

Ви не авторизовані

[Увійти](#) | [Реєстрація](#)

Реєстрація

Введіть логін

Введіть email

Введіть пароль

Вхід

Зареєструватись До входу

[Реєстрація](#)

Привіт, xcursi

[Вийти](#) | [Додати матеріал](#)

Ваші матеріали

У вас ще немає доданих матеріалів.

Додати матеріал

Введіть назву матеріалу

Введіть категорію

Введіть ціну

Введіть кількість

Додати Назад

Привіт, xcursi

[Вийти](#) | [Додати матеріал](#)

Ваші матеріали

Назва	Категорія	Ціна	Кількість	Дії
Бетон	Суміші	1000.00	100	Редагувати Видалити

Редагування матеріалу

Назва:

Бетон

Категорія:

Суміші

Кількість:

100

Ціна:

1000,00

[Зберегти](#)

[Назад](#)

Контрольні питання

- Що таке модель в архітектурі MVC (Model-View-Controller)?
Це частина програми, яка відповідає за роботу з даними та бізнес-логіку
- Як моделі взаємодіють із базою даних у веб-розробці?
Зазвичай використовують ORM або прямі SQL-запити
- Чому важливо використовувати моделі для роботи з даними, а не безпосередньо вбудовані запити до бази даних?
Централізація логіки, безпека

4. Як можна визначити та викликати методи моделі для обробки даних?

```
class User extends Model {  
    public function getActiveUsers() {  
        return $this->where('active', 1)->get();  
    }  
}
```

```
$activeUsers = User::getActiveUsers();
```

5. Як в моделях обробляються операції створення, оновлення та видалення записів в базі даних?
Через CRUD методи

6. Як в моделях вирішуються проблеми з безпекою та попередженням атак типу SQL Injection?

ORM автоматично екранує дані, підставляючи їх у запити без небезпечних вставок

7. Чому важливо визначати логіку бізнес-правил в моделях, а не у контролерах чи виду?

Контролери повинні керувати потоком запитів і викликами моделей, а не знати деталі бізнесу