Time: 60 minutes   **Marks: 10+10+10**

Question 1.

Write a PL/SQL function named **IS_READY_FOR_PROMOTION** that takes an **EMPLOYEE_ID** as input and returns a **VARCHAR2** result indicating whether the employee is ready for promotion.

The function should return:

- **YES** if the employee meets **all** the following criteria:
    - Has worked for at least 5 years since the **HIRE_DATE**
    - Has a **SALARY** greater than the **midpoint** of their job's **MIN_SALARY** and **MAX_SALARY**
    - Manages at least one subordinate.
- **NO** if the employee exists but fails any of the above conditions

For demonstration, you need to call the function **IS_READY_FOR_PROMOTION** for each employee and output whether they are eligible or not. Make sure to handle exceptions with appropriate messages.

```
..
..
IS 117 READY FOR PROMOTION? NO
IS 118 READY FOR PROMOTION? NO
IS 119 READY FOR PROMOTION? NO
IS 120 READY FOR PROMOTION? YES
IS 121 READY FOR PROMOTION? YES
IS 122 READY FOR PROMOTION? YES
IS 123 READY FOR PROMOTION? NO
IS 124 READY FOR PROMOTION? NO
..
..
```

Question 2.

First create a copy of the **EMPLOYEES** table in the HR schema, named **EMPLOYEES_COPY.** Do all the work in **EMPLOYEES_COPY** table.

Write a PL/SQL procedure that updates the salaries of all employees. The new salary for an employee will be calculated as follows:

```
New Salary = Old Salary + (Commission Percentage * Old Salary) + 0.1 * (Min
Salary of Employee's Job) + 0.1 * (Average Salary of Employee's Department)
```

Conditions:

1. If the employee's work period is **1 year or less**, his/her salary will **not** be updated. (Use the hire date of the newest employee as today's date instead of SYSDATE).
2. If the employee's new salary exceeds the **maximum salary** for his/her job, set the new salary to the **maximum salary** for that job.

The procedure should loop through all employees to update the salary. Handle appropriate exceptions.

To copy a table, execute the following script-

```
CREATE TABLE employees_copy AS
SELECT * FROM employees;
COMMIT;
```

To show output:

```
SELECT e.employee_id, e.salary as Old_Salary, ec.salary as New_Salary,
j.min_salary, j.max_salary, e.hire_date
FROM employees_copy ec join employees e
on ec.employee_id=e.employee_id
join jobs j on e.job_id=j.job_id;
```

	EMPLOYEE_ID	OLD_SALARY	NEW_SALARY	MIN_SALARY	MAX_SALARY	HIRE_DATE
1	100	24000	27941.33	20080	40000	17-JUN-03
2	101	17000	20564.71	15000	30000	21-SEP-05
3	102	17000	20683.53	15000	30000	13-JAN-01
4	103	9000	9976	4000	10000	03-JAN-06
5	104	6000	6000	4000	10000	21-MAY-07
6	105	4800	5795.52	4000	10000	25-JUN-05
7	106	4800	5815.43	4000	10000	05-FEB-06
8	107	4200	5235.74	4000	10000	07-FEB-07
9	108	12008	13688.13	8200	16000	17-AUG-02
10	109	9000	9000	4200	9000	16-AUG-02
11	110	8200	9000	4200	9000	28-SEP-05
12	111	7700	9000	4200	9000	30-SEP-05
13	112	7800	9000	4200	9000	07-MAR-06
14	113	6900	6900	4200	9000	07-DEC-07
15	114	11000	12215	8000	15000	07-DEC-02
16	115	3100	3785.25	2500	5500	18-MAY-03
17	116	2900	3596.67	2500	5500	24-DEC-05
18	117	2800	3508.28	2500	5500	24-JUL-05
19	118	2600	3320.09	2500	5500	15-NOV-06
20	119	2500	2500	2500	5500	10-AUG-07
21	120	8000	8500	5500	8500	18-JUL-04
22	121	8200	8500	5500	8500	10-APR-05
23	122	7900	8500	5500	8500	01-MAY-03
24	123	6500	7400.67	5500	8500	10-OCT-05
25	124	5800	5800	5500	8500	16-NOV-07
26	125	3200	3753.47	2008	5000	16-JUL-05
27	126	2700	3254.7	2008	5000	28-SEP-06
28	127	2400	2955.93	2008	5000	14-JAN-07
29	128	2200	2200	2008	5000	08-MAR-08
30	129	3300	3857.17	2008	5000	20-AUG-05

Question 3.

Create a trigger that activates when an employee's salary is updated.

Salary decrease>20% → demotion

1 new table: Demotions
(fields:
        employee_id,
        current_salary,
        Status- waiting and done,
        date
)

In case of demotion:
1. No change if that employee is not a manager
2. If that employee is a manager, switch him/her with the highest-paid employee under him/her. (**Do not switch salaries)**

No changes in the Job table and the Job_history table are necessary for your ease. All current date is the hire date of the newest employee instead of SYSDATE.