

# Funktion, exempel 1 - SQL

```
CREATE FUNCTION proj_hours(INT)
    RETURNS BIGINT AS $$
    SELECT SUM(hours)
    FROM employee_project
    WHERE pno = $1;
$$ LANGUAGE SQL;
```

Anropa funktionen:

```
SELECT proj_hours(2)
```

# Funktion, exempel 2 - SQL

```
CREATE FUNCTION add_project(VARCHAR(30), DATE, DATE)  
    RETURNS BIGINT AS $$
```

```
    INSERT INTO project(pno, name, startdate, enddate)  
    VALUES (nextval('project_pno_seq'), $1, $2, $3);  
    SELECT currval('project_pno_seq');
```

```
$$ LANGUAGE SQL;
```

**Anropa funktionen:**

```
SELECT add_project2('Android', CURRENT_DATE, NULL);
```

# Funktion, exempel 3 – PL/PGSQL

```
CREATE FUNCTION add_project2(VARCHAR(30), DATE, DATE)
    RETURNS BIGINT AS $$

    DECLARE
        seq_no BIGINT;
    BEGIN
        INSERT INTO project (pno,name,startdate,enddate)
        VALUES (nextval('project_pno_seq'), $1, $2, $3);
        SELECT currval('project_pno_seq') INTO seq_no;
        RETURN seq_no;
    END;

$$ LANGUAGE PLPGSQL;
```

# Funktion, exempel 4 - PL/PGSQL

```
CREATE FUNCTION get_manager(INT) RETURNS employee AS $$
DECLARE
    emp employee%ROWTYPE;
    mng department.mno%TYPE;
    dept_no department.dno%TYPE;
BEGIN
    SELECT dno INTO dept_no FROM department WHERE dno = $1;
    IF NOT FOUND THEN
        RAISE EXCEPTION 'No such department ''%''.', $1;
    END IF;
    SELECT mno INTO mng FROM department WHERE dno = $1;
    IF mng IS NULL THEN
        RAISE EXCEPTION 'Department ''%'' has no manager.', $1;
    ELSE
        SELECT e.* INTO emp FROM employee e, department d
        WHERE e.eno = d.mno AND d.dno = $1;
        RETURN emp;
    END IF;
END;
$$ LANGUAGE PLPGSQL;
```