

Вариант 1.

**Разработать БД Турист с тремя таблицами, установить связь между таблицами.
Реализовать SQL-запросы на выборку данных из БД**

Таблица "Туристы"

id (INT, PK) - уникальный идентификатор туриста
имя (VARCHAR)
фамилия (VARCHAR)
пол (VARCHAR)
дата_рождения (DATE)
номер_телефона (VARCHAR)
электронная_почта (VARCHAR)
Таблица "Туры"

id (INT, PK) - уникальный идентификатор тура
название (VARCHAR)
страна (VARCHAR)
город (VARCHAR)
дата_начала (DATE)
дата_окончания (DATE)
цена (DECIMAL)
Таблица "Бронирования"

id (INT, PK) - уникальный идентификатор бронирования
id_туриста (INT, FK) - идентификатор туриста, который сделал бронирование
id_тура (INT, FK) - идентификатор тура, на который было сделано бронирование
дата_бронирования (DATE)
кол-во_туристов (INT)

Для установки связей между таблицами использовать внешние ключи (FK) на таблицах "Бронирования" и "Туры", которые ссылаются на соответствующие поля таблицы "Туристы" и "Туры".

SQL-запросы на выборку данных из этой БД:

1. Вывести список всех туристов
2. Вывести список всех туров, отсортированных по цене в порядке убывания
3. Вывести список всех бронирований, сделанных в заданном городе
4. Вывести список всех туристов, сделавших бронирование в определенный период времени
5. Вывести список всех туров с указанием названия страны и города
6. Вывести список всех туристов, женщин, у которых дата рождения позже 01.01.1990
7. Вывести список всех туров, цена которых больше 5000
8. Вывести список всех туристов, которые сделали бронирование на конкретный тур
9. Вывести список всех туристов, которые сделали бронирование на тур в указанную дату
10. Вывести список всех туристов, у которых номер телефона начинается на "+7"

Вариант 3.

Разработать БД ЗАПЛАТА, содержащую две таблицы *Анкета* и *Больничные листы*. Установить связь между таблицами. Реализовать SQL-запросы на выборку данных из БД.

Таблица "Анкета"

id (INT, PK) - уникальный идентификатор сотрудника

имя (VARCHAR)

фамилия (VARCHAR)

дата_рождения (DATE)

пол (VARCHAR)

дата_найма (DATE)

должность (VARCHAR)

отдел (VARCHAR)

базовая_ставка (DECIMAL)

Таблица "Больничные листы"

id (INT, PK) - уникальный идентификатор больничного листа

id_сотрудника (INT, FK) - идентификатор сотрудника, на которого выписан больничный лист

дата_начала (DATE)

дата_окончания (DATE)

причина (VARCHAR)

диагноз (VARCHAR)

оплачен (BOOLEAN)

В данной структуре таблица "Больничные листы" связана с таблицей "Анкета" через внешний ключ id_сотрудника. Это означает, что каждый больничный лист относится к определенному сотруднику из таблицы "Анкета".

1. Вывести список всех сотрудников и их должностей
2. Вывести список всех сотрудников и их базовых ставок
3. Вывести список всех сотрудников, работающих в отделе "IT"
4. Вывести список всех сотрудников, принятых на работу после 1 января 2022 года
5. Вывести список всех больничных листов, выписанных сотруднику с id = 42
6. Вывести список всех больничных листов, оплаченных компанией
7. Вывести список всех сотрудников, имеющих больничные листы на текущий месяц
8. Вывести среднюю базовую ставку всех сотрудников
9. Вывести список всех сотрудников, имеющих базовую ставку выше 100 000
10. Вывести список всех сотрудников и общее количество дней, проведенных ими на больничном
11. Вывести информацию о сотрудниках и их больничных листах за последний месяц
12. Вывести среднюю продолжительность больничных листов сотрудников в каждом отделе
13. Вывести список сотрудников и информацию о последнем больничном листе, который они оформляли
14. Вывести список сотрудников и информацию о первом больничном листе, который они оформляли
15. Вывести список сотрудников и суммарную продолжительность их больничных листов в текущем году

Вариант 2.

Разработать БД ОПТОВАЯ БАЗА. Структура БД состоит из пяти таблиц: Товары, Магазины, Заявки магазинов, Количество товаров на складе, Состав. Реализовать SQL-запросы на выборку данных из БД.

Таблица "Товары"

id (INT, PK) - уникальный идентификатор товара
название (VARCHAR)
описание (VARCHAR)
единица измерения (VARCHAR)

Таблица "Магазины"

id (INT, PK) - уникальный идентификатор магазина
название (VARCHAR)
адрес (VARCHAR)
телефон (VARCHAR)

Таблица "Заявки магазинов"

id (INT, PK) - уникальный идентификатор заявки
id_магазина (INT, FK) - идентификатор магазина, который подал заявку
дата_заявки (DATE)

Таблица "Количество товаров на складе"

id (INT, PK) - уникальный идентификатор записи
id_товара (INT, FK) - идентификатор товара
количество (INT)

Таблица "Состав"

id (INT, PK) - уникальный идентификатор записи
id_заявки (INT, FK) - идентификатор заявки
id_товара (INT, FK) - идентификатор товара
количество (INT)

SQL-запросы на выборку данных:

1. Вывести список всех товаров и их описания
2. Вывести список всех магазинов и их адресов
3. Вывести список всех заявок магазинов и даты, на которые они были поданы
4. Вывести список товаров и количество их наличия на складе
5. Вывести список товаров и количество их наличия на складе в порядке убывания количества
6. Вывести список всех заявок магазинов и товаров, которые были в них заказаны
7. Вывести список всех товаров, у которых на складе количество меньше минимально допустимого
8. Вывести список всех заявок магазинов, которые были сделаны в определенный период времени
9. Вывести список всех магазинов, у которых суммарное количество товаров на складе меньше заданного значения

Вариант 4 (можно 4 человека)

Разработать БД ДЕКАНАТ содержащую следующие таблицы. Установить связь между таблицами. Реализовать SQL-запросы на выборку данных из БД.

Структура БД "Деканат":

Таблица "Факультеты":

id (INT, PK) - уникальный идентификатор факультета
название (VARCHAR)

Таблица "Кафедры":

id (INT, PK) - уникальный идентификатор кафедры
название (VARCHAR)
id_факультета (INT, FK) - идентификатор факультета, к которому относится кафедра

Таблица "Специальности":

id (INT, PK) - уникальный идентификатор специальности
название (VARCHAR)
id_кафедры (INT, FK) - идентификатор кафедры, которая готовит по данной специальности

Таблица "Предметы":

id (INT, PK) - уникальный идентификатор предмета
название (VARCHAR)

Таблица "Форма сдачи предмета":

id (INT, PK) - уникальный идентификатор формы сдачи предмета
название (VARCHAR)

Таблица "Учебный план":

id (INT, PK) - уникальный идентификатор учебного плана
id_специальности (INT, FK) - идентификатор специальности, для которой составлен учебный план
id_предмета (INT, FK) - идентификатор предмета, входящего в учебный план
id_формы_сдачи_предмета (INT, FK) - идентификатор формы сдачи предмета
кол_во_лекционных_часов (INT)
кол_во_практических_часов (INT)
кол_во_лабораторных_часов (INT)
курсовая_работа(bool)

Таблица "Абитуриенты":

| Поле | Тип данных | Описание |
|------------------|-------------|---|
| ID | Целое число | Уникальный идентификатор абитуриента |
| Фамилия | Текст | Фамилия абитуриента |
| Имя | Текст | Имя абитуриента |
| Отчество | Текст | Отчество абитуриента |
| Пол | Текст | Пол абитуриента |
| Дата рождения | Дата/время | Дата рождения абитуриента |
| Адрес | Текст | Адрес абитуриента |
| Телефон | Текст | Телефон абитуриента |
| Email | Текст | Email абитуриента |
| Дата поступления | Дата/время | Дата поступления абитуриента |
| Специальность | Текст | Специальность, на которую зачислен абитуриент |

SQL-запросы на выборку данных из этой БД:

1. Вывести список всех студентов, зачисленных на факультет, с указанием номера группы.
2. Вывести список всех специальностей факультета и количество студентов, обучающихся по каждой из них.
3. Вывести список всех кафедр факультета и количество студентов, обучающихся на каждой кафедре.
4. Вывести список всех предметов и количество часов, выделенных на каждый предмет в учебном плане каждой специальности.
5. Вывести список всех студентов, у которых есть неудовлетворительные оценки (меньше 4) по любому предмету.
6. Вывести список всех предметов, которые изучают студенты первого курса.
7. Вывести список всех студентов, которые сдают курсовую работу в этом семестре.
8. Вывести список всех абитуриентов, зачисленных на специальность "Информатика и вычислительная техника".
9. Вывести список всех предметов, которые изучают студенты группы 101.
10. Вывести список студентов и их оценки за все предметы на специальности "Программная инженерия"

Вариант 5

Разработать БД Библиотека содержащую следующие таблицы. Установить связь между таблицами. Реализовать SQL-запросы на выборку данных из БД.

Таблица Авторы:

КодАвтора (ключевое поле)

Фамилия

Имя

Таблица Книги:

КодКниги (ключевое поле)

Название

Раздел (поле подстановки из таблицы Разделы)

Издательство (поле подстановки из таблицы Издательства)

ГодИздания

МестоХранения

Таблица Разделы:

Раздел (ключевое поле)

Таблица Издательства:

Издательство (уникальный ключ)

Город

Таблица АвторКниги:

КодАвтораКниги (ключевое поле)

КодКниги

КодАвтора

SQL-запросы на выборку данных из этой БД:

1. Получить список всех книг, отсортированных по году издания.
2. Получить список всех книг заданного автора.
3. Получить список всех книг из заданного раздела.
4. Получить список всех книг, изданных заданным издательством.
5. Получить список всех авторов в алфавитном порядке.
6. Получить список всех книг, отсортированных по названию и году издания.
7. Получить список всех книг заданного автора, отсортированных по году издания.
8. Получить список всех книг, опубликованных в заданном году.
9. Получить список всех авторов, написавших книги для заданного издательства.
10. Получить список всех книг, в названии которых есть заданное слово.

Вариант 6

Разработать БД Аптека содержащую следующие таблицы. Установить связь между таблицами. Реализовать SQL-запросы на выборку данных из БД.

Структура таблицы "Лекарственные средства":

Код препарата (уникальный идентификатор препарата)
Название препарата
Применение
Страна-Производитель
Цена

Структура таблицы "Наличие на складе":

Код препарата (ссылка на таблицу "Лекарственные средства")
Количество
Дата использования

Структура таблицы "Аптечный пункт":

Номер пункта (уникальный идентификатор аптечного пункта)
Адрес
Название препарата (ссылка на таблицу "Лекарственные средства")
Заявки
Дата Заявки
Сумма заказа
Примеры запросов к БД АПТЕКА:

SQL-запросы на выборку данных из этой БД:

1. Вывести список всех препаратов с указанием количества их наличия на складе.
2. Вывести список всех препаратов, имеющих на складе в количестве менее 10 штук.
3. Вывести список всех препаратов, которые производятся в России.
4. Вывести список всех аптечных пунктов с указанием адреса и количества наличия препаратов в каждом пункте.
5. Вывести список всех лекарственных препаратов, цена которых меньше 100 грн, отсортированных по названию.
6. Вывести список лекарственных препаратов, которые заканчиваются на складе в определенном аптечном пункте.
7. Вывести список аптечных пунктов, в которых есть хотя бы одно лекарственное средство срок годности которого истекает в этом месяце.