# ИИ с нуля - SQL - домашнее задание:

* Следуя логике задач, вам нужно написать запросы, которые будут возвращать данные в требуемом виде.
* Под каждой задачей указан искомый результат в виде скриншота с таблицей. Как этот результат получить - решать вам.
* После каждой задачи впишите ваше решение, чтобы куратор мог его проверить.
* Всё необходимое для решения домашнего задания можно найти в материалах лекции.
* Среда, в которой можно проверить работу запросов - <https://www.sqlitetutorial.net/tryit/>

### 1. Агрегация значений

Вывести дату самой ранней известной покупки из таблицы с покупками.

* Назвать получившийся столбец *first\_date*.

**Образец:**



Answer:

SELECT min(InvoiceDate) as first\_date

FROM invoices;

### 2. Сортировка и уникальные значения

Вывести все страны, в которых когда-либо осуществлялись покупки.

* Названия стран не должны повторяться.
* Отсортировать страны по алфавиту.

**Образец:**



Answer:

SELECT distinct BillingCountry

FROM invoices

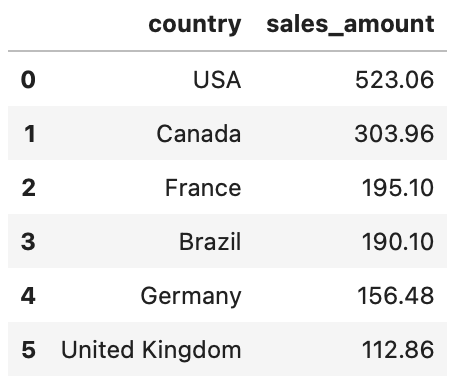
order by BillingCountry ASC

### 3. Агрегация и срезы данных

Вывести суммарные продажи каждой страны.

* Столбец со страной назвать country, а с продажами - *sales\_amount*.
* Оставить только те страны, где продажи более 100$.
* Упорядочить по убыванию *sales\_amount*.

**Образец:**



Answer:

SELECT BillingCountry as country, SUM(Total) AS sales\_amount

FROM invoices

GROUP BY country

HAVING sales\_amount > 100

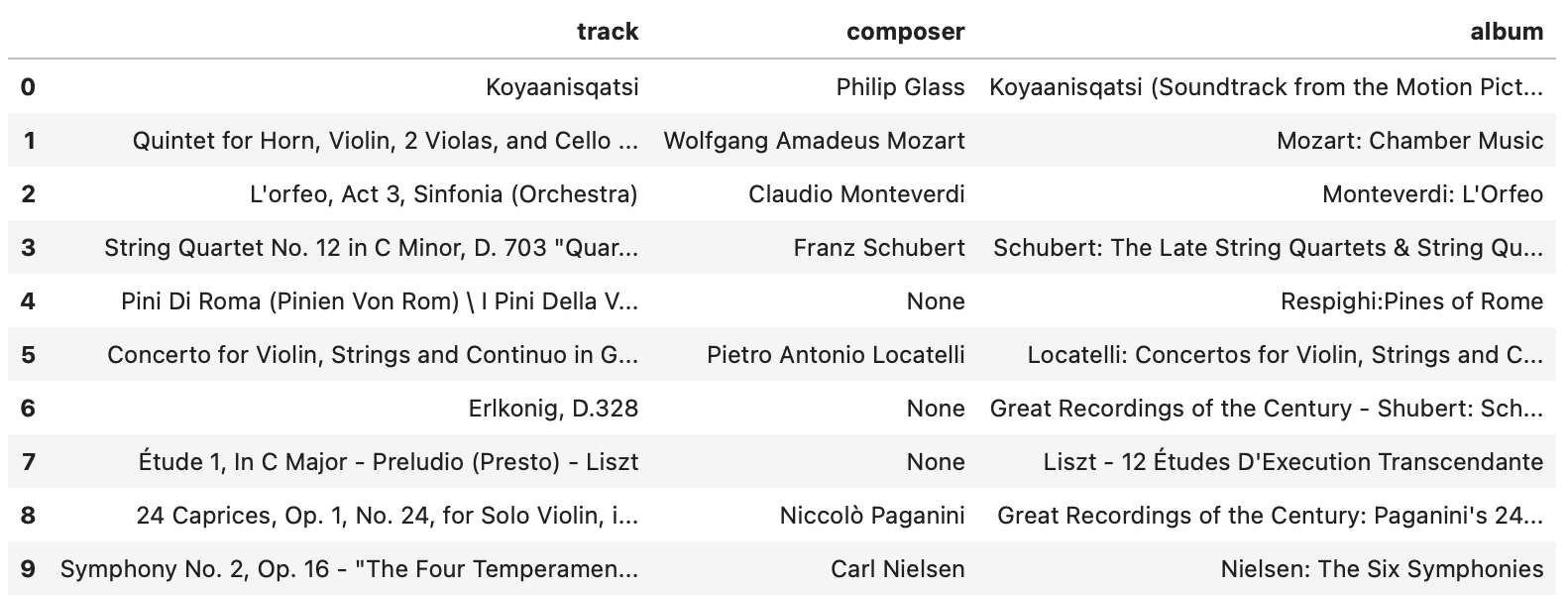
ORDER BY sales\_amount DESC;

### 4. Объединение таблиц

Вывести названия, композиторов и альбомы последних 10 треков. (JOIN)

* Порядок треков определяется их *id*.
* Дать получившимся столбцам имена *track*, *composer* и *album*.
* Пропуски оставить как есть.

**Образец:**



Answer:

SELECT tracks.name AS track, tracks.composer AS composer, albums.title AS album

FROM tracks

JOIN albums

using (albumid)

ORDER BY trackid DESC

limit 10;

### 5. Объединение, группировка, агрегация\* (За дополнительные баллы)

Вывести названия топ-10 композиций по суммарным продажам и сами продажи (назвать столбец *sales\_amount*) в порядке убывания. (GROUP BY, Агрегация, JOIN)

* Если продажи треков равны по сумме, отсортировать названия в алфавитном порядке.
* Столбцы назвать *track* и *sales\_amount*.

**Образец:**



Answer:

SELECT t.name AS track, SUM(t.unitprice) AS sales\_amount

FROM tracks t

LEFT JOIN invoice\_items i USING (trackid)

GROUP BY trackid

ORDER BY sales\_amount DESC, name

LIMIT 10;

### 6. Объединение, срезы данных\* (За дополнительные баллы)

Вывести имена, фамилии и должности подчинённых сотрудника Nancy Edwards. (JOIN, WHERE)

* Подчинённые должны быть отсортированы по дате приёма на работу.
* \*Попробуйте сделать, *не используя CROSS JOIN*: эта процедура требует много ресурсов в реальных условиях.

**Образец:**



Answer:

SELECT e.firstname, e.lastname, e.title

FROM employees e

JOIN employees b

on e.reportsto = b.employeeID

and b.firstname = "Nancy"

and b.lastname = "Edwards"

ORDER BY e.hiredate;