1. Создайте таблицу со столбцами {A integer ,B integer , C integer}. Сгенерируйте 1000 строк в созданную таблицу.

CREATE TABLE test\_table(

A INTEGER,

B INTEGER,

C INTEGER

);

INSERT INTO test\_table

SELECT

ROUND(random() \* 100)::integer,

ROUND(random() \* 100)::integer,

ROUND(random() \* 100)::integer

FROM generate\_series(1, 1000);

1. Создайте всевозможные стандартные Btree индексы для покрытия столбцов и их комбинаций между собой (A)…(AB)(BA)(AC) … (ABC)(BAC) ...

CREATE INDEX idx\_a ON test\_table (A);

CREATE INDEX idx\_b ON test\_table (B);

CREATE INDEX idx\_c ON test\_table (C);

CREATE INDEX idx\_ab ON test\_table (A, B);

CREATE INDEX idx\_ba ON test\_table (B, A);

CREATE INDEX idx\_ac ON test\_table (A, C);

CREATE INDEX idx\_ca ON test\_table (C, A);

CREATE INDEX idx\_bc ON test\_table (B, C);

CREATE INDEX idx\_cb ON test\_table (C, B);

CREATE INDEX idx\_abc ON test\_table (A, B, C);

CREATE INDEX idx\_acb ON test\_table (A, C, B);

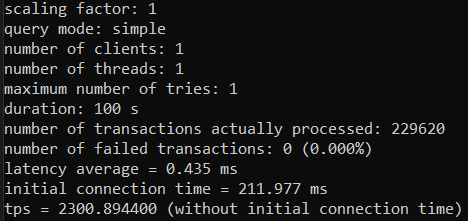
CREATE INDEX idx\_bac ON test\_table (B, A, C);

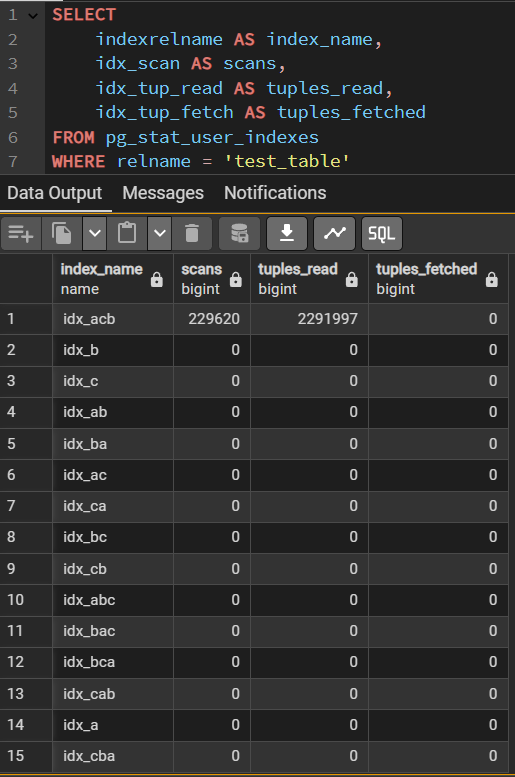
CREATE INDEX idx\_bca ON test\_table (B, C, A);

CREATE INDEX idx\_cab ON test\_table (C, A, B);

CREATE INDEX idx\_cba ON test\_table (C, B, A);

1. Benchmark скрипт на 100 секунд.





Из таблицы видно, что индекс (A, C, B) покрывает данное SQL выражение.