

Задача 1

Возьмите библиотечную систему, схему которой сделали на предыдущем задании

Reader(ID, LastName, FirstName, Address, BirthDate) Book (ISBN, Title, Author, PagesNum, PubYear, PubName) Publisher (PubName, PubAdress) Category (CategoryName, ParentCat) Copy (ISBN, CopyNumber, ShelfPosition)

Borrowing (ReaderNr, ISBN, CopyNumber, ReturnDate) BookCat (ISBN, CategoryName)

Напишите SQL-запросы (или выражения реляционной алгебры) для следующих вопросов: а) Какие фамилии читателей в Москве? б) Какие книги (author, title) брал Иван Иванов? в) Какие книги (ISBN) из категории "Горы" не относятся к категории "Путешествия"? Подкатегории не обязательно принимать во внимание! г) Какие читатели (LastName, FirstName) вернули копию книгу? д) Какие читатели (LastName, FirstName) брали хотя бы одну книгу (не копию), которую брал также Иван Иванов (не включайте Ивана Иванова в результат)?

Экземпляр: {[[Номер копии](#), [ISBN](#), Положение_на_полке]}

Запись_о_взятой_книге: {[Дата_возвращения, ISBN_взятой_книги, Номер_взятой_копии, Номер_читателя]}

Книга: {[[ISBN](#), Год, Название, Автор, Количество страниц, Имя_издательства]}

Категория: {[[Имя](#), Имя_старшей_категории]}

Издательство: {[[Имя](#), Адрес]}

Читатель: {[[Номер](#), Фамилия, Имя, Адрес, День_рождения]}

Категория_книги: {[[ISBN](#), [Имя категории](#)]}

а) Какие фамилии читателей в Москве?

SELECT Фамилия

FROM Читатель

WHERE 'Москва' IN Адрес

б) Какие книги (author, title) брал Иван Иванов?

SELECT Книга.Автор, Книга.Название

```
FROM Книга, Запись_о_взятой_книге, Читатель
WHERE Запись_о_взятой_книге.ISBN_взятой_книги = Книга.ISBN AND
      Читатель.Имя = 'Иван' AND Читатель.Фамилия = 'Иванов' AND
      Запись_о_взятой_книге.Номер_читателя = Читатель.Номер
```

в) Какие книги (ISBN) из категории "Горы" не относятся к категории "Путешествия"? Подкатегории не обязательно принимать во внимание!

```
SELECT ISBN
FROM Категория_Книги
WHERE Имя_категории = 'Горы'
EXCEPT
SELECT ISBN
FROM Категория_Книги
WHERE Имя_категории = 'Путешествия'
```

г) Какие читатели (LastName, FirstName) вернули копию книгу?

```
SELECT DISTINCT Читатель.Имя, Читатель.Фамилия
FROM Читатель, Запись_о_взятой_книге
WHERE Читатель.Номер = Запись_о_взятой_книге.Номер_читателя AND
      Запись_о_взятой_книге.Дата_возвращения IS NOT NULL
```

д) Какие читатели (LastName, FirstName) брали хотя бы одну книгу (не копию), которую брал также Иван Иванов (не включайте Ивана Иванова в результат)?

```
SELECT DISTINCT Читатель.Имя, Читатель.Фамилия
FROM Читатель, Запись_о_взятой_книге
WHERE Читатель.Номер = Запись_о_взятой_книге.Номер_читателя AND
      Запись_о_взятой_книге.ISBN_взятой_книги IN (
      SELECT Запись_о_взятой_книге.ISBN_взятой_книги
      FROM Запись_о_взятой_книге, Читатель
      WHERE Читатель.Номер = Запись_о_взятой_книге.Номер_читателя AND
```

Читатель.Имя = 'Иван' AND Читатель.Фамилия = 'Иванов')
 AND NOT (Читатель.Имя = 'Иван' AND Читатель.Фамилия = 'Иванов')

Задача 2

Возьмите схему для Поездов, которую выполняли в предыдущем задании.

City (Name, Region) Station (Name, #Tracks, CityName, Region) Train (TrainNr, Length, StartStationName, EndStationName) (FromStation, ToStation, TrainNr, Departure, Arrival)

Предположим, что отношение "Connection" уже содержит транзитивное замыкание. Когда поезд 101 отправляется из Москвы в Санкт-Петербург через Тверь, содержит кортежи для связи Москва->Тверь, Тверь-Санкт-Петербург и Москва->Санкт-Петербург. Сформулируйте следующие запросы в реляционной алгебре:

а) Найдите все прямые рейсы из Москвы в Тверь. б) Найдите все многосегментные маршруты, имеющие точно однодневный трансфер из Москвы в Санкт-Петербург (первое отправление и прибытие в конечную точку должны быть в одну и ту же дату). Вы можете применить функцию DAY () к атрибутам Departure и Arrival, чтобы определить дату. в) Что изменится в выражениях для а) и б), если отношение "Connection" не содержит дополнительных кортежей для транзитивного замыкания, поэтому многосегментный маршрут Москва-> Тверь-> Санкт-Петербург содержит только кортежи Москва-> Тверь и Тверь-Санкт-Петербург?

Station: {[Name, #Tracks, CityName, CityRegion]}

City: {[Name, Region]}

Train: {[TrainNr, Length, Start_Station, End_Station]}

Connection: {[Station1Name, Station2Name, Departure, Arrival, TrainID]}

а) Найдите все прямые рейсы из Москвы в Тверь.

$$\begin{aligned} & \Pi_{TrainID} (\sigma_{CityName1=Москва \wedge CityName2=Тверь} \\ & ((\Pi_{TrainID, CityName1, CityName \text{ as } CityName2} (\sigma_{Connection.Station2Name=Station.Name} \\ & (\Pi_{TrainID, Station1Name, Station2Name, CityName \text{ as } CityName1} \\ & (\sigma_{Connection.Station1Name=Station.Name} (Connection \times Station))) \times Station)))) \end{aligned}$$

б) Найдите все многосегментные маршруты, имеющие точно однодневный трансфер из Москвы в Санкт-Петербург (первое отправление и прибытие в конечную точку должны быть в одну и ту же дату). Вы можете применить функцию DAY () к атрибутам Departure и Arrival, чтобы определить дату.

Найдем Con1 и Con2 которые "слипаются" в каком-то промежуточном пункте, Con1 начинается в Москве, а Con2 заканчивается в Санкт-Петербурге (и у них совпадает номер поезда). Тогда это будет многосегментный маршрут.

$$\begin{aligned} & \Pi_{TrainID}(\sigma_{St2=Name \wedge CityName="Санкт-Петербург"}(\Pi_{TrainID, St2}(\sigma_{St1=Name \wedge CityName=Москва} \\ & ((\Pi_{TrainID, Con1.Station1Name \text{ as } St1, Con2.Station2Name \text{ as } St2} \\ & (\sigma_{Con1.TrainID=Con2.TrainID \wedge Con1.Station2Name=Con2.Station1Name \wedge DAY(C2.Arrival)=DAY(C1.Departure)})) \\ & (\rho_{Con1}(Connection) \times \rho_{Con2}(Connection))) \\ & \times Station)) \times Station)) \end{aligned}$$

в) Что изменится в выражениях для а) и б), если отношение "Connection" не содержит дополнительных кортежей для транзитивного замыкания, поэтому многосегментный маршрут Москва-> Тверь-> Санкт-Петербург содержит только кортежи Москва-> Тверь и Тверь-Санкт-Петербург?

В пункте а пришлось бы проверять декартово произведение Connection и искать маршруты, у которых совпадает поезд и город отправления/прибытия (как в пункте б)

В пункте б ничего не изменилось бы

Задача 3

Представьте внешнее объединение (outer join) в виде выражения реляционной алгебры с использованием только базовых операций (select, project, cartesian, rename, union, minus)

$$Join(R, S) = project_{R1, \dots, Rn, S1, \dots, Sm}(select \dots (cartesian(R, S)))$$

$$\begin{aligned} OuterJoin(R, S) = & union(Join(R, S), project_{R1, \dots, Rn, Null, \dots, Null} (R - project_{R1, \dots, Rn}(Join(R, S))), \\ & project_{Null, \dots, Null, S1, \dots, Sm} (S - project_{S1, \dots, Sm}(Join(R, S)))) \end{aligned}$$