Стан	Завершено
Розпочато	неділю 23 лютого 2025 09:31 AM
Завершено	неділю 23 лютого 2025 09:36 AM
Затрачений час	4 хв 18 сек
Оцінка	**9,50** з можливих 10,00 (**95**%)

Питання 1

Правильно

Балів 1,00 з 1,00

What is a SQL View?

a. A physical table that stores data

b. A query that retrieves and displays data from one or more tables ✓

c. A stored procedure that performs a set of operations on data

d. A group of related tables that are linked together

Your answer is correct.

Питання 2

Правильно

Балів 1,00 з 1,00

Which type of view combines data from one or more base tables (or views) into a new virtual table?

a. Standard views ✓

b. Indexed views

c. Partitioned views

d. Logical views

Your answer is correct.

Питання 3

Правильно

Балів 1,00 з 1,00

What is the benefit of using materialized (indexed) views?

a. They can combine data from one or more base tables (or views) into a new virtual table

b. They can help to simplify complex queries and join data from multiple tables

c. They can materialize (persist) the the results of view in a table-like form ✓

d. They can join horizontally partitioned data from one or more base tables across one or more servers

Your answer is correct.

Питання 4

Правильно

Балів 1,00 з 1,00

What is the difference between a regular view and a materialized view in PostgreSQL?

a. Regular views are read-only, while materialized views allow data modification

b. Regular views store data physically on disk, while materialized views are virtual

c. Materialized views are not updated automatically when underlying data changes, while regular views always reflect real-time data ✓

d. There is no difference; both terms are used interchangeably in PostgreSQL

Your answer is correct.

Питання 5

Правильно

Балів 1,00 з 1,00

Suppose you have two tables in a PostgreSQL database, `orders` and `products`. You want to create a view that shows the order ID, product name, and quantity for all orders that include a particular product. Which of the following CREATE VIEW statements would accomplish this?

Table Example:
`orders` table:

order_id	product_id	quantity
1	1	3
2	2	5
3	1	2
4	3	1

`products` table:

id	product_name
1	Widget A
2	Widget B
3	Widget C

a.
```
CREATE VIEW order_product AS
SELECT order_id, product_name, quantity
FROM orders
JOIN products ON orders.product_id = products.id
WHERE product_name = 'example_product';
```

b.
```
CREATE VIEW order_product AS
SELECT orders.order_id, products.product_name, orders.quantity
FROM orders, products
WHERE orders.product_id = products.id
AND products.product_name = 'example_product';
```

c.
```
CREATE VIEW order_product AS
SELECT order_id, product_name, quantity
FROM orders
JOIN products ON orders.product_id = products.id
HAVING product_name = 'example_product';
```

d.
```
CREATE VIEW order_product AS
SELECT orders.order_id, products.product_name, orders.quantity
FROM orders, products
WHERE orders.product_id = products.id
AND products.product_name LIKE '%example_product%';
```

Your answer is correct.

Питання 6

Частково правильно

Балів 0,50 з 1,00

Suppose you have a PostgreSQL database that contains three tables, `orders`, `order_details` and `customers`. The `orders` table contains basic information about each order, including the order ID and customer ID, `customers` table contains information about customer, while the `order_details` table contains the details of each order, including the product ID, quantity, and price. You want to create a view that shows the order ID, customer name, product name, quantity, and price for all orders.
Example of the `orders` table:

order_id	customer_id	order_date
1	1	2022-03-01
2	2	2022-03-02
3	1	2022-03-03
4	3	2022-03-04

Example of the `order_details` table:

order_id	product_id	quantity	price
1	1	3	10.00
2	2	5	15.00
1	3	2	20.00
4	1	1	8.50

Example of the `customers` table:

customer_id	customer_name	city
1	Laurence Lebihan	London
2	Diego Roel	Madrid

Which of the following CREATE VIEW statements would accomplish this?

a.
```
CREATE VIEW order_details_view AS
SELECT o.order_id, c.customer_name, d.product_name, d.quantity, d.price
FROM orders o, order_details d, customers c
WHERE o.order_id = d.order_id
AND o.customer_id = c.customer_id;
```

b.
```
CREATE VIEW order_details_view AS
SELECT o.order_id, c.customer_name, d.product_name, d.quantity, d.price
FROM orders o
JOIN order_details d ON o.order_id = d.order_id
JOIN customers c ON o.customer_id = c.customer_id;
```

c.
```
CREATE VIEW order_details_view AS
SELECT order_id, customer_name, product_name, quantity, price
FROM orders
JOIN order_details ON orders.order_id = order_details.order_id
JOIN customers ON orders.customer_id = customers.customer_id;
```

d.
```
CREATE VIEW order_details_view AS
SELECT order_id, customer_name, product_name, quantity, price
FROM orders
INNER JOIN order_details ON orders.order_id = order_details.order_id
INNER JOIN customers ON orders.customer_id = customers.customer_id;
```

Your answer is partially correct.

У вас правильних відповідей: 1.

Питання 7

Правильно

Балів 1,00 з 1,00

Suppose you have two tables in a PostgreSQL database, `customers` and `orders`, with the following data:
`customers` table:

customer_id	customer_name	address	city
1	John Smith	123 Main St.	Anytown
2	Jane Doe	456 Oak St.	Other town
3	Bob Johnson	789 Maple Ave.	Another town

`orders` table:

order_id	customer_id	order_date	total_amount
1	1	2022-03-01	150.00
2	2	2022-03-02	75.00
3	1	2022-03-03	25.00
4	3	2022-03-04	50.00

Which of the following CREATE VIEW statements would create a vertical view that displays the customer name, order ID, and total amount for all orders?

a.
```
CREATE VIEW customer_orders_view AS
SELECT customer_name, order_id, total_amount
FROM customers, orders
WHERE customers.customer_id = orders.customer_id
GROUP BY customers.customer_id;
```

b.
```
CREATE VIEW customer_orders_view AS
SELECT c.customer_name, o.order_id, o.total_amount
FROM customers c
JOIN orders o ON c.customer_id = o.customer_id;
```

c.
```
CREATE VIEW customer_orders_view AS
SELECT customer_name, order_id, total_amount
FROM customers
LEFT JOIN orders ON customers.customer_id = orders.customer_id;
```

d.
```
CREATE VIEW customer_orders_view AS
SELECT c.customer_name, o.order_id, o.total_amount
FROM customers c
LEFT JOIN orders o ON c.customer_id = o.customer_id;
```

Your answer is correct.

Питання 8

Правильно

Балів 1,00 з 1,00

Suppose you have a PostgreSQL database that contains a table named `sales`, which contains information about sales transactions. The `sales` table has columns for the date of the transaction, the name of the salesperson, the product sold, and the amount of the sale. You want to create a grouped view that shows the total sales amount for each salesperson and product combination.
Example of the `sales` table:

sale_date	salesperson	product	sale_amount
2022-03-01	John Smith	Product A	100.00
2022-03-01	Jane Doe	Product B	75.00
2022-03-02	John Smith	Product C	50.00
2022-03-03	Bob Johnson	Product A	25.00
2022-03-03	Jane Doe	Product B	150.00

Which of the following CREATE VIEW statements would accomplish this?

a. CREATE VIEW sales_view AS ✓
SELECT salesperson, product, SUM(sale_amount) AS total_sales
FROM sales
GROUP BY salesperson, product;

b. CREATE VIEW sales_view AS
SELECT salesperson, product, sale_amount
FROM sales
GROUP BY salesperson, product;

c. CREATE VIEW sales_view AS
SELECT salesperson, product, AVG(sale_amount) AS total_sales
FROM sales
GROUP BY salesperson, product;

d. CREATE VIEW sales_view AS
SELECT salesperson, product, MAX(sale_amount) AS total_sales
FROM sales
GROUP BY salesperson, product;

Your answer is correct.

Питання 9

Правильно

Балів 1,00 з 1,00

Which of the following is an advantage of using views in PostgreSQL?

a. Increased complexity of the database schema

b. Reduced data independence

c. Improved security by limiting access to specific data ✓

d. Slower database performance

Your answer is correct.

Питання 10

Правильно

Балів 1,00 з 1,00

What is view materialization in PostgreSQL?

a. The process of creating a new view by combining data from multiple tables

b. The process of storing the result of a query physically and update the data periodically. ✓

c. The process of modifying the structure of a view without affecting the underlying tables

d. The process of restricting access to a view based on user permissions

Your answer is correct.