Task: A query frequently runs:

```
SELECT * FROM bookings
WHERE doctor_id = ?
AND appointment_date >= ?
AND appointment_date <= ?;
```

Propose a composite index
Explain column order choice.

Proposed composite index: (doctor_id, appointment_date)

Why this column order makes sense:
doctor_id comes first because the query always filters on a single doctor using equality (=). That sharply narrows the search space right away.
appointment_date comes next because it’s used as a range (>= and <=). Once the database finds rows for that doctor, it can efficiently scan only the required date range.

Putting the range column first would be wasteful, because the index would have to scan many dates across all doctors before filtering by doctor_id.