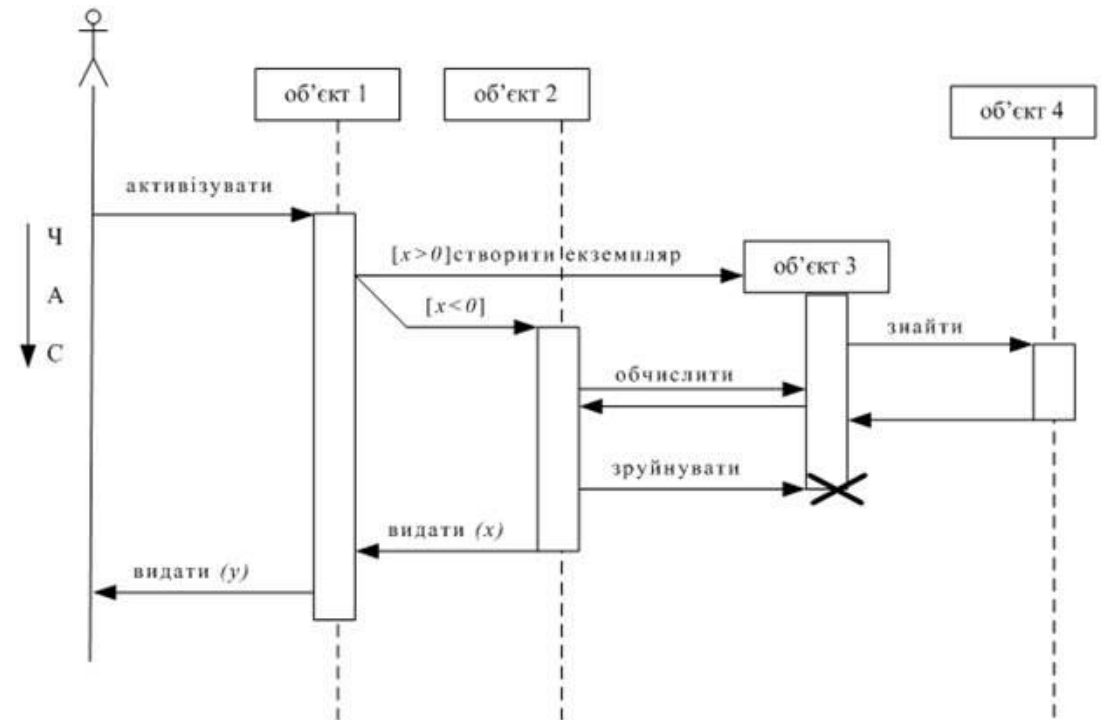
- Горизонтально - зліва направо у вигляді вертикальних ліній, кожна з яких зображує лінію життя окремого об'єкта, що бере участь у взаємодії.
- Вертикально - часова вісь, спрямована зверху вниз. Початковому моменту часу відповідає верхня частина діаграми.



Об'єкТИ
(учасники)

Об'єкт (учасник)

- Об'єкт – це створений екземпляр класу.
- Для заощадження ресурсів системи зовсім не обов'язково створювати всі об'єкти в початковий момент часу.
- У цьому разі прямокутник такого об'єкта зображується не у верхній частині діаграми послідовності, а в тій її частині, яка відповідає моменту створення об'єкта



Лінія життя об'єкта

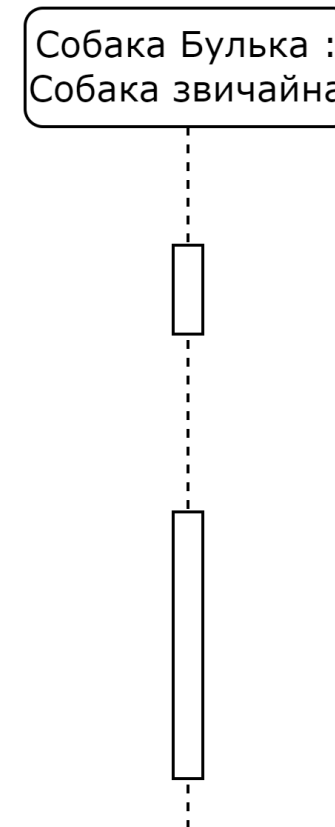
- Лінія життя об'єкта (object lifeline) зображується пунктирною вертикальною лінією, асоційованою з єдиним об'єктом на діаграмі послідовності.
- Лінія життя слугує для позначення періоду часу, протягом якого об'єкт існує в системі і, отже, може потенційно брати участь у всіх її взаємодіях.
- Якщо об'єкт існує в системі постійно, то і його лінія життя має продовжуватися по всій площині діаграми.
- Окремі об'єкти, виконавши свою роль у системі, можуть бути знищені (зруйновані), щоб звільнити ресурси, які вони займають.
- Для позначення моменту знищення об'єкта в мові UML використовується спеціальний символ у формі латинської літери "X"



Фокус
управління

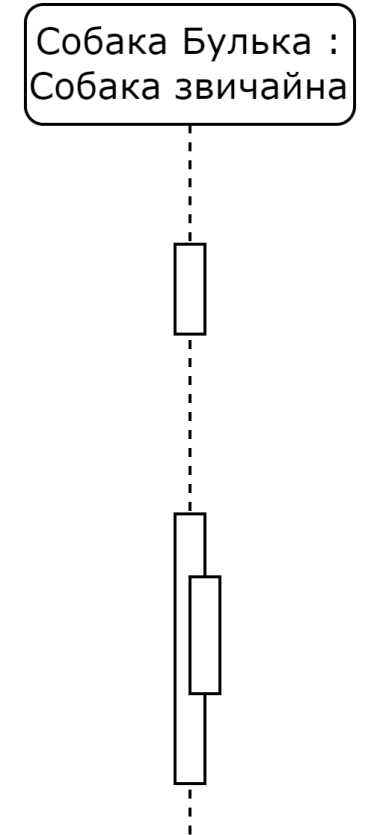
Фокус управління

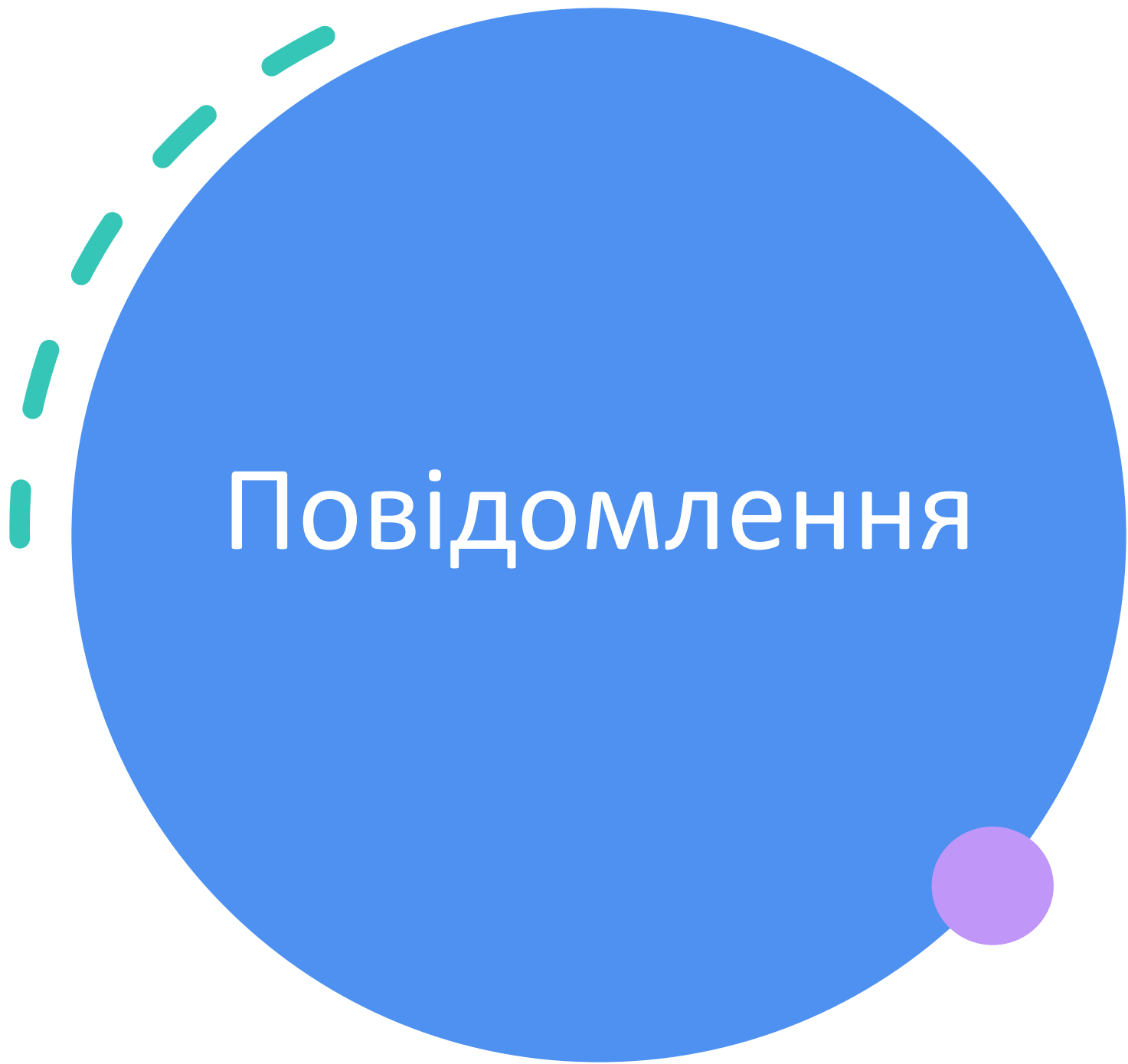
- Об'єкти можуть перебувати в активному стані, безпосередньо виконуючи певні дії або в стані пасивного очікування повідомлень від інших об'єктів.
- Щоб явно виокремити подібну активність об'єктів, у мові UML застосовують спеціальне поняття, що отримало назву фокусу керування (focus of control).
- Фокус управління зображується у формі витягнутого вузького прямокутника, верхня сторона якого позначає початок отримання фокусу управління об'єкта (початок активності), а її нижня сторона - закінчення фокусу управління (закінчення активності).
- Періоди активності об'єкта можуть чергуватися з періодами його пасивності або очікування. У цьому разі в такого об'єкта є кілька фокусів управління



Фокус управління – рекурсія

- Іноді деякий об'єкт може ініціювати рекурсивну взаємодію із самим собою.
- Наявність у багатьох мовах програмування спеціальних засобів побудови рекурсивних процедур вимагає візуалізації відповідних понять у формі графічних примітивів.
- На діаграмі послідовності рекурсія позначається невеликим прямокутником, приєднаним до правого боку фокуса управління того об'єкта, для якого зображується ця рекурсивна взаємодія.
- На цьому прикладі собака булька бігає за своїм хвостом.

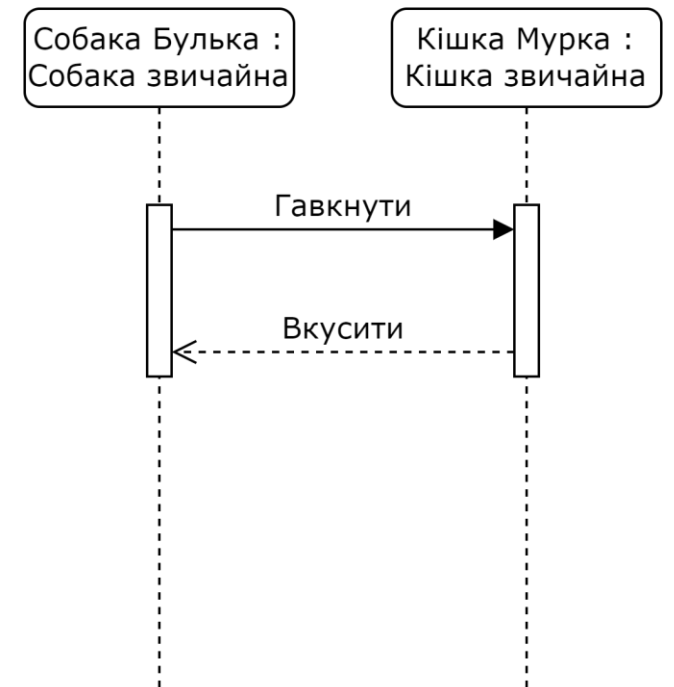




Повідомлення

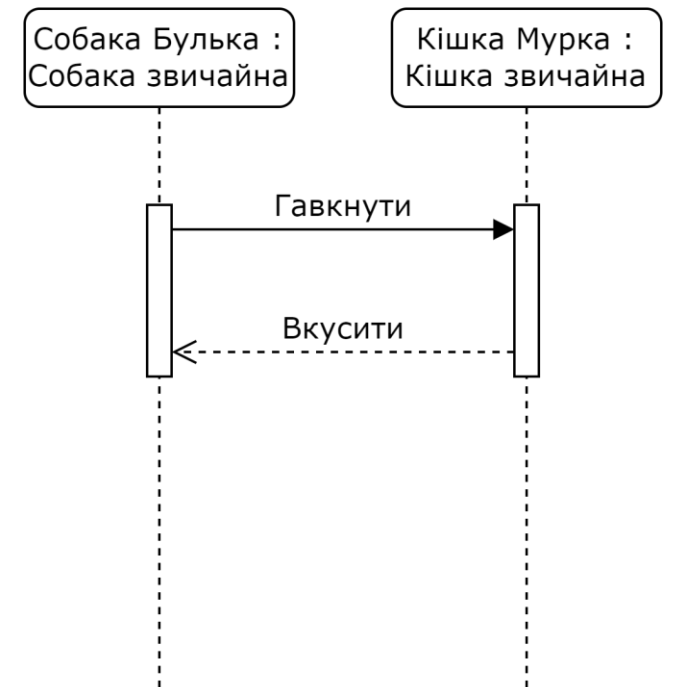
Повідомлення

- Повідомлення (message) являє собою закінчений фрагмент інформації, який надсилається одним об'єктом іншому.
- Повідомлення не тільки передають деяку інформацію, а й вимагають або припускають від об'єкта, що приймає, виконання очікуваних дій.
- Повідомлення можуть ініціювати виконання операцій об'єктом відповідного класу, а параметри цих операцій передаються разом із повідомленням.
- На діаграмі послідовності всі повідомлення впорядковані за часом свого виникнення.



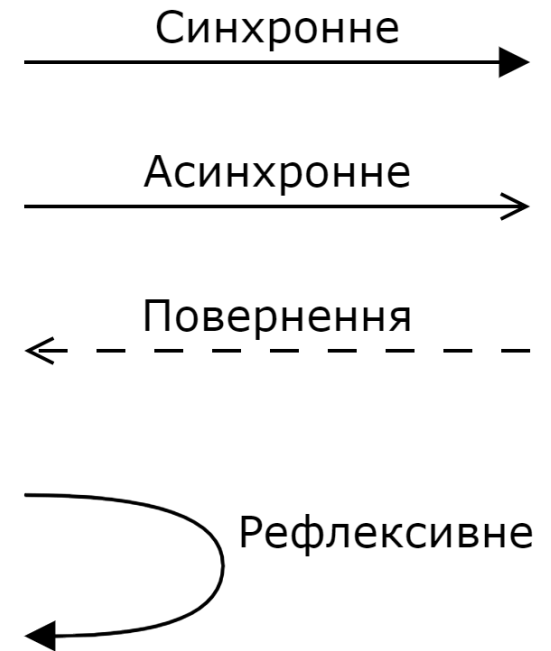
Повідомлення

- Зазвичай над повідомленнями вказується яке саме повідомлення відправляється
 - getPrice
 - getQuantity(quantity: number)
 - calculateBasePrice
- UML не має засобів відображення типу даних тому використовується параметр в повідомленні

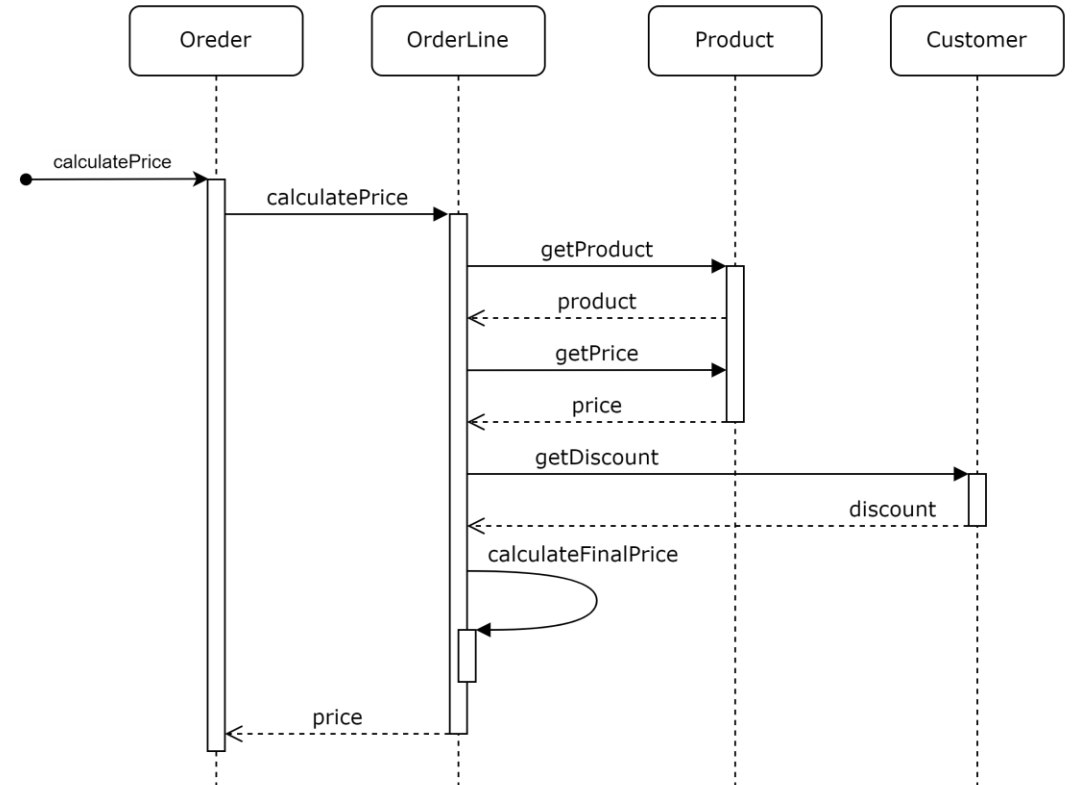
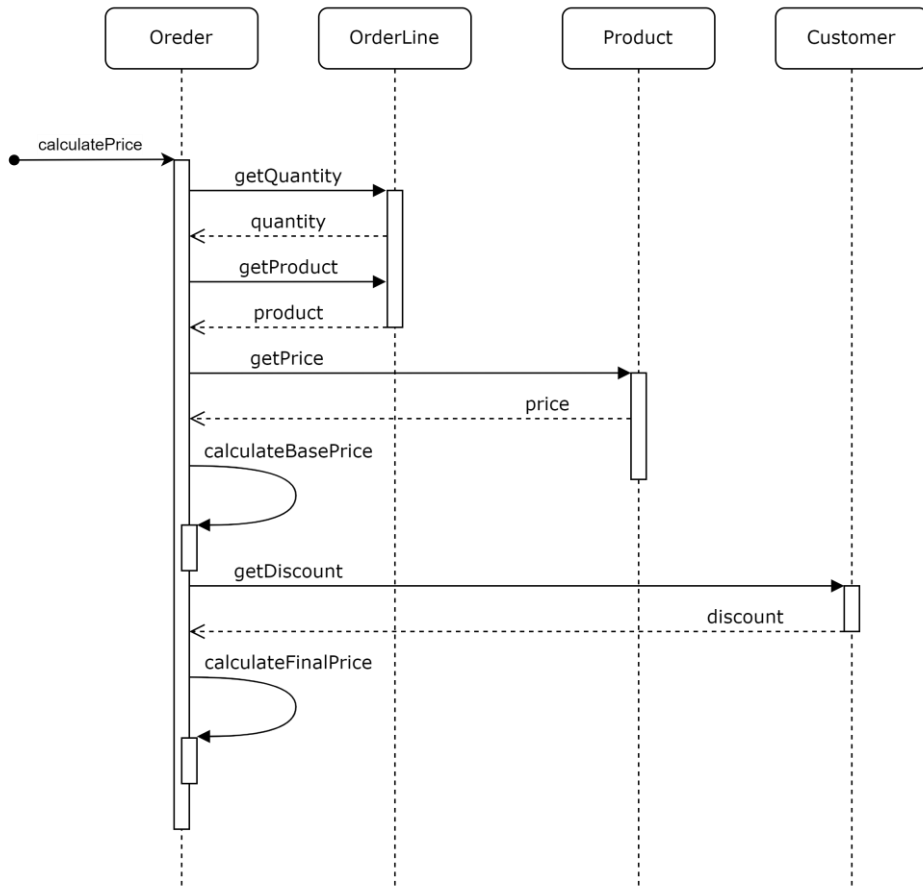



Типи повідомлень

- **Синхронне** повідомлення є найпоширенішим і використовується для виклику процедур, виконання операцій або позначення окремих вкладених потоків управління.
- **Асинхронне** повідомлення викликає дію іншого об'єкта, проте поточний об'єкт може продовжувати роботу та не чекати на відповідь
- Повернення результатів роботи процедури
- Рефлексивне - повідомлення самому собі.



Діаграма послідовностей

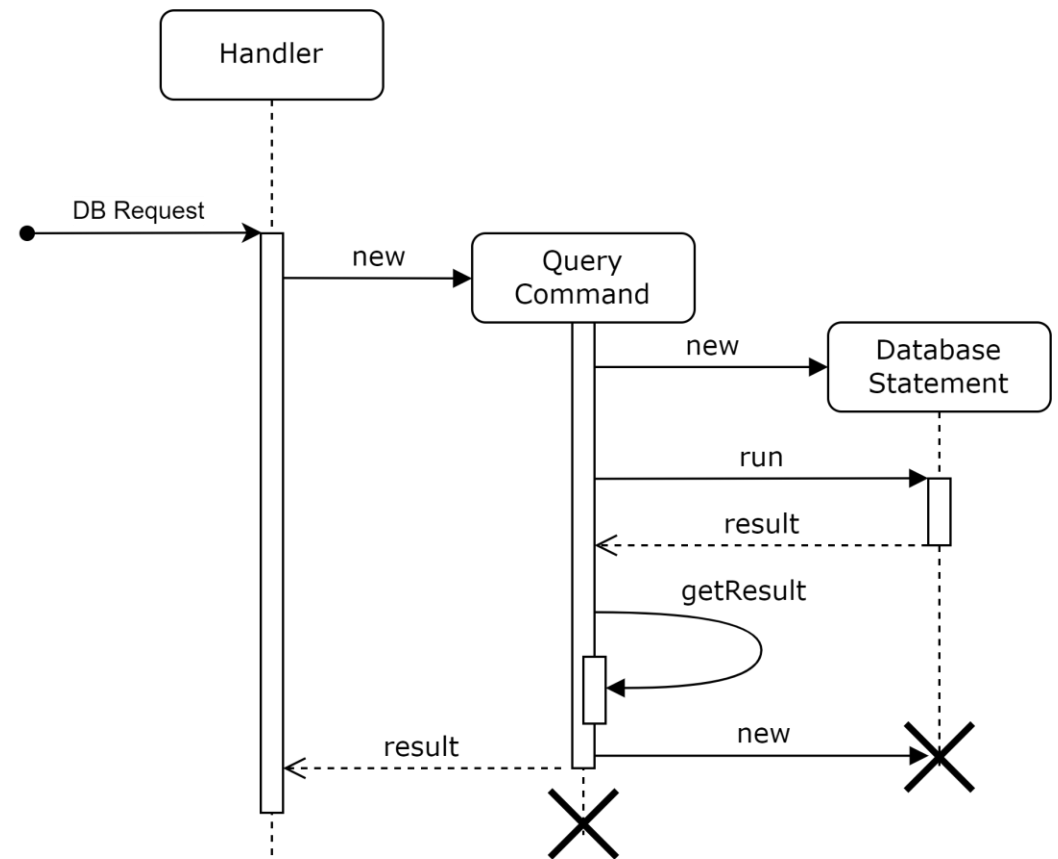




Створення та
видалення
об'єктів
(учасників)

Створення та видалення учасників

- Створення показується подією "new"
- Видалення або самовидалення показується символом X
- Видалення об'єктів є важливим з точки зору оптимізації додатку

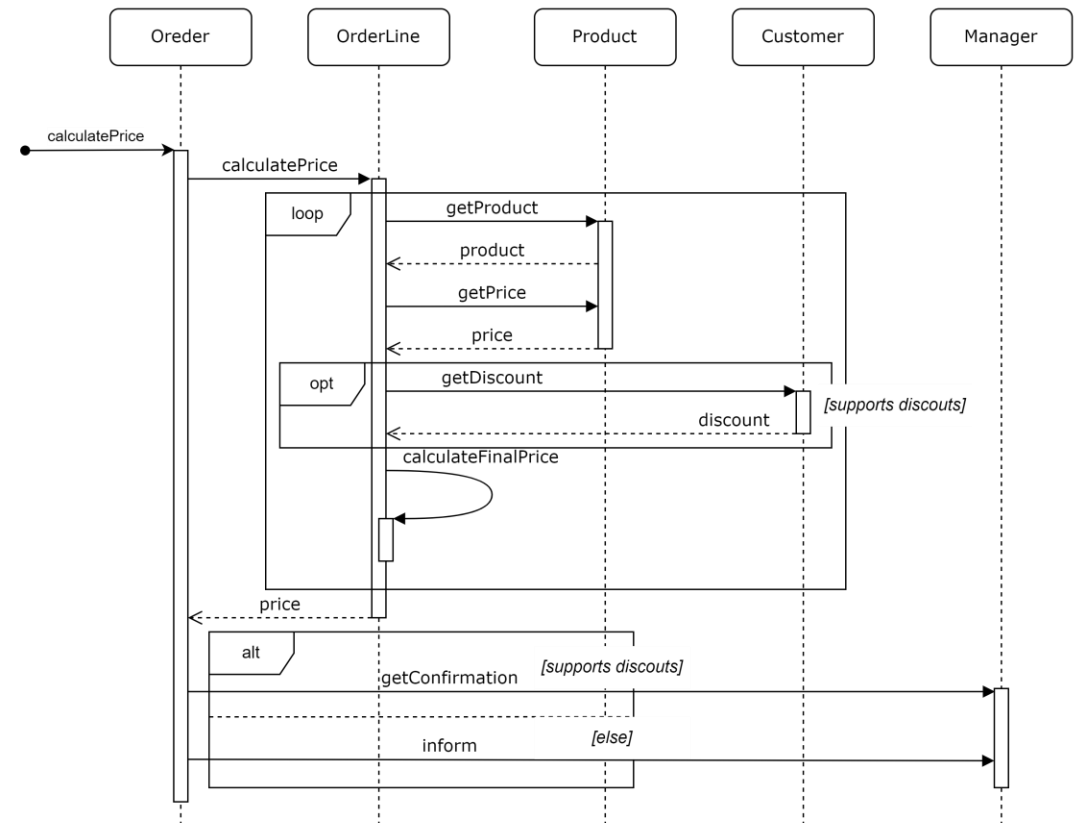




Фрейми

Фрейми на діаграмі послідовностей

- loop – цикл
- alt – альтернативні фрагменти
- alt – необов'язковий фрагмент
- par – паралельний фрагмент
- region – критична область, може мати тільки один потік
- neg – невірна взаємодія
- ref – посилання на взаємодію (інша діаграма)
- sd – інша діаграма послідовностей



Заключення

- Діаграма послідовностей використовується коли необхідно показати взаємодію декількох об'єктів в межах одного прецеденту
- Якщо потрібна поведінка одного об'єкта в межах декількох прецедентів – використовується діаграма станів
- Якщо потрібна поведінка декількох об'єктів в межах декількох прецедентів – використовується діаграма діяльності



Дякую

Ніколаєнко Дмитро
Володимирович