Тестовое задание для диагностического тестирования по дисциплине:

Базы данных

Код, направление подготовки	ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ
Направленность (профиль)	Информационные системы и технологии
Форма обучения	Очная
Кафедра-разработчик	Информатики и вычислительной техники
Выпускающая кафедра	Информатики и вычислительной техники

Проверяемая компетенция	Задание	Варианты ответов	Тип сложности вопроса
ОПК-3.3 ОПК-7.1	1. Что такое реляционные базы данных:	1. База данных, в которой информация хранится в виде двумерных таблиц, связанных между собой 2. База данных, в которой одна ни с чем не связанная таблица 3. Любая база данных - реляционная 4. Совокупность данных, не связанных между собой	Низкий
ОПК-3.3 ОПК-7.1	2. Как выглядит запрос, для вывода ВСЕХ значений из таблицы Orders:	1. select ALL from Orders; 2. select % from Orders; 3. select * from Orders; 4. select *.Orders from Orders;	Низкий
ОПК-3.3 ОПК-7.1	3. Какие данные мы получим из этого запроса? select id, date, customer_name from Orders;	1. Неотсортированные номера и даты всех заказов с именами заказчиков 2. Никакие, запрос составлен неверно 3. Номера и даты всех заказов с именами заказчиков, отсортированные по первой колонке 4. Номера и даты всех заказов с именами заказчиков, отсортированные по всем колонкам, содержащим слово Order	Низкий
ОПК-3.3 ОПК-7.1	4. Что покажет следующий запрос: select * from Orders where	1. Все данные по заказам, совершенным за 2017 год,	Низкий

	date between '2017-01-01' and '2017-12-31'	за исключением 01 января 2017 года 2. Все данные по заказам, совершенным за 2017 год, за исключением 31 декабря 2017 года 3. Все данные по заказам, совершенным за 2017 год 4. Ничего, запрос составлен неверно	
ОПК-3.3 ОПК-7.1	5. Что покажет следующий запрос: select DISTINCT seller_id order by seller_id from Orders;	1. Уникальные ID продавцов, отсортированные по возрастанию 2. Уникальные ID продавцов, отсортированные по убыванию 3. Ничего, запрос составлен неверно, ORDER BY всегда ставится в конце запроса 4. Неотсортированные никак уникальные ID продавцов	Низкий
ОПК-3.3 ОПК-7.1	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 2. найдет все имена, которые начинаются на mik, вне зависимости от того, из какого количества символов они состоят 3. найдет данные, где имя равно mik 4. запрос составлен неверно, в паре с оператором like не используются спецсимволы	Средний
ОПК-3.3 ОПК-7.1	7. Что покажет следующий запрос: select concat(`index`," ", `city`) AS delivery_address from Orders;	1. ничего, запрос составлен неверно 2. покажет уникальные значения индексов и адресов из таблицы Orders 3. соединит поля с индексом и адресом из таблицы Orders и покажет их с псевдонимом delivery_address 4. соединит поля с индексом и адресом из таблицы Orders, но покажет их без псевдонима	Средний

ОПК-3.3 ОПК-7.1	8. Для чего используется LIMIT: select * from Orders limit 10;	1. необходим, чтобы показать все заказы, содержащие цифру 10 2. необходим, чтобы показать первых 10 записей в запросе 3. необходим, чтобы показать рандомные 10 записей в запрос 4. не существует такого оператора	Средний
ОПК-3.3 ОПК-7.1	9. Выберите пример правильно составленного запроса с использованием агрегирующей функции SUM:	1. select sum(price) from Orders; 2. select sum(price), customer_name from Orders; 3. select * from Orders where price=sum(); 4. select sum() from Orders group by price desc;	Средний
ОПК-3.3 ОПК-7.1	10. Выберите корректно составленный запрос с функцией GROUP BY:	1. select count(*) from Orders GROUP seller_id; 2. select seller_id, count(*) from Orders GROUP seller_id; 3. select seller_id, count(*) from Orders GROUP BY seller_id; 4. select count(*) from Orders GROUP ON seller_id;	Средний
ОПК-3.3 ОПК-7.1	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 2. количество продавцов, у которых 2, 4 или 6 товаров 3. ничего, запрос составлен неверно, HAVING указывается до группировки 4. ничего, запрос составлен неверно, для указания условия должно быть использовано WHERE	Средний
ОПК-3.3 ОПК-7.1	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) 2. select * from Orders where price = max 3. select count(*) from Orders 4. select * from Orders where price = (select max(price) from Orders)	Средний
ОПК-3.3 ОПК-7.1	13. Выберите корректный пример составленного	1. select Orders.id, Orders.customer_name,	Средний

	запроса с использованием JOIN. Данный запрос выведет нам данные ID заказа, имя заказчика и продавца:	Sellers.id from Orders LEFT JOIN ON Sellers AND Orders.seller_id = Sellers.id; 2. select id AND customer_name AND seller_id from Orders LEFT JOIN Sellers ON seller_id = id; 3. select Orders.id, Orders.customer_name, Sellers.id from Orders LEFT JOIN Sellers ON Orders.seller_id = Sellers.id; 4. select Orders.id, Orders.customer_name, Sellers.id from Orders JOIN Sellers WHEN Orders.seller_id = Sellers.id;	
ОПК-3.3 ОПК-7.1	14. Как правильно добавить строку в таблицу? Какой запрос верный?	1. INSERT INTO `SimpleTable` (`some_text`) VALUES ("my text"); 2. INSERT INTO `SimpleTable` SET `some_text`="my text"; 3. SET INTO `SimpleTable` VALUE `some_text`="my text"; 4. UPDATE INTO `SimpleTable` SET `some_text`="my text";	Средний
ОПК-3.3 ОПК-7.1	15. Какие поля из таблицы обязательно перечислять в INSERT для вставки данных?	1. Конечно все 2. Только те, у которых нет DEFAULT значения 3. Те, у которых нет DEFAULT значения и которые не имеют атрибут auto_increment 4. Все поля имеют негласное DEFAULT значения, обязательных полей в SQL нет	Средний
ОПК-3.3 ОПК-7.1	16. В каких командах можно использовать LIMIT?	1. Только Select 2. Select и Insert 3. Select, Update, Delete 4. Select, Insert, Delete, Update	Высокий
ОПК-3.3 ОПК-7.1	17. Как можно заранее узнать, какие записи будут удалены при выполнении DELETE?	1. Зачем заранее, просто вызвать его и посмотреть какие записи пропали 2. Заменить DELETE на SELECT *, ведь в остальном синтаксис	Высокий

		DELETE похож на синтаксис простого SELECT 3. Сделать DELETE с LIMIT 1, одну запись не жалко 4. SQL создан для хранения данных, их нельзя удалять	
ОПК-3.3 ОПК-7.1	18. Какой командой можно создать новую таблицу?	1. CREATE TABLE 2. MAKE TABLE 3. SET TABLE 4. Создавать таблицы можно только через интерфейс СУБД, специальной SQL команды для этого нет	Высокий
ОПК-3.3 ОПК-7.1	19. Можно ли поменять тип данных поля в уже существующей таблице?	1. Да, при помощи команды ALTER 2. Да, достаточно сделать INSERT с новым типом данных 3. Нет, только пересоздать таблицу 4. Тип бывает только у таблицы, а не у поля таблицы	Высокий
ОПК-3.3 ОПК-7.1	20. Какого из перечисленных ниже видов JOIN на самом деле не существует:	1. LEFT JOIN - который выведет все записи первой таблицы, а для ненайденных пар из правой таблицы проставит значение NULL 2. RIGHT JOIN - который выведет все записи второй таблицы, а на место недостающей информации из первой таблицы проставит NULL 3. INNER JOIN - который показывает только те записи, для которых нашлись пары 4. TRUE JOIN - который выведет все верные значения	Высокий