CREATE DATABASE analytics\_db;

USE analytics\_db;

Database changed

mysql> -- 1. Users Table

mysql> CREATE TABLE users (

-> user\_id INT PRIMARY KEY

-> );

2. Cookies Table

mysql> CREATE TABLE cookies (

-> cookie\_id INT PRIMARY KEY,

-> user\_id INT,

-> FOREIGN KEY (user\_id) REFERENCES users(user\_id)

-> );

3. Visits Table

mysql> CREATE TABLE visits (

-> visit\_id INT PRIMARY KEY,

-> user\_id INT,

-> visit\_time DATETIME,

-> FOREIGN KEY (user\_id) REFERENCES users(user\_id)

-> );

4. Pages Table

mysql> CREATE TABLE pages (

-> page\_id INT PRIMARY KEY,

-> page\_name VARCHAR(100)

-> );

5. Categories Table

mysql> CREATE TABLE categories (

-> category\_id INT PRIMARY KEY,

-> category\_name VARCHAR(100)

-> );

6. Products Table

mysql> CREATE TABLE products (

-> product\_id INT PRIMARY KEY,

-> category\_id INT,

-> FOREIGN KEY (category\_id) REFERENCES categories(category\_id)

-> );

7. Events Table

mysql> CREATE TABLE events (

-> event\_id INT PRIMARY KEY,

-> visit\_id INT,

-> event\_type ENUM('view', 'cart', 'purchase'),

-> page\_id INT,

-> product\_id INT,

-> FOREIGN KEY (visit\_id) REFERENCES visits(visit\_id),

-> FOREIGN KEY (page\_id) REFERENCES pages(page\_id),

-> FOREIGN KEY (product\_id) REFERENCES products(product\_id)

-> );

INSERT INTO users VALUES (1), (2), (3);

INSERT INTO cookies VALUES (1,1), (2,1), (3,2);

INSERT INTO visits VALUES

-> (1, 1, '2025-07-01 10:00:00'),

-> (2, 1, '2025-07-02 11:00:00'),

-> (3, 2, '2025-07-02 12:00:00');

INSERT INTO pages VALUES

-> (1, 'home'),

-> (2, 'product'),

-> (3, 'checkout');

INSERT INTO categories VALUES

-> (1, 'Electronics'),

-> (2, 'Books');

INSERT INTO products VALUES

-> (1, 1),

-> (2, 2);

INSERT INTO events VALUES

-> (1, 1, 'view', 1, NULL),

-> (2, 1, 'view', 2, 1),

-> (3, 1, 'cart', 2, 1),

-> (4, 1, 'purchase', 3, 1),

-> (5, 2, 'view', 1, NULL),

-> (6, 3, 'view', 2, 2);

SELECT COUNT(\*) AS total\_users FROM users;

SELECT AVG(cookie\_count) AS avg\_cookies\_per\_user

-> FROM (

-> SELECT COUNT(\*) AS cookie\_count

-> FROM cookies

-> GROUP BY user\_id

-> ) AS user\_cookies;

SELECT DATE\_FORMAT(visit\_time, '%Y-%m') AS month, COUNT(DISTINCT visit\_id) AS unique\_visits

-> FROM visits

-> GROUP BY month;

SELECT event\_type, COUNT(\*) AS total\_events

-> FROM events

-> GROUP BY event\_type;

SELECT

-> ROUND(COUNT(DISTINCT e.visit\_id) \* 100.0 / (SELECT COUNT(\*) FROM visits), 2) AS purchase\_percentage

-> FROM events e

-> WHERE e.event\_type = 'purchase';

SELECT

-> ROUND(COUNT(\*) \* 100.0 / (SELECT COUNT(\*) FROM visits), 2) AS checkout\_no\_purchase\_percentage

-> FROM (

-> SELECT v.visit\_id

-> FROM visits v

-> JOIN events e1 ON v.visit\_id = e1.visit\_id

-> JOIN pages p ON e1.page\_id = p.page\_id

-> LEFT JOIN events e2 ON v.visit\_id = e2.visit\_id AND e2.event\_type = 'purchase'

-> WHERE p.page\_name = 'checkout' AND e2.event\_id IS NULL

-> GROUP BY v.visit\_id

-> ) AS temp;

SELECT p.page\_name, COUNT(\*) AS view\_count

-> FROM events e

-> JOIN pages p ON e.page\_id = p.page\_id

-> WHERE e.event\_type = 'view'

-> GROUP BY p.page\_id

-> ORDER BY view\_count DESC

-> LIMIT 3;

SELECT c.category\_name,

-> SUM(CASE WHEN e.event\_type = 'view' THEN 1 ELSE 0 END) AS view\_count,

-> SUM(CASE WHEN e.event\_type = 'cart' THEN 1 ELSE 0 END) AS cart\_adds

-> FROM events e

-> JOIN products p ON e.product\_id = p.product\_id

-> JOIN categories c ON p.category\_id = c.category\_id

-> GROUP BY c.category\_id;

SELECT p.product\_id, COUNT(\*) AS purchase\_count

-> FROM events e

-> JOIN products p ON e.product\_id = p.product\_id

-> WHERE e.event\_type = 'purchase'

-> GROUP BY p.product\_id

-> ORDER BY purchase\_count DESC

-> LIMIT 3;