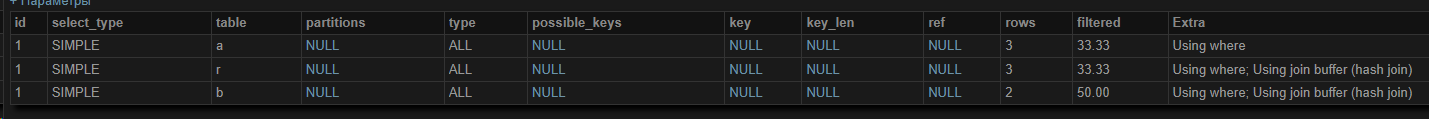
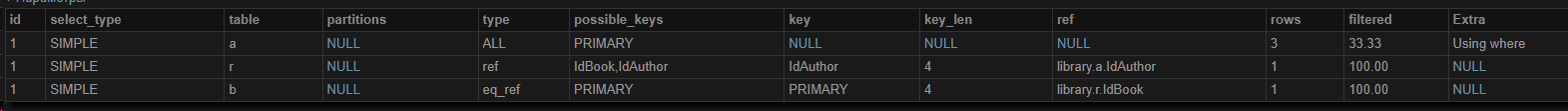
1)

Explain SELECT b.\*, a.Name As AuthorName FROM books As b Inner Join relbookauthor As r On b.IdBook = r.IdBook Inner Join AUTHORS As a On r.IdAuthor = a.IdAuthor where a.NAME = 'Ильф И.А.'

Без индекса необходимо просматривать все три таблицы полностью.



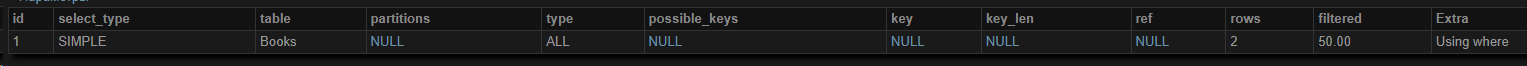
При наличии индексов в таблице AUTHORS будут просмотрены все записи, так как ищется текстовая информация, А в таблицах books И relbookauthor будут просмотрены только по одной записи



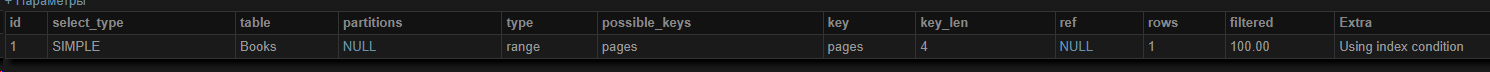
2)

EXPLAIN SELECT \* FROM Books WHERE Pages > 500

Без использования индекса просматриваются все записи таблицы



С использованием индекса была просмотрена только одна запись

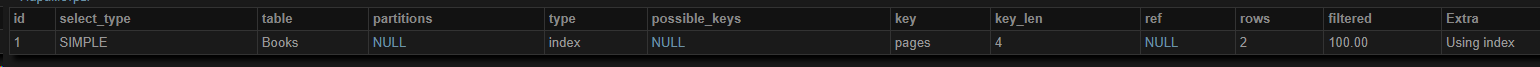


3)

EXPLAIN SELECT Sum(PAGES) As CountPages FROM Books

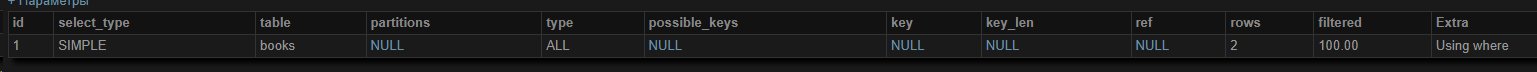
Не зависимо от наличия индекса будут просмотрены все записи, так как необходимо посчитать сумму страниц всех книг

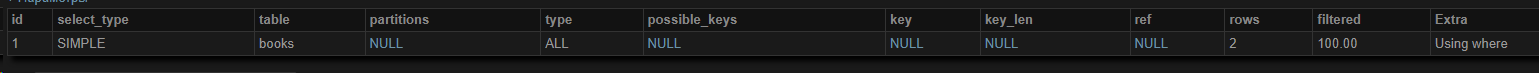




4)

Explain SELECT \*, length(books.NAME)-length(replace(books.NAME,' ' ,''))+1 As CountWord From books Where length(books.NAME)-length(replace(books.NAME,' ' ,''))+1 > 3





В данном запросе индексы не помогут, так как необходимо в каждой записи просчитать количество слов.

Теоретически можно создать колонку, предварительно заполненную рассчитанным количеством слов в наименовании, и ее проиндексировать. Но тогда получается избыточность данных и необходимость отслеживания редактирования наименования книги. Это будет интересно при очень большом количестве записей.