Excercise 4 - SQL JOIN

```
1. SELECT       A.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	Charle	A

```
2. SELECT       A.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

```
3. SELECT       COALESCEA (A product_id, B.product_id)
                AS product_id, product_name, quantity
   FROM         product 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	80
3	keyboard	NULL
4	NULL	30

```
4. SELECT        order_id, A.customer_id, amount, customer_name,
                 CASE
                  WHEN B customer_id IS NOT NULL THEN
                   'Returning Customer'
                  ElSE 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	900	Paul	Returning customer
2	102	300	Sarah	Returning Custome
3	105	0	Null	New Customer

```
5. SELECT       A.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       A.student_id, name, days_present,
   CASE         WHEN days_present >= 15 THEN 'Excellent'
                WHEN days_prescent BETWEEN 6 & 14 THEN 'Needs Improvement'
                WHEN days_present <= 5 THEN 'Poor Attendance'
                ELSE 'No Record'
   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	5	Poor Attendance
3	Charle	NULL	No Record

```
7. SELECT       A.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_totals IS NOT NULL THEN 'Returned'
              ELSE 'No Return'
   END        AS return_status
   FROM       orders AS A
   FULL OUTER JOIN returns AS B
   ON  A.cust_id = B.cust_id
```

cust_id	order_total	return total	return_status
11	120	20	Returned
12	250	NULL	No Return
13	180	NULL	No Return

```
9. SELECT     A.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. Dlamini	No Subject Assigned
1	Mr Hlongwane	Math
1	Mr Hlongwane	Science
2	Ms Ndabo	No Subject Assigned