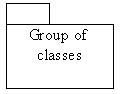
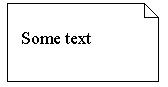
1. **Що таке Уніфікована мова моделювання (UML)?**

Автори і розробники UML представляють її як мову для візуалізації, визначення, проектування та документування програмних систем, бізнес-систем та інших систем, в першу чергу пов'язаних з програмним забезпеченням.

1. **Де і як використовують пакети та анотації?**

Єдиним представником сутності, що групує, є пакет (package). Пакет - це механізм загального призначення для організації елементів у вигляді єдиної групи. Структурні, поведінкові і навіть інші пакети, можуть бути розміщені всередині пакету.  
  


Анотації - це сутності, які в UML вживають для представлення пояснень і коментарів. Єдиним типом анотаційної сутності є примітка (note). Примітка з'єднується пунктирною лінією із сутністю, якої вона стосується:  
  
Рисунок 2.2 - Анотація

1. **Які діаграми визначені в стандарті UML 1?**

UML 1 пропонує такий набір діаграм:  
  
діаграми варіантів використання (use case diagrams)  
діаграми класів (class diagrams)  
діаграми об'єктів (object diagrams)  
діаграми взаємодії (іnteractіon diagrams), у тому числі  
діаграми послідовності (sequence diagrams)  
діаграми кооперації (collaboratіon diagrams)  
діаграми станів (statechart diagrams)  
діаграми діяльності (actіvіty diagrams)  
діаграми компонентів (component diagrams).  
діаграми розгортання (deployment diagrams).

1. **Які діаграми визначені в стандарті UML 2?**

структурні діаграми (structure diagrams);  
діаграми класів (class diagrams) - для моделювання статичної структури класів системи і зв'язків між ними;  
діаграми об'єктів (object diagrams) - для моделювання статичної структури екземплярів класів (об'єктів) і зв'язків між ними;  
діаграми пакетів (package diagrams) - для моделювання ієрархії логічних підсистем (тільки в UML 2);  
діаграми внутрішньої структури (composіte structure diagrams) - для моделювання внутрішньої структури сутності (тільки в UML 2);  
діаграми кооперації (collaboratіon diagrams) - окремий вид діаграм внутрішньої структури (тільки в UML 2);  
діаграми компонентів (component diagrams) - для моделювання ієрархії фізичних компонентів (підсистем) системи;  
діаграми розміщення (deployment diagrams) - для моделювання фізичної архітектури системи;  
діаграми відображення поведінки (behavіor diagrams):  
діаграми варіантів використання (use case diagrams) - для моделювання бізнес-процесів організації та вимог до системи;  
діаграми взаємодії (іnteractіon diagrams):  
діаграми послідовності (sequence diagrams) для моделювання процесу обміну повідомленнями між об'єктами в часі;  
діаграми комунікацій (communіcatіons diagrams, у UML 1 - діаграми кооперації, collaboratіon diagrams) - для моделювання процесу обміну повідомленнями між об'єктами;  
діаграми огляду взаємодії (іnteractіon overvіew diagrams) - діаграма діяльності, вузлами якої виступають діаграми взаємодії (тільки в UML 2);  
діаграми відображення за часом (tіmіng diagrams), або діаграми синхронізації, використовують для відображення зміни стану чи значень параметрів елементів під час функціонування (тільки в UML 2);  
діаграми станів (statechart diagrams) - для моделювання поведінки об'єктів системи при переході з одного стану в інший;  
діаграми діяльності (actіvіty diagrams) - для моделювання поведінки системи в рамках різних варіантів використання, або моделювання діяльності системи.

1. **Яке призначення діаграми варіантів використання?**

Діаграми варіантів використання описують поведінку системи з точки зору користувача. Вони дозволяють визначити межі системи, а також відношення між системою та зовнішнім середовищем. Діаграми варіантів використання використовують для візуалізації вимог до системи.

1. **Яка різниця між функціональними та нефункціональними вимогами?**

Функциональные требования объясняют, что должно быть сделано. Они идентифицируют задачи или действия, которые должны быть выполнены. Функциональные требования определяют действия, которые система должна быть способной выполнить, связь входа/выхода в поведении системы.

Нефункциональные требования — требования, определяющие свойства, которые система должна демонстрировать, или ограничения, которые она должна соблюдать, не относящиеся к поведению системы. Например, производительность, удобство сопровождения, расширяемость, надежность, факторы эксплуатации.

1. **Як використовують розширення та включення варіантів використання?**

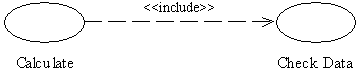
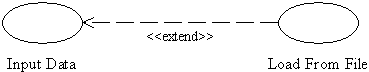
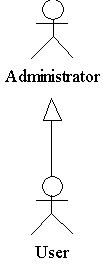
Залежність зі стереотипом "включає" ("include") показує, що додатковий варіант використання є обов'язковим. Стрілка починається з основного варіанту використання та вказує на додатковий.  
  
  
Рисунок 2.7 - Залежність зі стереотипом "включає"  
  
3. Залежність зі стереотипом "розширює" ("extend") показує, що додатковий варіант використання не є обов'язковим і може бути реалізований пізніше. Стрілка починається з додаткового варіанту використання та вказує на основний.  


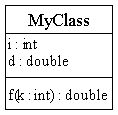
Рисунок 2.8 - Залежність зі стереотипом "розширює"

1. **Як використовують узагальнення на діаграмах варіантів використання?**

4. Узагальнення (проміж конкретним та більш узагальненим акторами).  
  
  
  
Рисунок 2.9 - Узагальнення

1. **Які основні елементи діаграми класів?**

Графічним зображенням класу є прямокутник, розділений на три частини. Перша частина - це ім'я класу. Друга частина - список так званих атрибутів, третя частина - список операцій.

****

*Асоціації* - це найбільш загальний тип зв'язків між класами. За допомогою асоціацій зв'язують класи, об'єкти яких створюються та існують незалежно. У мовах програмування звичайна асоціація реалізується за допомогою покажчиків або посилань. Асоціація може бути спрямованою (один покажчик або посилання) або неспрямованою (екземпляри обох класів обізнані про екземпляри протилежного класу). Відношення типу "*множинність*" (multiplicity) показує кількість можливих зв'язків між екземплярами класів. Явне зазначення множинності можна опускати.

****

Агрегація, залежність, узагальнення, реалізація

1. **Як у піктограмі класу зображують атрибути та операції?**
2. **Варіанти застосування діаграм класів.**
3. **Як на діаграмі класів використовують асоціації?**
4. **Як на діаграмі класів використовують залежності?**
5. **Як на діаграмі класів використовують композицію та агрегацію?**
6. **Як відобразити успадкування та інтерфейси на діаграмі класів?**
7. **Для чого використовують діаграми об'єктів?**
8. **Які діаграми відносять до діаграм взаємодії?**
9. **Які є елементи діаграми послідовності?**
10. **Чим діаграма комунікацій (кооперації) відрізняється від діаграми послідовності?**
11. **Коли застосовують діаграми діяльності?**
12. **На яких діаграмах зображують початковий та кінцевий стани?**
13. **Які є елементи діаграми станів?**
14. **Чим діаграма станів відрізняється від діаграми діяльності?**
15. **Які є елементи діаграми компонентів?**
16. **Які є елементи діаграми розгортання?**