1. Як працюють індекси у БД?

Індекси дозволяють оптимізувати роботу з базою – дані сортуються і СУБД не доводиться шукати значення серед усіх записів. На практиці це може скоротити час обробки запитів на кілька порядків.

2. Які є JOIN в MySQL?

## Inner join, left join, right join, full join.

3. Що таке dependency inversion?

Dependency Inversion Principle – принцип інверсії залежностей. Модулі верхнього рівня повинні залежати від модулів нижнього рівня. І ті, й інші мають залежати від абстракції. Абстракції не повинні залежати від деталей. Деталі мають залежати від абстракцій.

4. У чому різниця між GET та POST запитом?

Надсилання параметра запиту:

Метод Get передає параметр запиту у рядку URL.

Метод POST передає параметр запиту в тілі запиту.

Переданий обсяг даних:

Метод Get може передавати лише обмежений обсяг даних.

Метод Post може передавати велику кількість даних.

Закладка / Cache:

Метод Get полегшує створення закладок та кеш-пам'яті.

Метод Post не надає таких можливостей.

Мета:

Метод Get переважно використовується для перегляду.

Метод Post особливо використовується для оновлення.

Довжина:

Дані, що надсилаються методом Get, зазвичай дуже короткі.

Дані, що надсилаються методом Зщіе, зазвичай довші, ніж методом get.

Швидкість:

Оскільки метод get не потребує великої кількості даних, він швидкий.

Post повільніший, ніж метод get.

5. У чому різниця дзвінка self і static?

Коли ми використовуємо ключове слово self у класі, воно завжди буде ставитись до того класу, в якому воно згадується, а не до класу, що його успадковує.

Коли ми використовуємо ключове слово static, ми посилаємося на клас, який викликається під час виконання.

6. Яка різниця між MVP та MVC?

MVC - це патерн, створений з метою розбиття будь-якої програми, з якою взаємодіє користувач, на три логічні блоки:

Model - це бізнес-логіка програми, в ньому відбуваються найважчі процеси (обробка, сортування, запис даних, звернення до бази даних, генерація веб-сторінки і т.п.), і після виконаної події відправляє результат у View.

View – це інтерфейс, з яким взаємодіє користувач, візуальна частина (наприклад, сторінка сайту). Завдання View - рахувати інформацію, введену користувачем, і передати її в конкретний метод або функцію контролера, а також зчитувати дії користувача (натискання кнопки або оновлення сторінки, що теж викликає метод або функцію з контролера).

Controller — приймає дані або подію від View і може діяти таким чином, наприклад: за допомогою винятків вибирає конкретний метод Model, який потрібно викликати, спираючись на дані чи подію від View; формує об'єкт із отриманих параметрів, який ми передаємо Model за допомогою методу, функції Model або створюємо http-запит до Model.

Загалом у нас виходить циркуляційна взаємодія наших трьох модулів — тобто по колу.

MVP:

Замість модуля Controller використовується модуль Presenter. Модулі View і Model спілкуються виключно через модуль Presenter, який є посередником: робить все, що робив Controller, звертається до Model, чекає на відповідь, отримує її і передає результат View.

7. Яка різниця між == та ===?

== оператор простого порівняння;

=== оператор тотожного порівняння з перевіркою типу.

8. Що таке final class та final method?

Розмістивши final перед оголошеннями методів чи констант класу, можна запобігти їх перевизначення у дочірніх класах.

9. Якими фреймворками працювали?

Zend framework, Symfony framework.

10. Що можете розповісти про свої навички, що дають вам перевагу? Які з них відповідають нашим вимогам?

Маю практичні навички beckend розробки Web застосунків в системах SFCC, Magento, BigCommerce та досвід роботи з ecommerce, що підтверджує володіння мовами програмування JavaScript, PHP, SQL та основними принципами ООП. Працюю HTML, CSS, JQuery та з протоколами взаємодії HTTP, HTTPS та RESTful API. Є практичний досвід роботи з: Visual Code, PHP Storm, Postman, Dbeaver, Git, Confluence, Jira.