



DEPARTMENT OF

Discover. Learn. Empower.

Experiment 3

Student Name: Jay shankar kumar

Branch: CSE

Semester: 5th

Subject Name: ADBMS

UID: 23BCS10408

Section/Group: KRG_1B

Date of perf: 14 Aug, 2025

Subject Code: 23CSP-333

1. Aim of the practical

[EASY] Basic Table Creation and Duplicate Handling: Operations

- Generate an Employee relation with a single attribute ID.
- Retrieve the maximum ID value while excluding duplicates.

CODE:

```
-----EASY-----  
  
CREATE TABLE department (  
    id INT PRIMARY KEY,  
    dept_name VARCHAR(50)  
);  
  
-- Create Employee Table  
CREATE TABLE employee (  
    id INT,  
    name VARCHAR(50),  
    salary INT,  
    department_id INT,  
    FOREIGN KEY (department_id) REFERENCES department(id)  
);  
  
-- Insert into Department Table  
INSERT INTO department (id, dept_name) VALUES  
(1, 'IT'),  
(2, 'SALES');  
  
-- Insert into Employee Table  
INSERT INTO employee (id, name, salary, department_id) VALUES  
(1, 'JOE', 70000, 1),  
(2, 'JIM', 90000, 1),  
(3, 'HENRY', 80000, 2),  
(4, 'SAM', 60000, 2),  
(4, 'MAX', 90000, 1);  
  
select D.dept_name, E.name, E.salary  
from employee as E  
inner join  
department as D  
on D.id = E.department_id  
WHERE E.salary in (
```



DEPARTMENT OF

Discover. Learn. Empower.

```
select MAX(E2.salary)
from employee as E2
WHERE E2.department_id = E.department_id
)
```

OUTPUT:

	id	dept_name
1	1	IT
2	2	SALES

	id	name	salary	department_id
1	1	JOE	70000	1
2	2	JIM	90000	1
3	3	HENRY	80000	2
4	4	SAM	60000	2
5	4	MAX	90000	1

	dept_name	name	salary
1	SALES	HENRY	80000
2	IT	MAX	90000
3	IT	JIM	90000

Query executed successfully. | EDITH\SQLEXPRESS (16.0 RTM) | EDI

[MEDIUM] 2. Product Sales Analysis

- Select products which have never been sold.
- Calculate the total quantity sold for each respective product.

CODE:

-----MEDIUM-----

```
create table emp1(
    id int primary key,
    Ename varchar(20),
    salary int
);
create table emp2(
    id int primary key,
    Ename varchar(20),
    salary int
);
-- Insert into emp1
INSERT INTO emp1 (id, Ename, salary) VALUES
(1, 'Alice', 50000),
(2, 'Bob', 60000),
(3, 'Charlie', 55000),
(4, 'David', 70000),
(5, 'Eva', 65000);

-- Insert into emp2
INSERT INTO emp2 (id, Ename, salary) VALUES
```

DEPARTMENT OF

Discover. Learn. Empower.

```
(2, 'Bob', 62000),      -- Same ID & Name as emp1, salary
different
(3, 'Charlie', 58000), -- Same ID & Name, salary different
(5, 'Eva', 66000),     -- Same ID & Name, salary different
(6, 'Frank', 72000),
(7, 'Grace', 68000);
select * from emp1;
select * from emp2;
select id,Ename,min(salary) as salary
from
(
select * from emp1
union all
select* from emp2
) as INTERMEDIATE_RESULT
group by id,Ename;
```

OUTPUT:

90 % ✓ No issues found

Results		Messages	
	id	Ename	salary
1	1	Alice	50000
2	2	Bob	60000
3	3	Charlie	55000
4	4	David	70000
5	5	Eva	65000

	id	Ename	salary
1	2	Bob	62000
2	3	Charlie	58000
3	5	Eva	66000
4	6	Frank	72000
5	7	Grace	68000

	id	Ename	salary
1	1	Alice	50000
2	2	Bob	60000
3	3	Charlie	55000
4	4	David	70000
5	5	Eva	65000
6	6	Frank	72000
7	7	Grace	68000

✓ Query executed successfully.

Learning Outcomes:

- Learn how to define and create relational database tables using CREATE TABLE syntax. Understand the use of data types like INT and VARCHAR.
- Gain practical knowledge of establishing a primary key for uniquely identifying records.
- Understand how to create and enforce foreign key relationships to maintain data integrity between related tables (Books → Authors).
- Develop the ability to use INNER JOIN to combine data from multiple tables based on a common key (e.g. author_id).
- Understand how to design normalized relational tables with foreign key constraints for real-world entities like departments and courses.
- Gain proficiency in inserting multiple records into related tables using the INSERT INTO statement.
- Learn how to use subqueries with GROUP BY and HAVING to aggregate data and apply conditional logic.
- Apply filtering logic to retrieve records from a parent table based on results from a subquery on a related child table.