

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Косенок Сергей Михайлович  
Должность: ректор  
Дата подписания: 16.06.2025 12:30:13  
Уникальный программный ключ:  
e3a68f38-4a62674b54f4998099d13d6b1dcf836

## Тестовое задание для диагностического тестирования по дисциплине:

*Базы данных, 5 семестр*

|                             |                                              |
|-----------------------------|----------------------------------------------|
| Код, направление подготовки | 01.03.02 Прикладная математика и информатика |
| Направленность (профиль)    | Технологии программирования и анализ данных  |
| Форма обучения              | очная                                        |
| Кафедра-разработчик         | Кафедра прикладной математики                |
| Выпускающая кафедра         | Кафедра прикладной математики                |

| Проверяемая компетенция                                       | Задание                                                                                                                                                         | Варианты ответов                                                                                                                                                                                                                                                                                                      | Тип сложности вопроса |
|---------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 1. Отношение находится во второй нормальной форме, если оно находится в _____ нормальной форме и каждый не ключевой атрибут неприводимо зависит от _____ ключа. |                                                                                                                                                                                                                                                                                                                       | Низкий                |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 2. Как выглядит запрос, для вывода ВСЕХ значений из таблицы Orders:                                                                                             | 1. select ALL from Orders;<br>2. select % from Orders;<br>3. select * from Orders;<br>4. select *.Orders from Orders;                                                                                                                                                                                                 | Низкий                |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 3. Какие данные мы получим из этого запроса?<br>select id, date, customer_name from Orders;                                                                     | 1. Неотсортированные номера и даты всех заказов с именами заказчиков<br>2. Никакие, запрос составлен неверно<br>3. Номера и даты всех заказов с именами заказчиков, отсортированные по первой колонке<br>4. Номера и даты всех заказов с именами заказчиков, отсортированные по всем колонкам, содержащим слово Order | Низкий                |

|                                                               |                                                                                                                                                                                 |                                                                                                                                                                                                                                                                                                                          |         |
|---------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 4. Нормальная форма – требование, предъявляемое к структуре таблиц в теории _____ баз данных, для устранения из базы данных _____ функциональных зависимостей между атрибутами. |                                                                                                                                                                                                                                                                                                                          | Низкий  |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 5. Что покажет следующий запрос: select DISTINCT seller_id order by seller_id from Orders;                                                                                      | 1. Уникальные ID продавцов, отсортированные по возрастанию<br>2. Уникальные ID продавцов, отсортированные по убыванию<br>3. Ничего, запрос составлен неверно, ORDER BY всегда ставится в конце запроса<br>4. Неотсортированные никак уникальные ID продавцов                                                             | Низкий  |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 6. Что делает спецсимвол '_' в паре с оператором LIKE: select * from Orders where customer_name like 'mik_';                                                                    | 1. Найдет все имена, которые начинаются на mik и состоят из 4 символов<br>2. Найдет все имена, которые начинаются на mik, вне зависимости от того, из какого количества символов они состоят<br>3. Найдет данные, где имя равно mik<br>4. Запрос составлен неверно, в паре с оператором like не используются спецсимволы | Средний |

|                                                               |                                                                                                            |                                                                                                                                                                                                                                                                                                            |         |
|---------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 7. Что покажет следующий запрос:<br>select concat(`index`,` `, `city`)<br>AS delivery_address from Orders; | 1. Ничего, запрос составлен неверно<br>2. Покажет уникальные значения индексов и адресов из таблицы Orders<br>3. Соединит поля с индексом и адресом из таблицы Orders и покажет их с псевдонимом delivery_address<br>4. Соединит поля с индексом и адресом из таблицы Orders, но покажет их без псевдонима | Средний |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 8. Для чего используется LIMIT:<br>select * from Orders limit 10;                                          | 1. Необходимо, чтобы показать все заказы, содержащие цифру 10<br>2. Необходимо, чтобы показать первых 10 записей в запросе<br>3. Необходимо, чтобы показать случайные 10 записей в запрос<br>4. не существует такого оператора                                                                             | Средний |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 9. Выберите пример правильно составленного запроса с использованием агрегирующей функции SUM:              | 1. select sum(price) from Orders;<br>2. select sum(price), customer_name from Orders;<br>3. select * from Orders where price=sum();<br>4. select sum() from Orders group by price desc;                                                                                                                    | Средний |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 10. Выберите корректно составленный запрос с функцией GROUP BY:                                            | 1. select count(*) from Orders GROUP seller_id;<br>2. select seller_id, count(*) from Orders GROUP seller_id;<br>3. select seller_id, count(*) from Orders GROUP BY seller_id;<br>4. select count(*) from Orders GROUP ON seller_id;                                                                       | Средний |

|                                                               |                                                                                                                                                |                                                                                                                                                                                                                                                                                             |         |
|---------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 11. Что покажет следующий запрос: select seller_id, count(*) from Orders GROUP BY seller_id HAVING seller_id IN (2,4,6);                       | 1. Количество заказов, сгруппированное по продавцам 2, 4 и 6<br>2. Количество продавцов, у которых 2, 4 или 6 товаров<br>3. Ничего, запрос составлен неверно, HAVING указывается до группировки<br>4. Ничего, запрос составлен неверно, для указания условия должно быть использовано WHERE | Средний |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 12. Выберите пример корректно написанного запроса с использованием подзапроса, который выводит информацию о заказе с самой дорогой стоимостью: | 1. select * from Orders where price = (select big(price) from Orders)<br>2. select * from Orders where price = max<br>3. select count(*) from Orders<br>4. select * from Orders where price = (select max(price) from Orders)                                                               | Средний |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 13. Выберите операторы языка SQL                                                                                                               | 1. DECERT<br>2. SELECT<br>3. INSERT<br>4. DETECT                                                                                                                                                                                                                                            | Средний |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 14. Установите соответствие между понятиями теории реляционных БД и их определениями                                                           | 1. Строка таблицы<br>1. Кортеж<br>2. Заголовок столбца таблицы<br>2. Атрибут<br>3. Количество кортежей в отношении<br>3. Кардинальность                                                                                                                                                     | Средний |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 15. Какие поля из таблицы обязательно перечислять в INSERT для вставки данных?                                                                 | 1. Все<br>2. Только те, у которых нет DEFAULT значения<br>3. Те, у которых нет DEFAULT значения и которые не имеют атрибут auto_increment                                                                                                                                                   | Средний |

|                                                               |                                                                                 |                                                                                                                                                                                                                                                                                             |         |
|---------------------------------------------------------------|---------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
|                                                               |                                                                                 | 4. Все поля имеют негласное DEFAULT значения, обязательных полей в SQL нет                                                                                                                                                                                                                  |         |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 16. В каких командах можно использовать LIMIT?                                  | 1. Только Select<br>2. Select и Insert<br>3. Select, Update, Delete<br>4. Select, Insert, Delete, Update                                                                                                                                                                                    | Высокий |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 17. Как можно заранее узнать, какие записи будут удалены при выполнении DELETE? | 1. Зачем заранее, просто вызвать его и посмотреть какие записи пропали<br>2. Заменить DELETE на SELECT *, ведь в остальном синтаксис DELETE похож на синтаксис простого SELECT<br>3. Сделать DELETE с LIMIT 1, одну запись не жалко<br>4. SQL создан для хранения данных, их нельзя удалять | Высокий |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 18. Какой командой можно создать новую таблицу?                                 | 1. CREATE TABLE<br>2. MAKE TABLE<br>3. SET TABLE<br>4. Создавать таблицы можно только через интерфейс СУБД, специальной SQL команды для этого нет                                                                                                                                           | Высокий |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 19. Можно ли поменять тип данных поля в уже существующей таблице?               | 1. Да, при помощи команды ALTER<br>2. Да, достаточно сделать INSERT с новым типом данных<br>3. Нет, только пересоздать таблицу<br>4. Тип бывает только у таблицы, а не у поля таблицы                                                                                                       | Высокий |
| ПК-4.1,<br>ПК-4.2,<br>ПК-4.3,<br>ПК-3.1,<br>ПК-3.2,<br>ПК-3.3 | 20. Выберите элементы реляционной модели                                        | 1. Домен<br>2. Запрос<br>3. Отношение<br>4. Кардинальность                                                                                                                                                                                                                                  | Высокий |