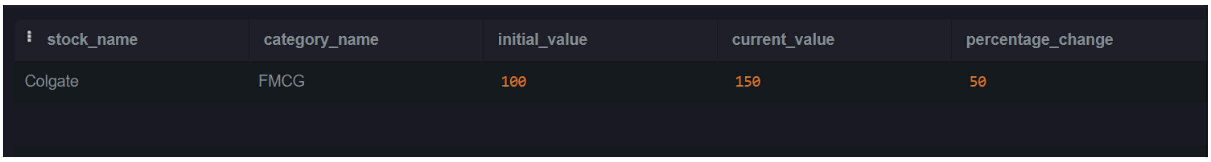
```
CREATE TABLE stocks (
    stock_id INTEGER PRIMARY KEY AUTOINCREMENT,
    stock_name TEXT NOT NULL,
    category_name TEXT NOT NULL,
    initial_value REAL NOT NULL
);
CREATE TABLE purchase_history (
    purchase_id INTEGER PRIMARY KEY AUTOINCREMENT,
    user_id INTEGER NOT NULL,
    stock_id INTEGER NOT NULL,
    current_value REAL NOT NULL,
    FOREIGN KEY (stock_id) REFERENCES stocks(stock_id)
);


INSERT INTO stocks (stock_name, category_name, initial_value) VALUES
('Colgate', 'FMCG', 100.00),
('Dabur', 'FMCG', 150.00),
('Tata', 'Metals', 200.00),
('JSW', 'Metals', 250.00),
('Maruti', 'Automotive', 500.00);


INSERT INTO purchase_history (user_id, stock_id, current_value) VALUES
(1, 1, 150.00),
(1, 2, 140.00),
(1, 3, 220.00),
(1, 4, 270.00),
(1, 5, 550.00);



SELECT
    s.stock_name,
    s.category_name,
    s.initial_value,
    usr.current_value,
    ((usr.current_value - s.initial_value) / s.initial_value * 100) AS
percentage_change
FROM
    purchase_history usr
JOIN
    stocks s ON usr.stock_id = s.stock_id
ORDER BY
    percentage_change DESC
      LIMIT 1;
```

HERE I have created two tables. Table 1 stores information of stocks and Table 2 stores information on stocks purchased by the user. This query fetches and displays the stock name, category, initial value, current value, and percentage change for each stock, ordered by category. The result is limited to the first row, showing which stock category has moved the highest.

#OUTPUT



stock_name	category_name	initial_value	current_value	percentage_change
Colgate	FMCG	100	150	50