1. INSERT INTO products (name, price, can\_be\_returned) VALUES ('chair', 44.00, false);

2. INSERT INTO products (name, price, can\_be\_returned) VALUES ('stool', 25.99, true);

3. INSERT INTO products (name, price, can\_be\_returned) VALUES ('table', 124.00, false);

4. SELECT \* FROM products;

5. SELECT name FROM products;

6. SELECT name, price FROM products;

7. INSERT INTO products (name, price, can\_be\_returned) VALUES ('Smartphone Holder', 19.99, true);

8. SELECT \* FROM products WHERE can\_be\_returned = true;

9. SELECT \* FROM products WHERE price < 44.00;

10. SELECT \* FROM products WHERE price BETWEEN 22.50 AND 99.99;

11. UPDATE products SET price = price - 20;

12. DELETE FROM products WHERE price < 25;

13. UPDATE products SET price = price + 20;

14. ALTER TABLE products ALTER COLUMN can\_be\_returned SET NOT NULL;

UPDATE products SET can\_be\_returned = true;

----------------------------------------------------------------------------

1. SELECT \* FROM analytics WHERE id = 1880;

2. SELECT id, app\_name FROM analytics WHERE last\_updated = '2018-08-01';

3. SELECT category, COUNT(\*) as app\_count FROM analytics GROUP BY category;

4. SELECT app\_name, COUNT(\*) as review\_count FROM analytics GROUP BY app\_name ORDER BY review\_count DESC LIMIT 5;

5. SELECT app\_name, COUNT(\*) as review\_count FROM analytics WHERE rating >= 4.8 GROUP BY app\_name ORDER BY review\_count DESC LIMIT 1;

6. SELECT category, AVG(rating) as avg\_rating FROM analytics GROUP BY category ORDER BY avg\_rating DESC;

7. SELECT app\_name, price, rating

FROM analytics

WHERE rating < 3

ORDER BY price DESC

LIMIT 1;

8. SELECT app\_name, rating, min\_installs

FROM analytics

WHERE min\_installs <= 50 AND rating IS NOT NULL

ORDER BY rating DESC;

9. SELECT app\_name

FROM analytics

WHERE rating < 3 AND reviews >= 10000;

10. SELECT app\_name, reviews

FROM analytics

WHERE price >= 0.10 AND price <= 1.00

ORDER BY reviews DESC

LIMIT 10;

11. SELECT app\_name, last\_updated

FROM analytics

WHERE last\_updated = (

SELECT MIN(last\_updated)

FROM analytics

);

12. SELECT app\_name, price

FROM analytics

WHERE price = (SELECT MAX(price) FROM analytics)

13. SELECT SUM(reviews) as total\_reviews

FROM analytics;

14. SELECT category, COUNT(\*) as app\_count

FROM analytics

GROUP BY category

HAVING COUNT(\*) > 300;

15. SELECT app\_name, reviews, min\_installs, CAST(min\_installs AS FLOAT)/reviews AS proportion

FROM analytics

WHERE min\_installs >= 100000

ORDER BY CAST(min\_installs AS FLOAT)/reviews DESC

LIMIT 1;