

Министерство науки и высшего образования Российской Федерации

федеральное государственное бюджетное образовательное учреждение
высшего образования

«Санкт-Петербургский государственный технологический институт
(технический университет)»

Кафедра систем автоматизированного проектирования и управления

ЭКЗАМЕНАЦИОННЫЕ ВОПРОСЫ И ЗАДАЧИ

по дисциплине

«Базы данных»

Части 1 – 4

Зав. кафедрой
проф. Т.Б. Чистякова

Лектор
доц. В.Ю. Плonsкий

**Санкт-Петербург
2021**

1 (Теоретический вопрос)

- 1.01)** Многоуровневая архитектура БД. Внешняя, концептуальная и внутренняя модели данных. Словарь данных.
- 1.02)** Инфологическое проектирование. Модель сущность-связь. ER-диаграмма. Модель Питера Чена.
- 1.03)** Даталогическое проектирование. Модель сущность-связь. ER-диаграмма. Модель Crow's Foot.
- 1.04)** Функциональный и предметный принцип проектирования информационной структуры. Диаграммы потоков данных.
- 1.05)** Потенциальные и внешние ключи. Виды связей, обязательность связей.
- 1.06)** Идентифицирующие и неидентифицирующие связи. Мощность связи.
- 1.07)** Дополнительные типы связей: многие-ко-многим. Реализация в РСУБД.
- 1.08)** Дополнительные типы связей: рекурсивные связи. Реализация в РСУБД.
- 1.09)** Композитный и суррогатный ключи. Примеры их использования.
- 1.10)** Расширенная модель сущность-связь. Супертипы и подтипы. Пересекающиеся и непересекающиеся подтипы.
- 1.11)** Расширенная модель сущность-связь. Супертипы и подтипы. Частичная и абсолютная полнота.
- 1.12)** Дополнительные типы связей: связи высокого порядка. Реализация в РСУБД.
- 1.13)** Реляционная модель данных. Определения домена, отношения, атрибута, кортежа. Фундаментальные свойства отношений.
- 1.14)** Нормализация данных. Функциональные зависимости. Аномалии обновления. Первая и вторая НФ.
- 1.15)** Нормализация данных. Транзитивные зависимости. Третья НФ и БКНФ.
- 1.16)** Реляционная модель данных. Потенциальные и внешние ключи.
- 1.17)** Реляционная алгебра. Операции выборки и проекция. Их соответствие SQL.
- 1.18)** Реляционная алгебра. Операции объединения и разности. Их соответствие SQL.

- 1.19)** Реляционная алгебра. Операции декартова произведения и переименования. Их соответствие SQL.
- 1.20)** Реляционная алгебра. Пересечение и естественное соединение. Их соответствие SQL.
- 1.21)** Реляционная алгебра. Операции выборки и проекция. Их соответствие SQL.
- 1.22)** Реляционная алгебра. Операции присваивания и деления. Их соответствие SQL.
- 1.23)** Реляционная алгебра. Операции тета-соединения и внешнего соединения. Их соответствие SQL.
- 1.24)** Реляционная алгебра. Операции естественного и тета-соединения. Их соответствие SQL.
- 1.25)** Реляционная алгебра. Операции внешнего соединения, их разновидности. Их соответствие SQL.
- 1.26)** Реляционная алгебра. Операции обобщенной проекции, агрегации и группировки. Их соответствие SQL.
- 1.27)** Полусоединения и антисоединения. Коррелирующие и некоррелирующие подзапросы.
- 1.28)** Дерево запроса. Оптимизатор запросов. План выполнения запроса.
- 1.29)** Декларативная поддержка целостности БД. Правило целостности сущностей. Создание / удаление первичного ключа таблицы: операторы CREATE TABLE и ALTER TABLE.
- 1.30)** Декларативная поддержка целостности БД. Правило ссылочной целостности. Стратегии поддержки ссылочной целостности. Создание / удаление внешнего ключа: Операторы CREATE TABLE и ALTER TABLE.
- 1.31)** Процедурная поддержка целостности БД. Хранимые процедуры и функции.
- 1.32)** Процедурная поддержка целостности БД. Триггеры.
- 1.33)** Определение транзакции. Свойства транзакций. Проблемы одновременного доступа. Уровни изолированности транзакций. Синтаксис работы с транзакциями.
- 1.34)** Блокирование данных. Протокол двухфазной блокировки и его разновидности. Эскалация блокировок.

2 (Практический вопрос – SQL)

Замечание: типовой пример задач на составление SQL-запроса (схема БД не будет меняться, но формулировки запросов могут отличаться на экзамене при сохранении уровня сложности).

В схему БД добавлены три таблицы:

```
-- Поставщики
create table vendor (
    v_code int primary key,
    v_name varchar (30),
    v_status tinyint not null -- статус
);
```

```
-- Продукты (товары)
create table product (
    p_code int primary key,
    p_name varchar (30),
    p_price decimal (10,2), -- отпускная цена
    p_stock decimal (10,3) -- количество на складе
);
```

```
-- Цены поставщиков
create table vendor_price (
    vp_code int auto_increment primary key,
    v_code int, -- код поставщика
    p_code int, -- код продукта (товара, материала, сырья)
    vp_data datetime, -- дата установки новой закупочной цены
    vp_price decimal (10, 2) not null, -- закупочная цена
    constraint foreign key (v_code) references vendor (v_code),
    constraint foreign key (p_code) references product (p_code)
);
```

Сначала заполняются таблицы поставщиков и продуктов:

```
insert into vendor values
    (1, 'альфа', 5),
    (2, 'гамма', 7),
    (3, 'дельта', 9),
    (4, 'сигма', 2),
    (5, 'омега', 3);
```

```

insert into product values
( 1, 'яблоко', 55, 1000),
( 2, 'груша', 77, 1500),
( 3, 'капуста', 20, 1700),
( 4, 'картофель', 35, 1400),
( 5, 'киви', 90, 800),
( 6, 'свёкла', 33, 2900),
( 7, 'морковь', 28, 2400),
( 8, 'бананы', 62, 700),
( 9, 'апельсин', 88, 900),
(10, 'мандарин', 79, 1100);

```

Таблица **vendor_price** содержит *историю* изменения цен на продукцию от разных поставщиков. Актуальной ценой продукта считается цена, ближайшая к данному моменту времени (возможно, в разрезе поставщика).

Для начального заполнения таблицы **vendor_price** можно использовать хранимую процедуру. Пример хранимой процедуры **price_init**, реализующей заполнение таблицы, показан ниже. В параметре процедуры указывается количество вставляемых записей в таблицу. Процедура формирует тестовые данные для таблицы (случайные сочетания поставщика, продукта, закупочной цены и даты):

```

create procedure price_init(in cycle int)
prog: begin
    declare a int default 0; -- счетчик
    declare max_vend int;
    declare max_prod int;
    set max_vend = (select max(v_code) from vendor);
    set max_prod = (select max(p_code) from product);
    if max_vend = 0 or max_prod = 0 then
        leave prog;
    end if;
    while a < cycle do
        insert into vendor_price (v_code, p_code, vp_price, vp_data)
        values (FLOOR ( 1 + RAND() * max_vend),
                FLOOR ( 1 + RAND() * max_prod),
                FLOOR (10 + RAND() * 100),
                DATE_SUB(now(), INTERVAL FLOOR (1+RAND()*31) DAY));
        set a = a + 1;
    end while;
end;

```

```

mysql> call price_init(20);
Query OK, 1 row affected (0.05 sec)

```

Замечание: Вы можете модифицировать текст процедуры, чтобы заполнять данные не только за последний месяц.

```
mysql> select * from vendor_price order by 2,3,4;
```

vp_code	v_code	p_code	vp_data	vp_price
16	1	4	2021-11-17 12:59:13	69.00
2	1	4	2021-12-03 12:59:13	41.00
6	1	6	2021-12-06 12:59:13	48.00
17	1	8	2021-12-15 12:59:13	35.00
9	2	1	2021-12-14 12:59:13	36.00
3	2	2	2021-12-01 12:59:13	11.00
15	2	6	2021-11-29 12:59:13	74.00
1	2	7	2021-12-01 12:59:13	57.00
13	2	8	2021-12-04 12:59:13	24.00
8	3	1	2021-11-29 12:59:13	62.00
20	3	3	2021-12-01 12:59:13	18.00
11	3	4	2021-11-21 12:59:13	26.00
5	3	4	2021-11-28 12:59:13	43.00
4	3	4	2021-12-07 12:59:13	19.00
19	3	4	2021-12-17 12:59:13	29.00
12	4	1	2021-11-24 12:59:13	29.00
18	4	1	2021-12-07 12:59:13	17.00
7	4	4	2021-11-22 12:59:13	66.00
10	5	8	2021-12-16 12:59:13	35.00
14	5	9	2021-12-05 12:59:13	67.00

Замечание: **актуальная цена** – цена, наиболее близкая к текущему моменту времени. Так, актуальная цена на продукцию №4 от поставщика №1 – 41 руб. (установлена 03.12.2021, vp_code = 2). Актуальная цена на продукцию №4 безотносительно поставщиков – 29 руб. (установлена 17.12.2021, vp_code = 19).

Подготовьте тексты запросов:

2.01) Получите количество видов товаров, которые когда-либо поставляли поставщики с нечетным статусом (1, 3, 5, ...) до определенного момента времени включительно. Момент задается параметром.

2.02) Измените статус поставщиков, придерживаясь правила: если поставщик поставляет более 3 видов товаров, то его статус должен стать равным количеству видов поставляемых товаров.

2.03) Получите названия пяти товаров, отсортировав их по разности отпускной цены и актуальной закупочной цены (5 самых «выгодных» товаров).

2.04) В результате ошибок ввода в таблице product появились дубликаты (отличаются только значением первичного ключа, остальные значения полей совпадают). Удалите дубликаты (остаются записи с наименьшим значением первичного ключа).

- 2.05)** Получите наименования поставщиков, которые поставляют все виды товаров.
- 2.06)** Получите наименования поставщиков, которые поставляют на данный момент времени все свои товары, дороже чем другие поставщики (самые «жадные» поставщики).
- 2.07)** Получите наименования товаров, которые не поставляются ни одним поставщиком.
- 2.08)** Получите наименования товаров, запасы которых менее 1000 с указанием минимальной актуальной закупочной цены (подбор наиболее «выгодных» цен).
- 2.09)** Получите наименования продуктов, информация о ценах закупки которых не обновлялась, начиная с определенной даты, задаваемой параметром («устаревшие цены»).
- 2.10)** Получите наименования поставщиков, которые поставляют хотя бы один товар дешевле, чем поставщик с определенным кодом, который задается параметром.
- 2.11)** Получите наименования поставщиков и наименования наиболее выгодных для покупки у них товаров.
- 2.12)** Получите наименование товаров, которые наиболее выгодно заказать у определенного поставщика, основываясь на актуальных ценах. Код поставщика задается параметром.
- 2.13)** Получите наименования продуктов, закупочная цена на которые выросла более, чем в 1.5 раза за определенный период времени. Период задается параметрами.
- 2.14)** Получите для каждого наименования продукта значение прибыли (разность суммы продажи по отпускной цене и актуальной цены закупки безотносительно поставщиков) при условии полной ликвидации запасов.
- 2.15)** Получите 3 самых дорогих для закупки продукта на данный момент, запасы которых между 100 и 1000 (основываясь на минимальных актуальных закупочных ценах).
- 2.16)** Обновите данные поставщиков, добавив к текущему статусу 3 позиции, если они поставляют хотя бы один продукт, дешевле чем другие поставщики, основываясь на актуальных ценах закупки.
- 2.17)** Обновите отпускные цены, установив их на уровне в 1.5 выше, чем максимальная актуальная цена их закупки по всем поставщикам.
- 2.18)** Обновите отпускные цены, уменьшив их в 2 раза для тех продуктов, запасы которых больше 1000, и есть больше трех поставщиков, которые поставляют данный продукт.
- 2.19)** Напишите хранимую, которая для поставщиков с определенным статусом удаляет все записи из таблицы `vendor_price` кроме записей с актуальными ценами. Значение статуса передается в качестве аргумента.
- 2.20)** Напишите хранимую процедуру, которая на определенный % изменяет все отпускные цены на товары, запасы которых больше определенного уровня. Значения % и уровня запасов передаются в качестве аргумента.

3 (Практический вопрос – Нормализация)

Замечание: Типовой пример исходной таблицы может отличаться на экзамене.

Исходные данные

В учебном заведении для планирования учебного процесса периодически подготавливается отчет, в котором содержится информация, представленная следующей таблицей:

Фамилия	Номер	Должность	Оклад	Дисциплина	Занятие	Группа	Семестр	Кол студ
Иванов	121212	Ст. преп.	10000	БД, ОС	Практ.	111, 333	3	20, 27
Иванов	212121	Ст. преп.	10000	Информатика	Лабор.	111	3, 4	20
Петров	131313	Доцент	20000	Информатика	Лекция, Практ.	333	1	27
Петров	131313	Доцент	20000	Робототехника	Лекция	444	5, 6	18
Федоров	313131	Проф.	40000	ОС	Лекция	111	3	20
Федоров	313131	Проф.	40000	ТАУ	Лекция, Лабор.	222	5	14
Федоров	313131	Проф.	40000	ТАУ	Курс. Проект	444	6	18
Яковлев	141414	Ст. преп.	10000	Информатика	Лабор.	222	2	14
Яковлев	141414	Ст. преп.	10000	Робототехника	Лабор.	444, 555	6	18, 21

Замечание: в исходной таблице указано, что, например:

Петров читает лекции и проводит практику для группы 333 в первом семестре, количество студентов в этой группе – 27; также Петров читает лекции по дисциплине «Робототехника» для группы 444 в пятом и шестом семестрах, количество студентов в этой группе – 18.

Иванов (с табельным номером 121212) проводит практические занятия по дисциплинам «БД» и «ОС» для групп 111 и 333 в третьем семестре; количество студентов в группах 333 и 444 – соответственно 20 и 27.

Задача

Пошагово приведите эту таблицу к схеме БД, для которой выполняются требования БКНФ (исходная таблица \Rightarrow 1НФ \Rightarrow 2НФ \Rightarrow 3НФ \Rightarrow БКНФ). Обоснуйте каждый шаг, **указав функциональные зависимости** в получаемых отношениях.

4 (Тест)

4.1) Дано отношение Book:

Id	Title	Author	Publisher	Year

Выберите выражение реляционной алгебры, позволяющее получить перечень авторов, чьи книги издавались во всех издательствах, упомянутых в этом отношении:

1. $R = (Book[Author, Publisher]) / (Book[Author])$
2. $R = (Book[Author, Publisher]) / (Book[Publisher])$
3. $R = Book - (Book[Author])$
4. $R = Book[Author]$

4.2) Дано отношение Book:

Id	Title	Author	Publisher	Year

Что получится в результате применения выражения реляционной алгебры (отношение R):

$R = (Book[Publisher] \bowtie 'Азбука' \text{ AND } Year \bowtie 2000) [Title]$

1. Перечень названий книг, за исключением изданных в 2000 году или в издательстве «Азбука».
2. Полная информация о всех книгах, за исключением изданных в 2000 году или в издательстве «Азбука».
3. Перечень названий книг, за исключением изданных в 2000 году в издательстве «Азбука».
4. Полная информация о всех книгах, за исключением изданных в 2000 году в издательстве «Азбука».

4.3) Дано отношение Book:

Id	Title	Author	Publisher	Year
1	Война и мир	Толстой Л.Н.	Мысль	1981
2	Война и мир	Толстой Л.Н.	Мысль	1981

Выполняются запросы:

- (1) `SELECT * FROM Book;`
- (2) `SELECT DISTINCT * FROM Book;`
- (3) `SELECT Title, Author, Publisher, Year FROM Book;`
- (4) `SELECT DISTINCT Title, Author, Publisher, Year FROM Book;`

Выберите правильное утверждение:

1. Все запросы вернут одинаковое число записей.
2. Запросы (1) и (3) вернут равное число записей, запросы (2) и (4) вернут также равное число записей, но меньшее чем у первой пары.
3. Запросы (1) и (2) вернут равное число записей, запросы (3) и (4) вернут также равное число записей, но меньшее чем у первой пары.
4. Запросы (1), (2), (3) вернут равное число записей, запрос (4) вернет меньшее число записей.

4.4) Дано отношение Book:

Id	Title	Author	Publisher	Year

Выберите запросы, возвращающие информацию о книгах, изданных в издательстве «Азбука» или «Политехника».

1. `SELECT * FROM Book WHERE Publisher = 'Политехника', Publisher = 'Азбука';`
2. `SELECT * FROM Book WHERE Publisher IN ('Политехника', 'Азбука');`
3. `SELECT * FROM Book WHERE Publisher = 'Политехника' OR Publisher = 'Азбука';`
4. `SELECT * FROM Book WHERE Publisher = 'Политехника'`
`UNION`
`SELECT * FROM Book WHERE Publisher = 'Азбука';`

4.5) Дано отношение Book:

Id	Title	Author	Publisher	Year

Выполняется запрос:

`SELECT * FROM Book WHERE Publisher = 'Политехника'`

`UNION`

`SELECT Title, Author, Publisher FROM Book;`

Выберите правильное утверждение о результате запроса:

1. Синтаксическая ошибка.
2. Результат, аналогичный запросу `SELECT * FROM Book;`
3. Результат, аналогичный запросу `SELECT * FROM Book where publisher = 'Азбука';`
4. Результат, аналогичный запросу `SELECT Title, Author, Publisher FROM Book;`

4.6) Дано отношение Book:

Id	Title	Author	Publisher	Year

Выберите выражение реляционной алгебры, позволяющее найти всех авторов, ни одной книги которых не было выпущено в издательстве «Азбука»:

1. (Book[Publisher <> 'Азбука'])[Author]
2. (Book[Author]) - ((Book[Publisher = 'Азбука'])[Author])
3. Book[Publisher <> 'Азбука']
4. Ни один из предложенных вариантов

4.7) Даны отношения Student и Result:

Id	Name
123	Иванов
124	Петров

Id	Subject	Mark
123	Математика	5
124	Математика	3
123	БД	5

Выберите все правильные варианты, которые позволяют отобразить ФИО студентов, сдавших хотя бы один экзамен на оценку 5:

1. SELECT DISTINCT Name FROM Student, Result WHERE Student.Id = Result.Id AND Mark=5;
2. SELECT DISTINCT Name FROM Student LEFT JOIN Result ON Student.Id = Result.Id AND Mark=5;
3. SELECT DISTINCT Name FROM Student INNER JOIN Result ON Student.Id = Result.Id AND Mark=5;
4. SELECT DISTINCT Name FROM Student WHERE Id NOT IN (SELECT Id FROM Result WHERE Mark <> 5);

4.8) Даны отношения:

Даны отношения Student и Result:

Id	Name
123	Иванов
124	Петров

Id	Subject	Mark
123	Математика	5
124	Математика	3
123	БД	5

Выберите все запросы, которые для приведенных данных вернут пустое множество:

1. SELECT * FROM Student;
2. SELECT * FROM Student LEFT JOIN Result ON 1<>1;
3. SELECT COUNT(*) FROM Student GROUP BY Name;
4. SELECT * FROM Student INNER JOIN Result ON 1<>1;
5. SELECT COUNT(*) FROM Student GROUP BY Name HAVING COUNT(*) > 1;

4.9) Дано отношение Book:

Id	Title	Author	Publisher	Year

Выбрать запрос, возвращающий сведения о тех авторах, чьи книги переиздавались:

1. SELECT DISTINCT Author FROM Book GROUP BY Author, Title HAVING COUNT(Id)>1;
2. SELECT DISTINCT Author FROM Book WHERE COUNT(Id)>1;
3. SELECT DISTINCT Author FROM Book GROUP BY Author HAVING COUNT(Id)>1;

4.10) Даны отношения Book и Book_in_Lib. Подготовлены запросы (1) и (2), которые должны выбрать книги, имеющиеся в Book, но отсутствующие в Book_in_Lib:

Id	Title	Author	Publisher	Year
1	Химия	Ахметов	Политехника	1996
2	Базы данных	Гончаров	BHV	2000
...

LibId	Id	Status
10	1	хранится
11	1	выдана
...

- (1) SELECT * FROM Book WHERE Id NOT IN (SELECT Id FROM Book_in_Lib);
- (2) SELECT * FROM Book WHERE NOT EXISTS (SELECT * FROM Book_in_Lib WHERE Book.Id = Book_in_Lib.Id);

Выберите правильно утверждение:

1. Запрос (1) даст неправильный результат.
2. Запрос (2) даст неправильный результат.
3. Оба запроса дадут правильный результат, но производительность запроса (2) выше.
4. Оба запроса дадут правильный результат, но производительность запроса (1) выше.

4.11) Даны отношения Student и Result:

Id	Name
123	Иванов
124	Петров

Id	Subject	Mark
123	Математика	5
124	Математика	3
123	БД	5

Выберите все запросы, которые приведут к ошибкам:

1. SELECT S.Id FROM Student S;
2. SELECT * FROM Student WHERE Mark > 5;
3. SELECT * FROM Student WHERE NOT EXISTS (SELECT * FROM Result WHERE Id =2);
4. SELECT * FROM Student GROUP BY Name;

4.12) Дано отношение:

Id	Title	Author	Publisher	Year

Необходимо увеличить на 1 год издания всех книг, выпущенных в издательстве «Политехника». Выберите правильные варианты запроса для решения этой задачи:

1. UPDATE Book SET Year++ WHERE Publisher = 'Политехника';
2. UPDATE Book SET Year=Year + 1 WHERE Publisher = 'Политехника';
3. UPDATE Year FROM Book SET Year=Year + 1 WHERE Publisher = 'Политехника';
4. UPDATE Book SET Year=Year++ WHERE Publisher IN ('Политехника');
5. UPDATE Book SET Year=Year + 1 WHERE Publisher NOT IN ('Политехника');

4.13) Дано отношение Book:

Id	Title	Author	Publisher	Year
1	Война и мир	Толстой Л.Н.	Мысль	1981
2	Война и мир	Толстой Л.Н.	Мысль	1981

Выберите запрос, который вернёт перечень авторов и названий книг без повторений:

1. SELECT * FROM Author, Title;
2. SELECT DISTINCT * FROM Book;

3. SELECT DISTINCT * FROM Author, Title;
4. SELECT DISTINCT Author, Title FROM Book;
5. SELECT Author, Title FROM Book;

4.14) Дано отношение:

Id	Title	Author	Publisher	Year

Была создана таблица Book_Old с аналогичной структурой. Необходимо поместить в неё информацию о всех книгах, которые были изданы до 2000 года. Выберите правильный вариант запроса:

1. INSERT INTO Book_Old SELECT * FROM Book WHERE Year < 2000;
2. INSERT INTO Book_Old VALUES (SELECT * FROM Book WHERE Year < 2000);
3. SELECT * FROM Book WHERE Year < 2000 TO Book_Old;
4. COPY * FROM Book WHERE Year < 2000 TO Book_Old;
5. SELECT * FROM Book WHERE Year LESS THEN 2000 TO Book_Old;

4.15) Даны отношения Book и Result:

Id	Title	Author	Publisher	Year
1	Война и мир	Толстой Л.Н.	Мысль	1981
2	Война и мир	Толстой Л.Н.	Мысль	1981

Id	Subject	Mark
123	Математика	5
124	Математика	3
123	БД	5

Отношение R равно декартову произведению Book и Result. Выберите правильный вариант, описывающий степень и кардинальное число отношения R:

1. Степень R = 3; кардинальное число R = 2;
2. Степень R = 3; кардинальное число R = не определено;
3. Степень R = 8; кардинальное число R = не определено;
4. Степень R = 8; кардинальное число R = 6;
5. Степень R = 2; кардинальное число R = 6;
6. Степень R = 2; кардинальное число R = 5;