1) Find monthly order totals in 1997. Calculate order totals as (unit_price * quantity)*(1- discount).

Northwind database, orders and order_details tables.

```
SELECT
DATE_PART('month', order_date) AS order_month,
SUM((unit_price * quantity)*(1- discount)) AS order_total
FROM orders o
INNER JOIN order_details od
ON o.order_id = od.order_id
WHERE DATE_PART('year', order_date) = 1997
GROUP BY order_month
ORDER BY order_month;
```

2) Find employees who have been hired more than 32 years ago. ***Northwind database, employees table.***

```
SELECT *
FROM employees e
WHERE hire_date < CURRENT_DATE - INTERVAL '32 years';
```

3) Find the average order processing days (order processing day = shipped_date – order_date).

Northwind database, orders table.

```
SELECT AVG(shipped_date - order_date) AS average_processing_days
FROM orders;
```

4) Find out the number of times movie “King Evolution” rented in each month of 2005? ***Dvdrental database, film, inventory, and rental tables.***

SELECT DATE_PART('month', rental_date) AS rental_month, COUNT(rental_id) AS rental_count

FROM film f

INNER JOIN inventory i

ON f.film_id = i.film_id

INNER JOIN rental r

on i.inventory_id = r.inventory_id

WHERE DATE_PART('year', rental_date) = 2005 AND title = 'King Evolution'

GROUP BY rental_month

ORDER BY rental_month;

5) Find the monthly number of orders by truncating the order date to the start of each month. ***Northwind database, orders table.***

SELECT DATE_TRUNC('month', order_date) as start_of_month, COUNT(*) AS order_count

FROM orders

GROUP BY start_of_month

ORDER BY start_of_month;

6) Find the number of rentals on an hourly basis to understand the peak rental times. ***Dvdrental database, rental table.***

SELECT DATE_PART('hour', rental_date) as rental_hour, COUNT(*) AS order_count

FROM rental

GROUP BY rental_hour

ORDER BY order_count DESC;

7) Retrieve the list of unique film titles in the dvdrental database that were rented more than 19 years, 1 month, and 8 days ago based on the most recent rental date. ***Dvdrental database, film, inventory, and rental tables.***

SELECT DISTINCT(title), MAX(rental_date) AS recent_rental_date

FROM film f

INNER JOIN inventory i

ON f.film_id = i.film_id

INNER JOIN rental r

on i.inventory_id = r.inventory_id

GROUP BY DISTINCT(title)

HAVING MAX(rental_date) < CURRENT_DATE - INTERVAL '19 years 1 month 8 days';

8) Retrieve the number of employees hired in each year, but only include years where more than 5 employees were hired. Sort the years in a descending way. ***HR database, employees table.***

SELECT DATE_PART('year', hire_date) as hire_year, COUNT(*) AS number_of_hires

FROM employees

GROUP BY hire_year

HAVING COUNT(*) > 5

ORDER BY hire_year DESC;

9) Calculate the exact tenure of each employee (in years and months) since their hire date and display the result along with employee_id, first_name, last_name, and department names. ***HR database, employees and departments table.***

SELECT employee_id, first_name, last_name, department_name, AGE(CURRENT_DATE, hire_date) AS tenure

FROM employees e

INNER JOIN departments d

ON e.department_id = d.department_id;

10) Display each employee's hire date as shown below in the **formatted_hire_date column**, along with how many years they've been with the company, employee_id, first_name, and last_name**.**

HR database, employees table.

	employee_id [PK] integer	first_name character varying (20)	last_name character varying (25)	formatted_hire_date text	years_with_company numeric
1	100	Steven	King	Wednesday, 17 June 1987	37
2	101	Neena	Kochhar	Thursday , 21 September 1989	35
3	102	Lex	De Haan	Wednesday, 13 January 1993	31
4	103	Alexander	Hunold	Wednesday, 03 January 1990	34

SELECT employee_id, first_name, last_name, TO_CHAR(hire_date, 'Day, DD Month YYYY') AS formatted_hire_date,

DATE_PART('year', CURRENT_DATE) - DATE_PART('year', hire_date) AS years_with_company

FROM employees;