

Вариант 1.

Создать таблицы.

Корабли:

- Номер
- Название
- Год постройки
- Водоизмещение

Порт:

- Позиция
- Название
- Регион

Позиция – уникальное трехзначное число в интервале от 111 до 999

Самостоятельно определить первичные ключи и столбцы для обеспечения связи. Правило связи: каждому порту может быть приписано несколько кораблей, каждый корабль может быть приписан только в одном порту.

Задачи.

1. Создать процедуру приписки корабля. Входные параметры – название корабля, название порта, год постройки и водоизмещение корабля. Выходной параметр – суммарное количество кораблей в порту, с учетом приписки нового корабля.
2. С помощью пользовательской функции определить расстояние между двумя портами. Входные параметры – название 1 порта, название 2 порта. Выходной параметр – расстояние в единицах.
предполагается, что позиция порта – координаты, первая цифра – номер порта, вторая цифра – координата X, третья цифра – координата Y.
3. С помощью триггера предотвратить переполнение порта по водоизмещению. Предполагается, что суммарное водоизмещение всех судов, находящихся в порту не может превышать 100 000 тонн.

Вариант 2.

Создать таблицы.

Регион:

- Номер
- Название
- Площадь

Город:

- Позиция
- Название
- Население

Позиция – уникальное трехзначное число в интервале от 111 до 999

Самостоятельно определить первичные ключи и столбцы для обеспечения связи.

Задачи.

1. Создать процедуру переподчинения города другому административному региону. Входные параметры – название города, название региона. Выходной параметр – суммарное количество населения после включения города в регион. Предусмотреть блокировку от повторного переподчинения.
2. С помощью пользовательской функции определить регион с наибольшей плотностью населения. Входные параметры – отсутствуют. Выходной параметр – название региона с наибольшей плотностью населения.
3. С помощью триггера предотвратить критическое снижение численности населения региона. Для нормальной жизнедеятельности региона необходимо не менее 40 000 чел.

Вариант 3.

Создать таблицы.

Группа:

- Номер
- Шифр группы
- Специализация

Студенты:

- Код студента
- Фамилия
- Имя
- Адрес проживания

Аттестация:

- Дата аттестации
- Оценка

Самостоятельно определить первичные ключи и столбцы для обеспечения связи. Правило связей: в одной группе может обучаться несколько студентов, каждый студент обучается только в одной группе. Каждый студент за одну аттестацию может получить несколько оценок. В одной аттестации может участвовать несколько студентов.

- Формат поля «Адрес проживания»: Город, Улица, Дом, номер квартиры
- Возможные значения поля «Специализация»: «ИН», «ПИ», «ЭИ».

Задачи.

1. Создать процедуру зачисления студента в группу. Входные параметры – фамилия, имя студента, специализация. Выходной параметр – номер группы, в которую зачислен студент. Зачисление производить в группу указанной специализации с наименьшей численностью студентов.
2. С помощью пользовательской функции определить средний бал студенческой группы, в которой наибольшее количество успевающих студентов. Входные параметры – отсутствуют. Выходной параметр – средний бал группы.
3. С помощью триггера заменить операции по удалению записи о студентах на операции архивирования данных об этих студентах. В архивных записях следует хранить: Код студента, Фамилия и Имя, Адрес проживания, Специализация, Средний балл успеваемости

Вариант 4.

Создать таблицы.

Склад:

- Номер
- Категория склада
- Емкость склада

Транспорт:

- Код транспорта
- Тип транспорта
- Грузоподъемность

Товар:

- Код товара
- Наименование
- Объем
- Вес

Самостоятельно определить столбцы для обеспечения связи. Правило связей: каждый склад обслуживается несколькими транспортными единицами, транспорт обслуживает только один склад. Каждый транспорт одновременно может перевозить несколько товаров.

Задачи.

1. Создать процедуру заполнения склада. Входные параметры – номер склада, перечень товаров, включающий до 10 наименований. Выходной параметр – номера транспортных единиц, на которых будут доставляться указанные товары на склад.
2. С помощью пользовательской функции определить оставшееся место на складе. Входные параметры – номер склада. Выходной параметр – свободное место на складе (предполагается, что все товары, доставляемые транспортом, остаются на складе).
3. С помощью триггера предотвратить перегрузку транспорта. В случае превышения допустимого веса перевозимого груза, подобрать другой свободный транспорт, обслуживающий тот же склад. В случае, если это невозможно, предотвратить операцию и вывести сообщение.

Вариант 5.

Создать таблицы.

Факультет:

- Код факультета
- Факультет

Деканат:

- Код деканата
- Аудитория
- Номер телефона

Сотрудники:

- Код сотрудника
- Фамилия Имя Отчество
- Должность

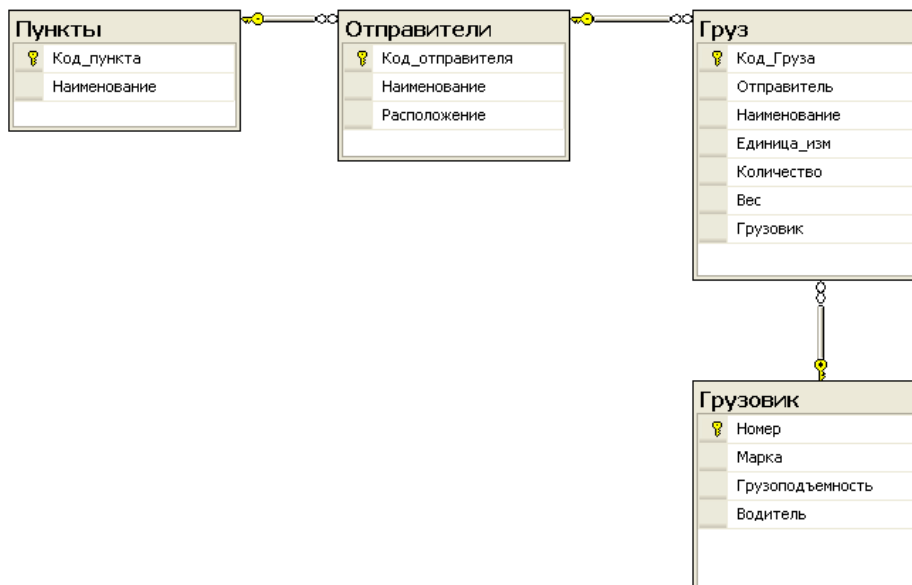
Самостоятельно определить столбцы для обеспечения связи. Правило связей: каждый факультет имеет только один деканат. В каждом деканате работает несколько сотрудников, совмещения должностей на разных факультетах запрещены. Один из сотрудников является деканом факультета, все сотрудники подчиняются декану.

- Значение поля «Аудитория» - трехзначное число, где первая цифра – номер этажа

Задачи.

1. Создать процедуру для реорганизации деканата. Перевести методистов (по одному человеку) из деканатов с самой большой численностью сотрудников в следующие деканаты (приращение по коду деканата). Входные параметры – нет. Выходной параметр – фамилии методистов, номер деканата в который они переведены.
2. С помощью пользовательской функции определить количество методистов факультета. Методист назначается на каждые 5 сотрудников деканата. Входные параметры – код факультета. Выходной параметр – количество методистов.
3. С помощью триггера предотвратить превышение численности сотрудников деканата (максимальная численность сотрудников деканата – 15 чел.). В случае превышения – сотрудника перевести в другой деканат. Если это невозможно (все деканаты укомплектованы полными штатами) – занести сведения о кандидате в резервную таблицу.

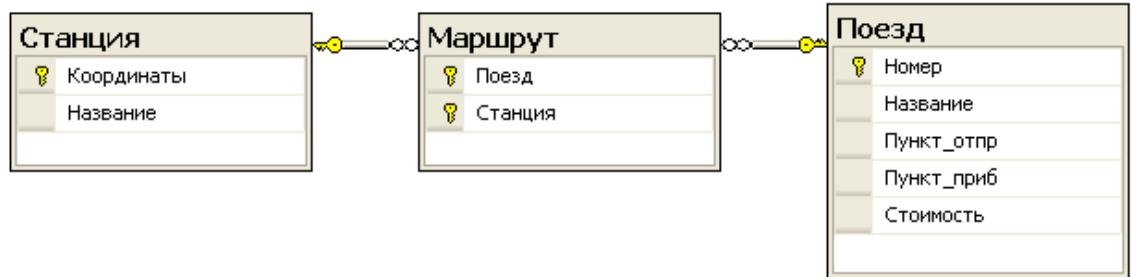
Вариант 6.



1. Создать базу данных «Грузоперевозки». Схема БД и состав полей таблиц приведен на рисунке.
2. Создать программные объекты для решения следующих задач:
 - a. Сформировать грузы отправителя и загрузить грузовик..
 - b. Определить суммарное количество грузов, находящихся в указанном пункте отправления.
 - c. Определить объем грузоперевозок указанной компании.
 - d. Выдать сведения о грузопотоках (наименование пунктов, суммарный вес доставляемых грузов, количество транспортных единиц) по указанным отправителям.
3. Создать триггер для предотвращения перегрузки транспорта. В случае превышения допустимого веса груза перегрузить его на другой грузовик той же марки.

Варианты задач к контрольной работе
«Программные объекты БД»

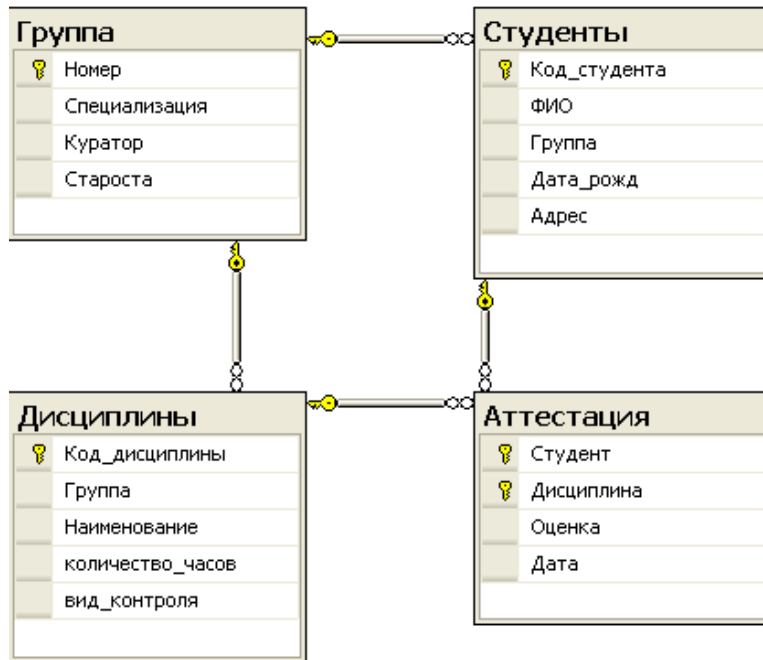
Вариант 7.



1. Создать базу данных «Движение поездов». Схема БД и состав полей таблиц приведен на рисунке.
2. Создать программные объекты для решения следующих задач:
 - а. Выдать список поездов, в которых стоимость проезда одной станции соответствует указанному пользователем критерию.
 - б. Рассчитать стоимость проезда между указанными станциями.
 - в. Выдать маршрут движения указанного поезда.
 - г. Рассчитать оптимальный по стоимости маршрут между указанными станциями.
3. Создать триггер для ограничения предельной стоимости поездки на маршруте.

Варианты задач к контрольной работе
«Программные объекты БД»

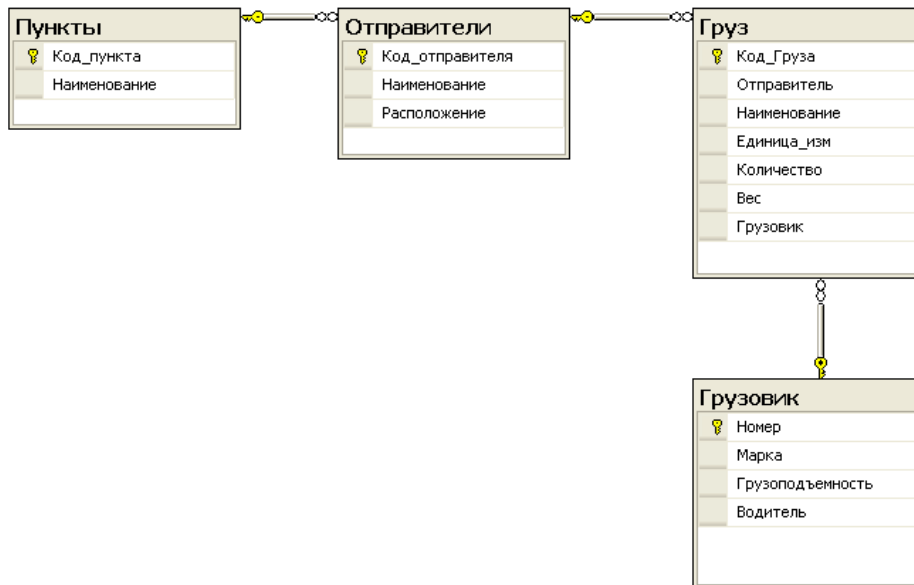
Вариант 8.



1. Создать базу данных «Учет успеваемости». Схема БД и состав полей таблиц приведен на рисунке.
2. Создать программные объекты для решения следующих задач:
 - а. Определить средний балл успеваемости по указанной дисциплине.
 - б. Рассчитать средний балл успеваемости по указанной группе
 - с. Определить лучших по успеваемости студентов по указанным дисциплинам.
 - д. Выдать отчет по успеваемости (количество оценок, средний балл успеваемости, количество задолженностей) в разрезе специальностей, дисциплин, студенческих групп.
3. Создать триггер для предотвращения удаления сведений о студентах, не завершивших сессию. (сессия продолжается с 5 по 20 января и с 5 по 20 июня)

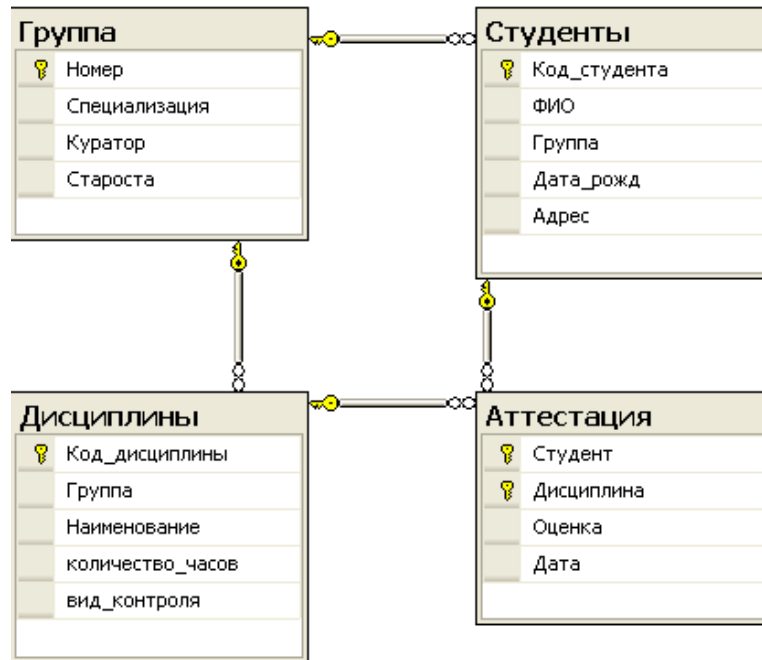
Варианты задач к контрольной работе
«Программные объекты БД»

Вариант 9.



1. Создать базу данных «Грузоперевозки». Схема БД и состав полей таблиц приведен на рисунке.
2. Создать программные объекты для решения следующих задач:
 - a. Выдать перечень грузов, определенных категорий (категория груза обозначается первой цифрой кода).
 - b. Рассчитать стоимость перевозки грузов указанного отправителя (тариф на перевозку единицы груза указывается в качестве параметра).
 - c. Обеспечить оптимальную загрузку указанного грузовика.
 - d. Отчет по грузоперевозкам в разрезе отправителей, пунктов отправления (суммарный объем грузов, количество транспортных единиц).
3. Создать триггер на ограничение количества грузов в пункте отправления.

Вариант 10.



1. Создать базу данных «Учет успеваемости». Схема БД и состав полей таблиц приведен на рисунке.
2. Создать программные объекты для решения следующих задач:
 - a. Выдать список старост групп с лучшей успеваемостью по указанной дисциплине.
 - b. Рассчитать средний балл успеваемости, отклонение от среднего по каждой дисциплине за указанный период.
 - c. Определить перечень дисциплин, разность среднего балла успеваемости по которым максимальна.
 - d. Рассчитать отставание каждого студента от успеваемости в своей группе.
3. Создать триггер для предупреждения о предельном количестве неуспевающих студентов по дисциплине.