10 advanced SQL interview practical query questions along with their solutions

1. Question: Retrieve the top 5 highest-paid employees for each department, sorted by salary in descending order.

   Solution

```sql
SELECT
  department,
  employee_name,
  salary
FROM
  (
    SELECT
      department,
      employee_name,
      salary,
      ROW_NUMBER() OVER (
        PARTITION BY department
        ORDER BY
          salary DESC
      ) AS rank
    FROM
      employees
  ) ranked
WHERE
  rank <= 5;
```

2. Question: Calculate the total sales for each month of the current year, including months with zero sales.

```sql
SELECT
  to_char(sale_date, 'YYYY-MM') AS month,
  COALESCE(
    SUM(sales_amount),
    0
  ) AS total_sales
FROM
  generate_series(
    DATE_TRUNC('YEAR', CURRENT_DATE),
    DATE_TRUNC('YEAR', CURRENT_DATE) + INTERVAL '1 year' - INTERVAL '1 day',
    INTERVAL '1 month'
  ) AS months(sale_date)
  LEFT JOIN sales ON to_char(sale_date, 'YYYY-MM') = to_char(sales_date, 'YYYY-MM')
GROUP BY
  month;
```

3. Question: Find customers who have made a purchase every month for the last six months.

```sql
SELECT
  customer_id
FROM
  customers
WHERE
  date_trunc('month', CURRENT_DATE) - INTERVAL '6 months' <= ALL (
    SELECT
      date_trunc('month', purchase_date)
    FROM
      purchases
    WHERE
      customer_id = customers.customer_id
  );
```

4. Question: Calculate the running total of sales for each day within the past month.

```sql
SELECT
  date,
  SUM(sales_amount) OVER (
    ORDER BY
      date
  ) AS running_total
FROM
  generate_series(
    DATE_TRUNC('MONTH', CURRENT_DATE) - INTERVAL '1 month',
    DATE_TRUNC('MONTH', CURRENT_DATE) - INTERVAL '1 day',
    INTERVAL '1 day'
  ) AS dates(date)
  LEFT JOIN sales ON dates.date = sales.sales_date;
```

5. Question: List the products that have been sold in all cities where the company operates.

```sql
SELECT
  product_id,
  product_name
FROM
  products
WHERE
  product_id NOT IN (
    SELECT
      DISTINCT product_id
    FROM
      sales
    WHERE
      city NOT IN (
        SELECT
          DISTINCT city
        FROM
          locations
      )
  );
```

6. Question: Retrieve the top 10 customers who have spent the most on their single purchase.

```sql
SELECT
  customer_id,
  MAX(purchase_amount) AS max_purchase_amount
FROM
  purchases
GROUP BY
  customer_id
ORDER BY
  max_purchase_amount DESC
LIMIT
  10;
```

7. Question: Find the employees who manage the same number of employees as their manager.

```sql
SELECT
  e1.employee_name AS employee,
  e1.managed_count AS direct_reports
FROM
  employees e1
  JOIN employees e2 ON e1.manager_id = e2.employee_id
WHERE
  e1.managed_count = e2.managed_count;
```

8. Question: Calculate the 30-day moving average of sales for each product.

```sql
SELECT
  product_id,
  sales_date,
  sales_amount,
  AVG(sales_amount) OVER (
    PARTITION BY product_id
    ORDER BY
      sales_date RANGE BETWEEN INTERVAL '30 days' PRECEDING
      AND CURRENT ROW
  ) AS moving_avg
FROM
  sales;
```

9. Question: List the departments where the average salary is higher than the company's overall average salary.

```sql
SELECT
  department
FROM
  employees
GROUP BY
  department
HAVING
  AVG(salary) > (
    SELECT
      AVG(salary)
    FROM
      employees
  );
```

10. Question: Retrieve the top 3 most recent orders for each customer.

```sql
SELECT
  customer_id,
  order_id,
  order_date
FROM
  (
    SELECT
      customer_id,
      order_id,
      order_date,
      ROW_NUMBER() OVER (
        PARTITION BY customer_id
        ORDER BY
          order_date DESC
      ) AS rank
    FROM
      orders
  ) ranked
WHERE
  rank <= 3;
```