

## Лабораторна робота № 8

**Мета:** набуття практичних навичок щодо кодогенерації визначень додатку в середовищі розробки.

### Хід роботи

#### Завдання 8.1. Створення UML-моделі через Реверс-інжиніринг

Замість ручного створення діаграми класів, ми використаємо функцію реверс-інжинірингу (Reverse Engineering) CASE-інструменту Visual Paradigm, щоб автоматично згенерувати UML-модель на основі існуючої DDL-схеми (файлу .sql).

#### Процес:

1. У Visual Paradigm ми запускаємо інструмент Tools > Code & DB > Reverse Database....
2. Обираємо цільову базу даних MySQL та запускаємо процес.

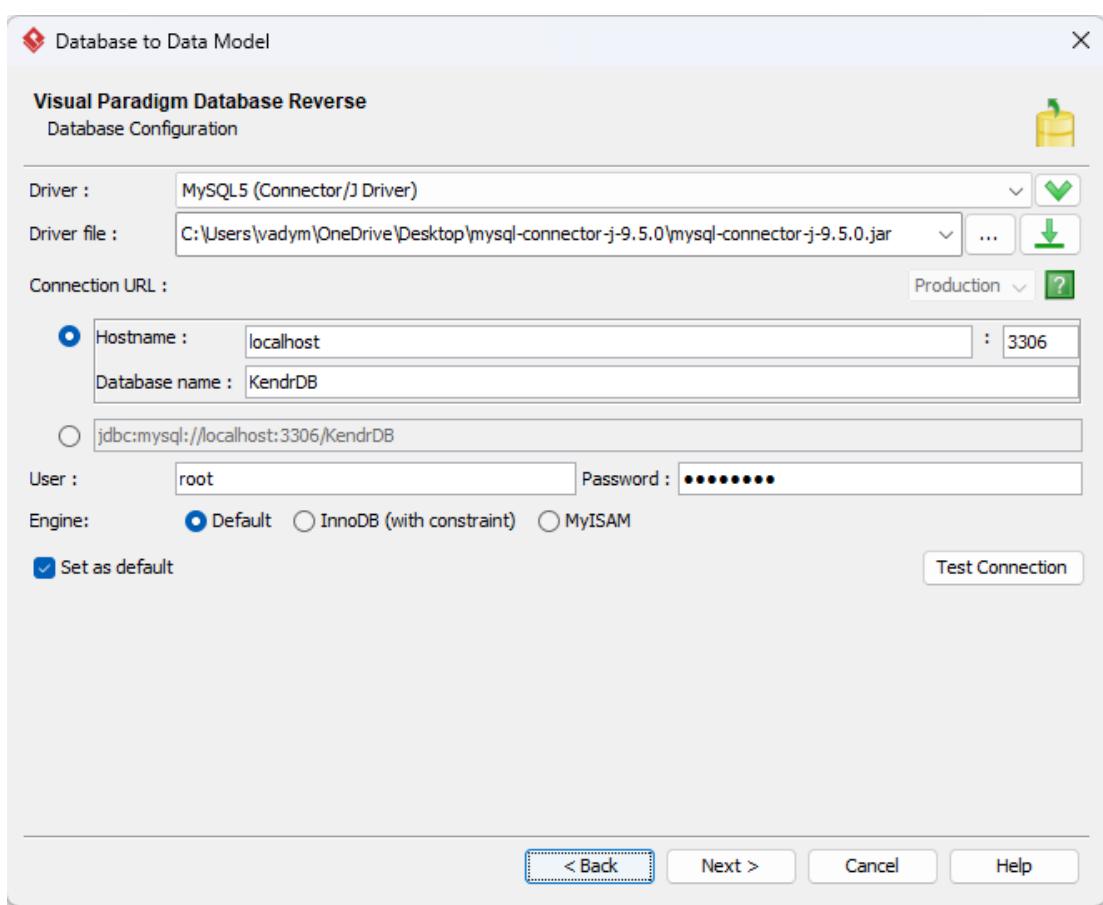


Рис. 8.1. Налаштування з'єднання для зворотного проектування бази даних

Змн.	Арк.	№ докум.	Підпис	Дата	ДУ «Житомирська політехніка».25.121.19.000–Пр8		
Розроб.	Леус В.О.				Звіт з лабораторної роботи		
Перевір.	Левківський В.Л.						
Керівник					<b>ФІКТ Гр. ІПЗ-22-3</b>		
Н. контр.							
Зав. каф.					Lіт.	Арк.	Аркушів
						1	5

Програма аналізує SQL-файл і автоматично створює повну UML-діаграму класів, що точно відповідає структурі бази даних, включно з усіма атрибутами, типами даних та зв'язками (FOREIGN KEY).

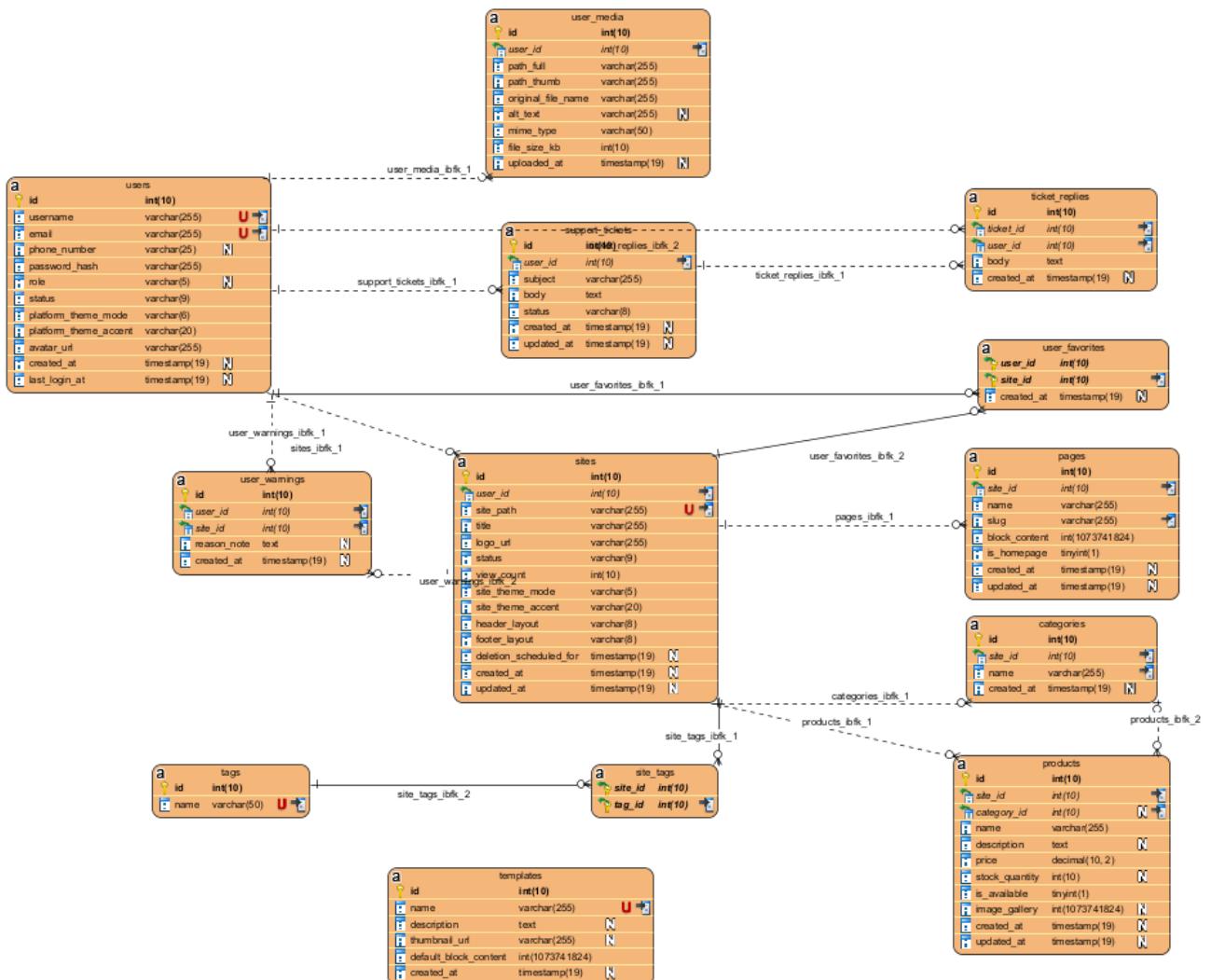


Рис. 8.2. Діаграма класів, автоматично згенерована з SQL-файлу

## **Завдання 8.2. Генерація коду Java**

Тепер, коли ми маємо готову UML-модель (отриману в Завданні 8.1), ми можемо використати її для генерації "каркасів" коду (Пряме проектування).

1. Ми обираємо всі класи, згенеровані на діаграмі, у "Model Explorer".
  2. Через контекстне меню запускаємо Code > Instant Generator....
  3. У вікні генератора обираємо мову Java та вказуємо шлях для збереження файлів.

		Леус В.О.				
		Левківський В.Л.				
Змн.	Арк.	№ докум.	Підпис	Дата	ДУ «Житомирська політехніка».25.121.19.000 – Пр8	Арк. 2

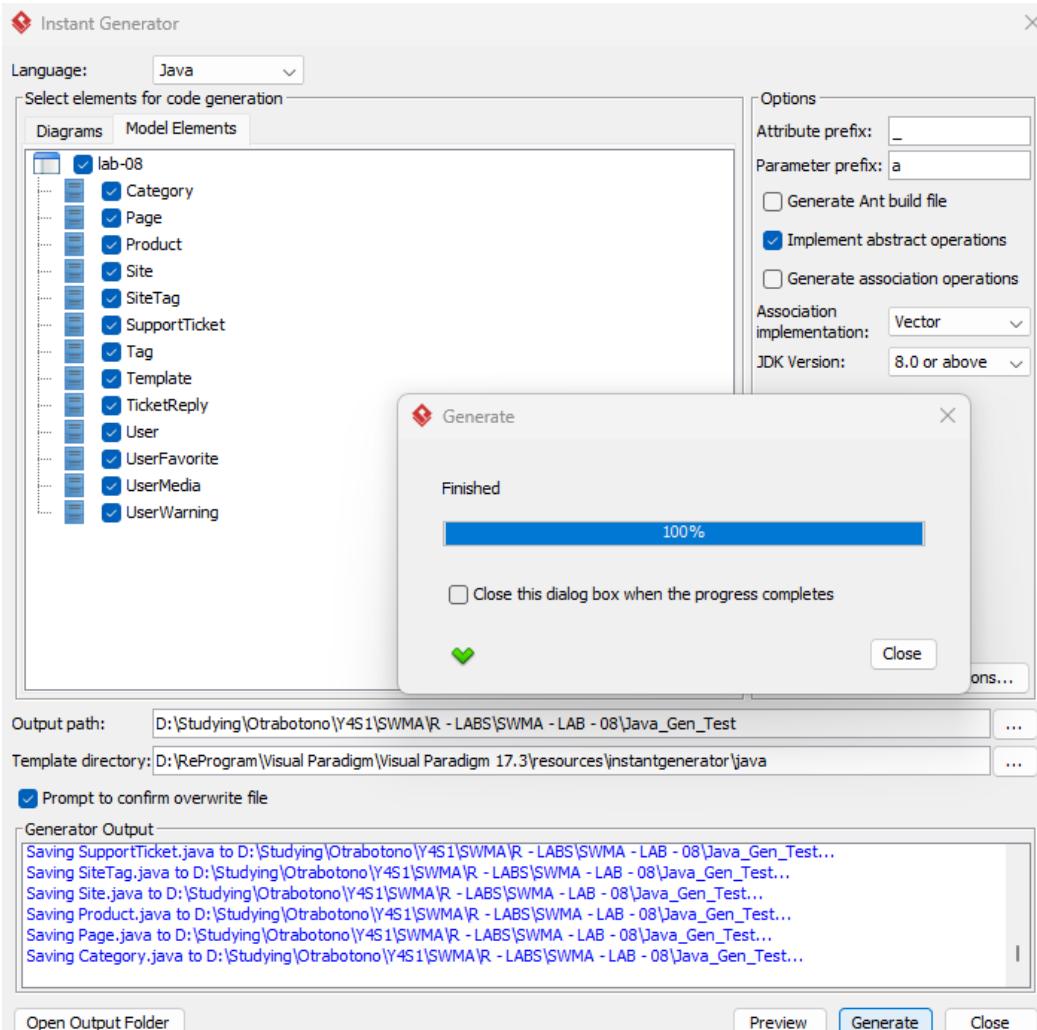


Рис. 8.1. Налаштування генератора коду (Instant Generator) для Java

Після запуску генератора, програма автоматично створює .java файли для всіх обраних класів у вказаній директорії.



Рис. 8.2. Згенеровані файли класів

Приклад згенерованого коду Category.java:

```
import java.sql.Timestamp;

public class Category {
    private Integer _id;
    private Integer _siteId;
    private String _name;
    private String _description;
    private Integer _sortOrder;
    private Timestamp _createdAt;

    public Category() {
        throw new UnsupportedOperationException();
    }
}
```

Леус В.О.	Левківський В.Л.	ДУ «Житомирська політехніка».25.121.19.000 – Пр8	Арк.
Змн.	Арк.	№ докум.	Підпис
Дата			3

```

    }

    public void setId(Integer aId) {
        this._id = aId;
    }

    public Integer getId() {
        return this._id;
    }

    public void setSiteId(Integer aSiteId) {
        this._siteId = aSiteId;
    }

    public Integer getSiteId() {
        return this._siteId;
    }

    public void setName(String aName) {
        this._name = aName;
    }

    public String getName() {
        return this._name;
    }

    public void setDescription(String aDescription) {
        this._description = aDescription;
    }

    public String getDescription() {
        return this._description;
    }

    public void setSortOrder(Integer aSortOrder) {
        this._sortOrder = aSortOrder;
    }

    public Integer getSortOrder() {
        return this._sortOrder;
    }

    public void setCreatedAt(Timestamp aCreatedAt) {
        this._createdAt = aCreatedAt;
    }

    public Timestamp getCreatedAt() {
        return this._createdAt;
    }
}

```

Приклад згенерованого коду User.java:

```

import java.sql.Timestamp;

public class User {
    private Integer _id;
    private String _username;
    private String _email;
    private String _role;
    private String _avatarUrl;
    private Timestamp _createdAt;
    private Timestamp _updatedAt;

    public User() {
        throw new UnsupportedOperationException();
    }

    public void setId(Integer aId) {
        this._id = aId;
    }

    public Integer getId() {
        return this._id;
    }

    public void setUsername(String aUsername) {

```

		Леус В.О.			Арк.
		Левківський В.Л.			
Змн.	Арк.	№ докум.	Підпис	Дата	ДУ «Житомирська політехніка».25.121.19.000 – Пр8 4

```

        this._username = aUsername;
    }

    public String getUsername() {
        return this._username;
    }

    public void setEmail(String aEmail) {
        this._email = aEmail;
    }

    public String getEmail() {
        return this._email;
    }

    public void setRole(String aRole) {
        this._role = aRole;
    }

    public String getRole() {
        return this._role;
    }

    public void setAvatarUrl(String aAvatarUrl) {
        this._avatarUrl = aAvatarUrl;
    }

    public String getAvatarUrl() {
        return this._avatarUrl;
    }

    public void setCreatedAt(Timestamp aCreatedAt) {
        this._createdAt = aCreatedAt;
    }

    public Timestamp getCreatedAt() {
        return this._createdAt;
    }

    public void setUpdatedAt(Timestamp aUpdatedAt) {
        this._updatedAt = aUpdatedAt;
    }

    public Timestamp getUpdatedAt() {
        return this._updatedAt;
    }
}

```

**Висновок:** в ході виконання лабораторної роботи були набуті практичні навички щодо кодогенерації визначень додатку в середовище розробки.

Змн.	Арк.	Леус В.О.			ДУ «Житомирська політехніка».25.121.19.000 – Пр8	Арк.
		Левківський В.Л.				5