Write a SQL query to find the top 3 departments with the highest average salary of employees. Ensure departments with no employees show an average salary of NULL.

CREATE TABLE departments ( department\_id NUMBER PRIMARY KEY, department\_name VARCHAR2(100) NOT NULL );

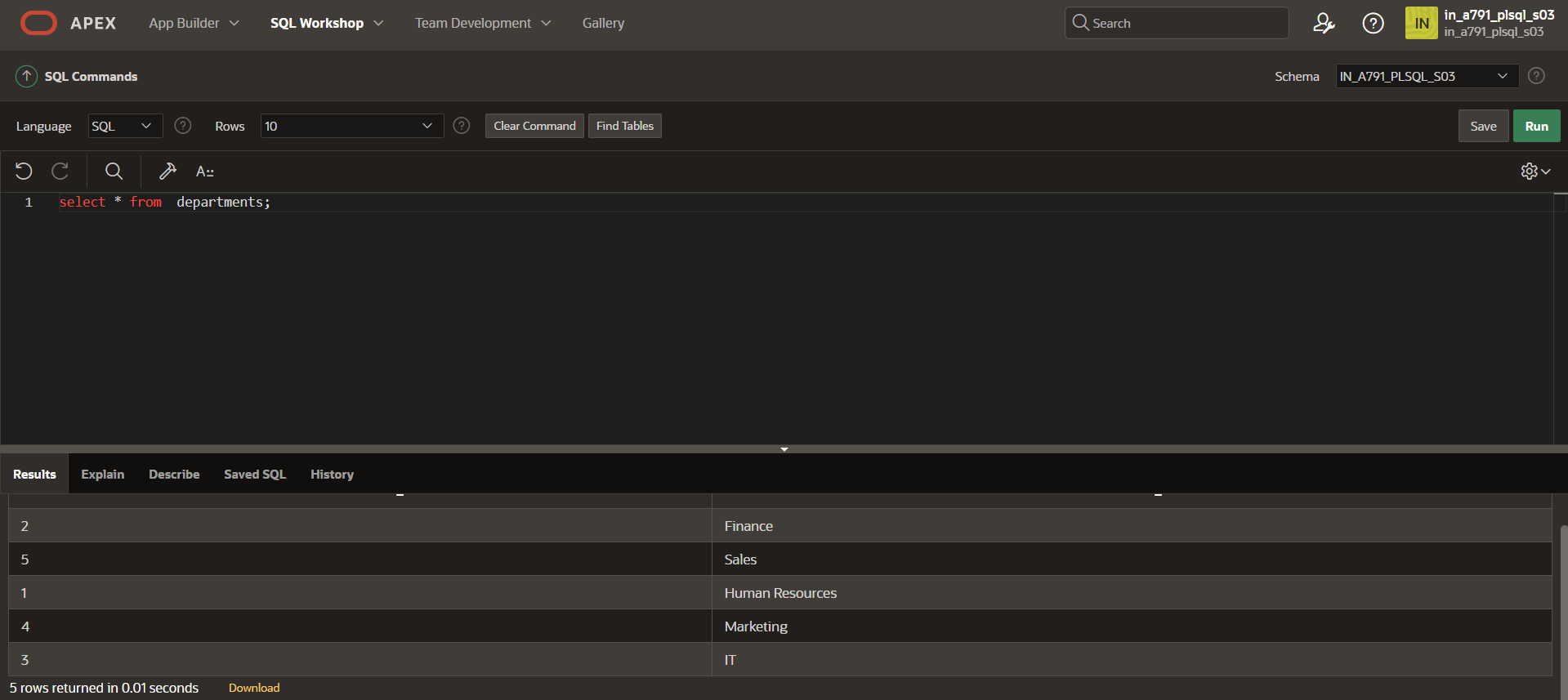
CREATE TABLE employees ( employee\_id NUMBER PRIMARY KEY, employee\_name VARCHAR2(100) NOT NULL, salary NUMBER, department\_id NUMBER, CONSTRAINT fk\_department FOREIGN KEY (department\_id) REFERENCES departments(department\_id) );

INSERT INTO departments (department\_id, department\_name) VALUES (1, 'Human Resources');

INSERT INTO departments (department\_id, department\_name) VALUES (2, 'Finance'); INSERT INTO departments (department\_id, department\_name) VALUES (3, 'IT');

INSERT INTO departments (department\_id, department\_name) VALUES (4, 'Marketing'); INSERT INTO departments (department\_id, department\_name) VALUES (5, 'Sales');

select \* from departments;



INSERT INTO employees VALUES (1, 'Harsha', 60000, 1);

INSERT INTO employees VALUES (2, ‘Vishal’, 75000, 1);

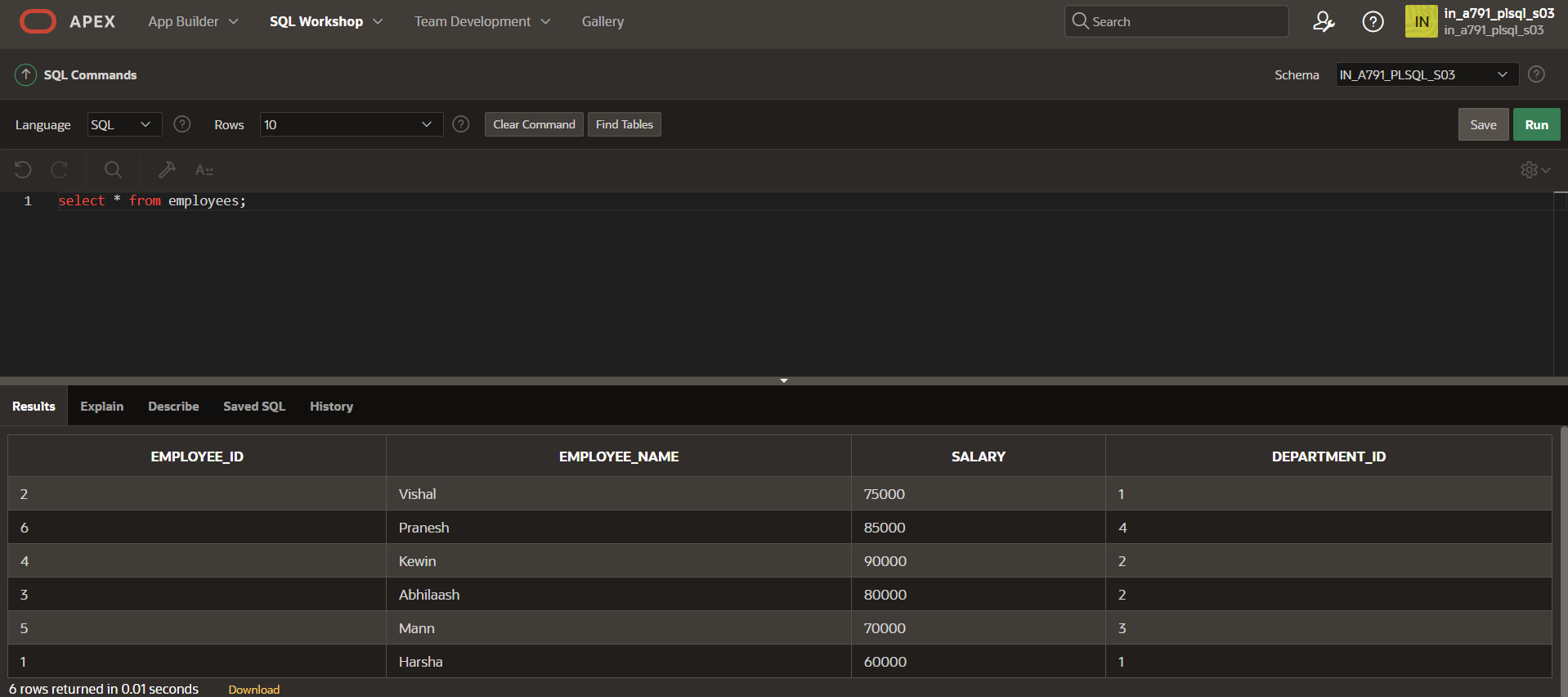
INSERT INTO employees VALUES (3, 'Abhilaash', 80000, 2);

INSERT INTO employees VALUES (4, 'Kewin', 90000, 2);

INSERT INTO employees VALUES (5, 'Mann', 70000, 3);

INSERT INTO employees VALUES (6, 'Pranesh', 85000, 4);

Select \* from employees;



CET

CREATE TABLE categories (

category\_id int PRIMARY KEY,

category\_name VARCHAR2(100) NOT NULL,

parent\_id int);

INSERT INTO categories (category\_id, category\_name, parent\_id) VALUES (1, 'Electronics', NULL);

INSERT INTO categories (category\_id, category\_name, parent\_id) VALUES (2, 'Computers', 1);

INSERT INTO categories (category\_id, category\_name, parent\_id) VALUES (3, 'Laptops', 2);

INSERT INTO categories (category\_id, category\_name, parent\_id) VALUES (4, 'Desktops', 2);

INSERT INTO categories (category\_id, category\_name, parent\_id) VALUES (5, 'Smartphones', 1);

INSERT INTO categories (category\_id, category\_name, parent\_id) VALUES (6, 'Cameras', 1);

INSERT INTO categories (category\_id, category\_name, parent\_id) VALUES (7, 'Digital Cameras', 6);

INSERT INTO categories (category\_id, category\_name, parent\_id) VALUES (8, 'DSLR', 7);

WITH CategoryHierarchy (category\_id, category\_name, parent\_id, full\_path) AS ( SELECT category\_id,category\_name,parent\_id,category\_name AS full\_path FROM categories WHERE parent\_id IS NULL UNION ALL SELECT c.category\_id, c.category\_name, c.parent\_id, ch.full\_path || ' > ' || c.category\_name FROM categories c JOIN CategoryHierarchy ch ON c.parent\_id = ch.category\_id) SELECT category\_id,category\_name,full\_path FROM CategoryHierarchy ORDER BY full\_path;

