

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені ТАРАСА ШЕВЧЕНКА**



ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра прикладних інформаційних систем

Звіт до лабораторної роботи №4

3 курсу

«Інженерія програмного забезпечення»

студента 2 курсу

групи ПП-22

спеціальності 122 «Комп'ютерні науки»

ОП «Прикладне програмування»

Шевлюк Вікторії Віталіївни

Перевірила:

Доц. Бойко Юлія Петрівна

Київ 2022

Тема: Синтез обов'язків, атрибутів і асоціацій класів

Мета роботи – вивчення додавання атрибутів класам, додавання зв'язків між сутностями та додавання класів-асоціацій.

Хід роботи:

Вправа 19. Додавання атрибутів к класам

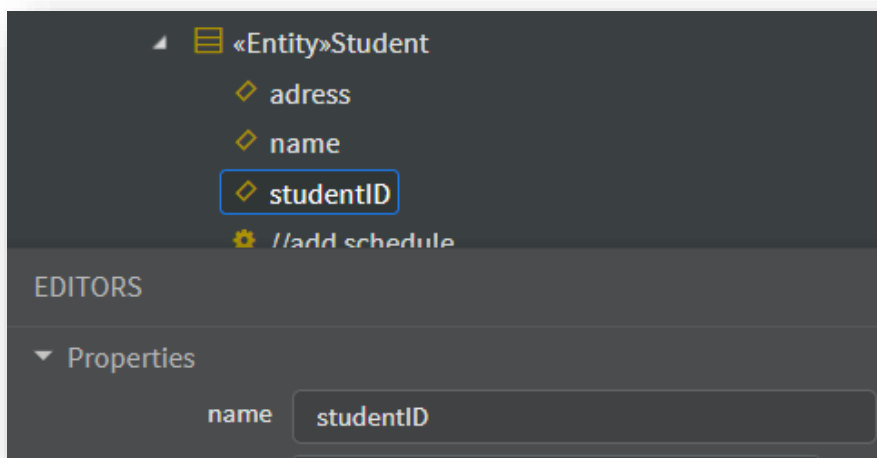
Атрибути класів аналізу визначаються, виходячи зі знань про предметну область, вимог до системи і глосарію.

Налаштування

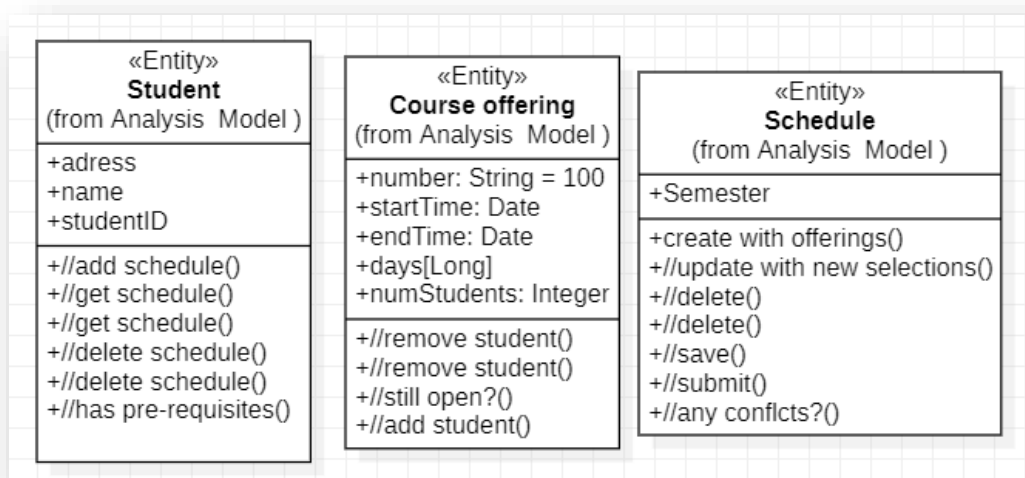
1. Вікно діаграми класів VOPC (classes only) повинно бути відкрито.
2. В меню моделі оберіть пункт *Tools/ Options*.
3. Перейдіть на вкладку Diagram.
4. Переконайтесь, що перемикач ☐ Show All Attributes відмічено.
5. Переконайтесь, що перемикачі ☐ Suppress Attributes и ☐ Suppress Operations не відмічені.

Додавання атрибутів

6. Натисніть в браузері правою кнопкою миші на класі Student.
7. У меню, виберіть пункт New / Attribute.
8. Введіть новий атрибут adress замість name.
9. Натисніть Enter.
10. Повторіть кроки 1 - 4, додавши атрибути name і studentID.










11. Натисніть правою кнопкою миші на класі CourseOffering.
12. У меню, виберіть пункт Open Specification Відкриється вікно Class Specification. Перейдіть на вкладку Attributes.
13. Натисніть правою кнопкою миші на порожньому білому списку.
14. У меню, виберіть пункт Insert. З'явиться рядок параметрів атрибута name.
15. На рядку зліва зробіть подвійне клацання. З'явиться вікно Class Attribute
16. У вікні задайте параметри атрибута: Name = number, Type = String, Initial Value = "100" .Натисніть ОК.
17. Повторіть кроки 8 - 11, додавши атрибути startTime, endTime, days та numStudents з типами по рис. 27.
18. Натисніть ОК.

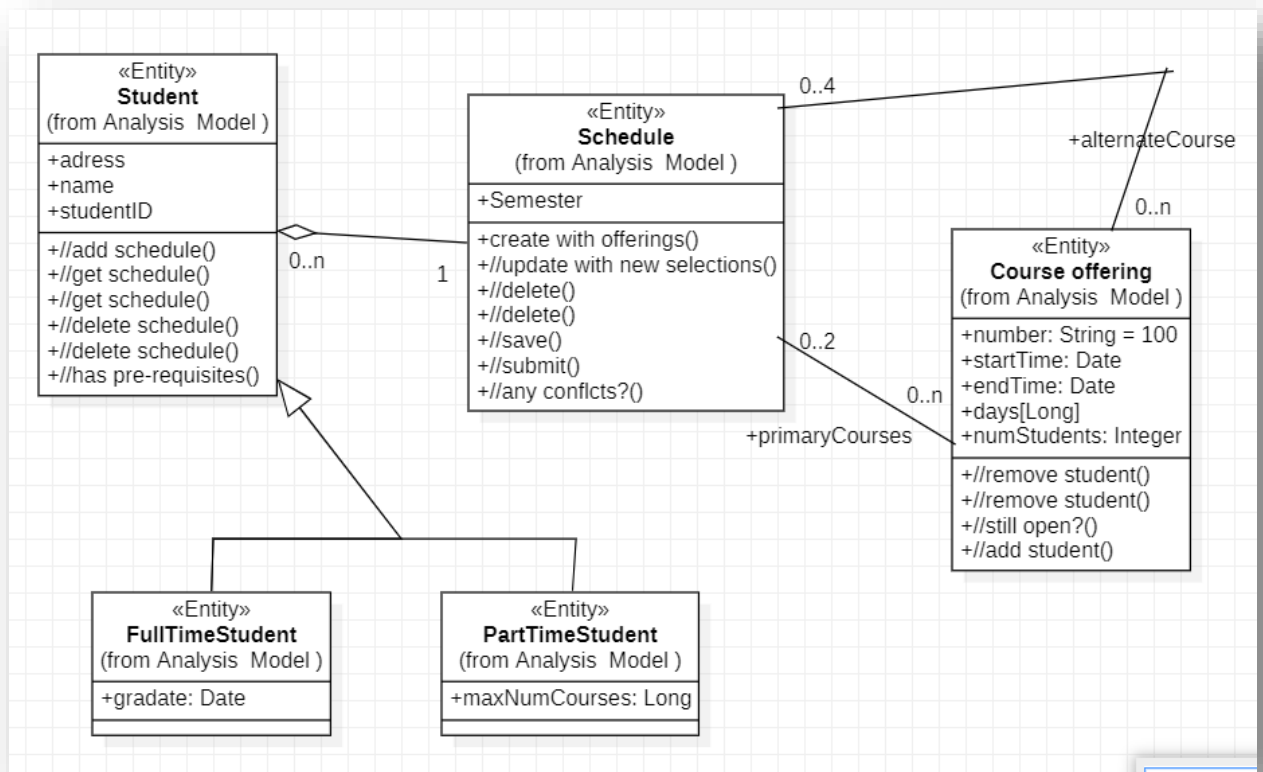


Вправа 20. Додавання зв'язків між сутностями

1. Додамо зв'язку до класів, які беруть участь у варіанті використання Register for Courses. Для відображення зв'язків між класами побудуємо три нових діаграми класів в кооперації Register for Courses пакета Use-Case Realization - Register for Courses.

2. В кооперації Register for Courses створіть діаграму класів з ім'ям EntityClasses. Відкрийте її вікно.

3. В пакеті  Analysis Model логічного представлення створіть клас  FulltimeStudent со стереотипом entity.
- Права кнопка миші, меню *New, Class*
 - Задати ім'я класу FulltimeStudent
 - Права кнопка миші, меню *Open Specification*. З'явиться вікно **Class Specification**.
- На вкладці General задайте Stereotype= entity.
 - На вкладці Attributes додайте атрибут з ім'ям Name = gradate с типом Type=Date.
 - Натисніть двічі ОК.
4. В пакеті  Analysis Model аналогічно пункту 4 створіть клас  ParttimeStudent зі стереотипом entity і одним атрибутом maxNumCourses типу Long.
5. Перетягніть у вікно п'ять класів сутностей.
6. За допомогою панелі діаграм, кнопок , ,  побудуйте зв'язки між класами.
7. Натисніть двічі на прямій лінії зв'язку асоціації між CourseOffering і Schedule. З'явиться вікно **Association Specification**.
8. На вкладці Role A Detail задайте поле Multiplicity=0..n, на вкладці Role B Detail задайте поля Role= primaryCourses і Multiplicity=0..4. Натисніть ОК.
9. Аналогічно пунктам 8, 9 задайте поля Role і Multiplicity ще для двох зв'язок асоціації і агрегації.






Збережіть модель *File/ Save*. Збережіть модель в своєму каталозі за допомогою пункту меню *File/ Save As* під новим ім'ям Прізвище5. Четвертий етап створення моделі залишиться в файлі Прізвище4.

Агрегація між класами **Student** і **Schedule** показує той факт, що кожен графік є власністю конкретного студента, належить тільки йому.

Передбачається також, що в системі буде зберігатися не тільки графік поточного семестру, а всі графіки студента за різні семестри. Між класами **Schedule** і **CourseOffering** введено дві асоціації, оскільки конкретний курс може входити в графік студента в якості основного (не більше чотирьох курсів) і альтернативного (не більше двох курсів). До класу **Student** додані два нових підкласи - **FulltimeStudent** (студент очного відділення) і **ParttimeStudent** (студент вечірнього відділення).

Вправа 21. Додавання класів-асоціацій

1. В кооперації Register for Courses створіть діаграму класів з іменем  CourseOfferingInfo. Відкрийте її вікно.


2. В пакеті  Analysis Model натисніть двічі на класі сутності  PrimaryScheduleOfferingInfo.

3. Відкриється вікно **ClassSpecification**.


4. На вкладці Attributes додайте атрибут з ім'ям Name = grade.


5. На вкладці Operations додайте три операції з назвами Operation: //is enrolled in?, //mark as enrolled in, //mark as committed.


6. Натисніть ОК.

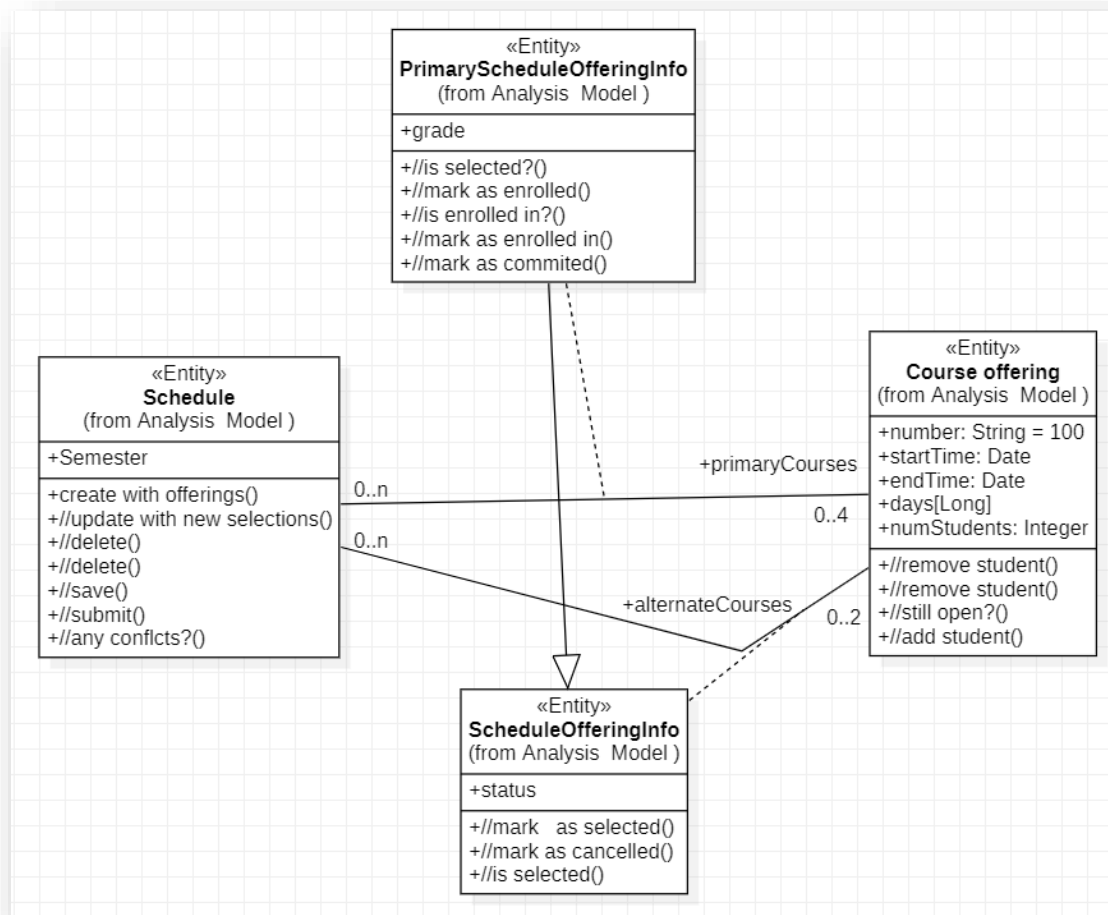
7. В пакеті  Analysis Model створіть клас з ім'ям ScheduleOfferingInfo. Аналогічно пунктам 2-6 задайте йому стереотип entity, один атрибут status, три операції: //mark as selected, //mark as cancelled, //is selected.

8. Перетягніть класи CourseOffering, Schedule, PrimaryScheduleOfferingInfo, ScheduleOfferingInfo на вікно діаграми CourseOfferingInfo. Відмітимо, що між класами CourseOffering і Schedule з'являться дві введені нами раніше асоціації. На діаграмі вони злилися в одну лінію.

Між класами PrimaryScheduleOfferingInfo і ScheduleOfferingInfo проведемо зв'язок узагальнення .

Між класами PrimaryScheduleOfferingInfo і асоціацією primaryCourses проведемо зв'язок асоціативного класу .

Між класом ScheduleOfferingInfo і асоціацією alternateCourses проведемо зв'язок асоціативного класу .



Вправа 22. Побудова повної діаграми класів

1. Через меню *Tools/Options*, вкладка *Diagram*, *Stereotype Display* ☐ Ікон зображення всіх класів зробить у вигляді іконок.

2. В кооперації *Register for Courses* створіть діаграму класів з іменем **VOPC full**. Відкрийте її вікно.

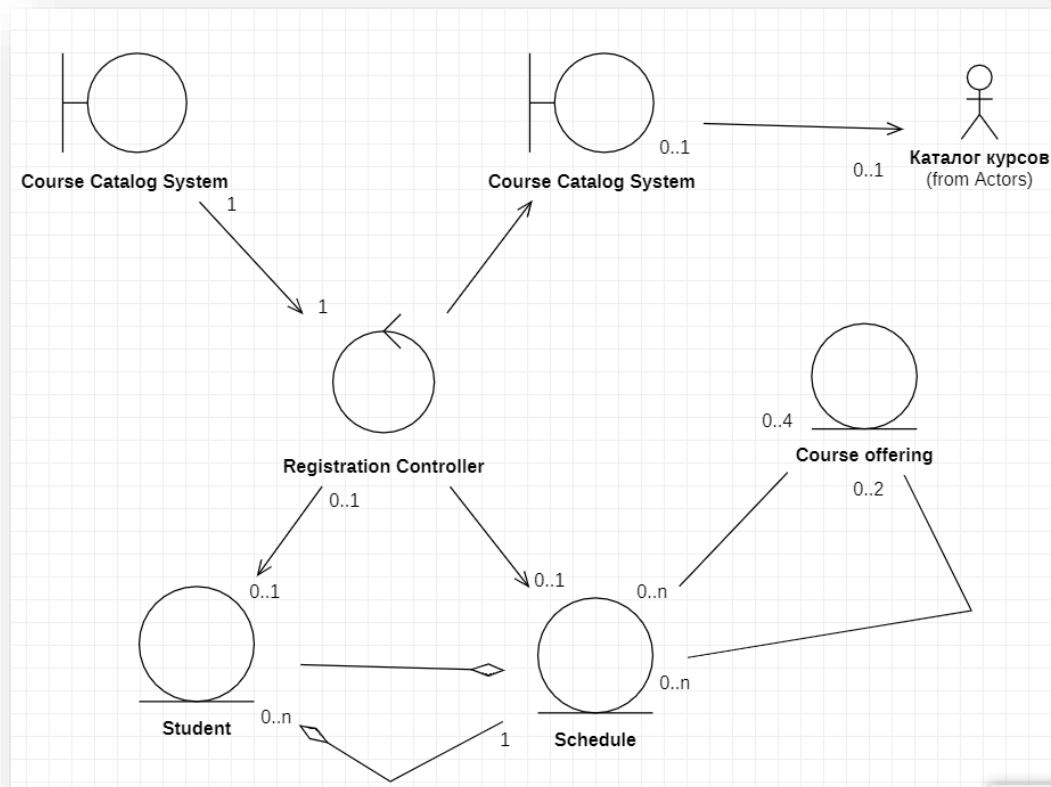
3. Перетягніть на нього шість класів *RegisterForCoursesForm*, *CourseCatalogSystem*, *RegistrationController*, *CourseOffering*, *Student*, *Schedule* і одного актора *Каталог курсів*, вказаних на рис. 30. Зв'язки між деякими з них з'являться самостійно.

4. Створіть на діаграмі відсутні зв'язки односпрямованої асоціації.

5. Через подвійний натиск на зв'язку, вікно **Association Specification**, вкладки *Role A(B)Detail* задаємо параметри ролі (Role) і потужності

(Multiplicity) на всіх зв'язках.

На рис показана повна діаграма класів VOPC full варіанти використання «Зареєструватися на курси» (без атрибутів і операцій). Асоціацію між граничними і керуючими класами, а також між керуючими класами і класами-сутностями введені на основі аналізу кооперативних діаграм. На відміну від стійких структурних і семантичних зв'язків між сутностями вони відображають зв'язку, динамічно що виникають між відповідними об'єктами в потоці управління або в процесі роботи програми. Оскільки для асоціацій це не властиво, в подальшому (в процесі проектування) вони можуть бути перетворені в залежності.



Висновок: у ході даної лабораторної роботи я вивчила додавання атрибутів класам, додавання зв'язків між сутностями та додавання класів-асоціацій у середовищі StarUml.