Exercise 4 : SQL JOINS

1)
```
SELECT student_id,
       student_name,
       grade
FROM students AS A
INNER JOIN grades AS B
ON A.student_id = B.student_id;
```

student_id	student_name	grade
2	Bob	B
3	Charlie	A

2)
```
SELECT emp_id,
       emp_name
       dept_name
FROM employees AS A
LEFT JOIN departments AS B
ON A.emp_id = B.emp_id
```

emp_id	emp_name	dept_name
1	John	NULL
2	Lisa	HR
3	Mike	NULL

see back for correct JS

3)
```
SELECT product_id,
       product_name,
       quantity
FROM products AS A
FULL OUTER JOIN sales AS B
ON A.product_id = B.product_id;
```

product_id	product_name	quantity
1	Laptop	NULL
2	Mouse	50
3	Keyboard	NULL
4	NULL	30

4)
```
SELECT order_id
       customer_id,
       amount
       customer_name,
CASE   WHEN B.customer_id IS NOT NULL THEN 'Returning Customer'
       ELSE 'New Customer'
END AS customer_type
FROM orders AS A
LEFT JOIN customers AS B
ON A.customer_id = B.customer_id;
```

order_id	customer_id	amount	customer_name	customer_type
1	101	500	Paul	Returning Customer
2	102	300	Sarah	Returning Customer
3	105	0	NULL	New Customer

5

```
SELECT region_id,
       region_name,
       SUM(amount) AS total_sales
FROM regions AS A
LEFT JOIN sales AS B
ON A.region_id = B.region_id
GROUP BY A.region_id, region_name;
```

region_id	region_name	total_sales
1	North	2000
2	South	3500
3	East	NULL

6)

```
SELECT student_id
       name,
       days_present,
CASE   WHEN days_present >= 15 THEN 'Excellent'
       WHEN days_present BETWEEN 6 AND 14 THEN 'Needs Improvement'
       WHEN days_present <= 5 THEN 'Poor Attendance'
    ELSE 'No Record'
    END AS attendance_status
FROM students AS A
LEFT JOIN attendance AS B
ON A.student_id = B.Student_id;
```

Student id	name	days_present	attendance_status
1	Alice	18	Excellent
2	Bob	8	Poor Attendance
3	Charlie	NULL	No Record

7)

```
SELECT project_id,
       name,
       COUNT(task_id) AS task_count
FROM projects AS A
INNER JOIN tasks AS B
ON A.project_id = B.project_id
GROUP BY A.project_id, name;
```

project_id	name	task_count
1	AI Chatbot	2
2	Website	1

8.
```
SELECT COALESCE (A.cust_id, B.cust_id) AS cust_id,
        order_total,
        return_total,
CASE    WHEN return_total IS NOT NULL THEN
        ELSE 'No Return'
  END AS return_status
FROM orders AS A
FULL OUTER JOIN returns AS B
ON A.cust_id = B.cust_id
WHERE order_total > 100;
```

cust_id	order total	return total	return status
11	120	20	Returned
12	250	NULL	No Return
13	180	NULL	No Return

9.
```
SELECT user_id,
       name,
       COUNT (login_date) AS login_count
FROM users AS A
LEFT JOIN logins AS B
ON A.user_id = B.user_id
GROUP BY A.user_id, name
ORDER BY login_count DESC;
```

user id	name	login_count
2	Gloria	2
3	Steve	1
1	Nelson	0

10.
```
SELECT A.teacher_id,
       teacher name
   COALESCE (subject_name, 'NO Subject Assigned') AS subject_name
FROM teachers AS A
LEFT JOIN subjects AS B
ON A.teacher_id = B.teacher_id
ORDER BY teacher_name ASC;
```

teacher_id	teacher_name	subject_name
3	Mr Diamini	No Subject Assigned
1	Mr. Hlongwane	Math
1	Mr. Hlongwane	Science
2	Ms. Ndaba	No Subject Assigned