

5 Stored Procedures

1. Procedure: Discharge a Patient

```
CREATE OR REPLACE PROCEDURE discharge_patient(p_patient_id INT)
LANGUAGE plpgsql
AS $$
BEGIN
    UPDATE INPATIENT
    SET discharge_date = CURRENT_DATE
    WHERE patient_id = p_patient_id AND discharge_date IS NULL;

    INSERT INTO BILL (patient_id, bill_date, doctor_fee, room_charge,
test_charge, total_amount, doctor_id)
    SELECT p_patient_id, CURRENT_DATE, 1000, 0, 0, 1000, pd.doctor_id
    FROM PATIENT_DOCTOR pd
    WHERE pd.patient_id = p_patient_id
    LIMIT 1;

    COMMIT;
END;
$$;
```

2. Procedure: Assign Nurse to Ward

```
CREATE OR REPLACE PROCEDURE assign_nurse_to_ward(p_nurse_id INT,
p_ward_id INT)
LANGUAGE plpgsql
AS $$
BEGIN
    INSERT INTO WARD_NURSE (nurse_id, ward_id, assignment_date)
    VALUES (p_nurse_id, p_ward_id, CURRENT_DATE);
END;
$$;
```

3. Procedure: Admit Patient to Ward

```
CREATE OR REPLACE PROCEDURE admit_patient(p_patient_id INT, p_ward_id
INT)
LANGUAGE plpgsql
AS $$
BEGIN
    INSERT INTO INPATIENT (patient_id, ward_id, admission_date)
    VALUES (p_patient_id, p_ward_id, CURRENT_DATE);
END;
$$;
```

4. Procedure: Schedule Next Appointment for Outpatient

```
CREATE OR REPLACE PROCEDURE schedule_next_appointment(p_patient_id
INT, p_next_date DATE)
LANGUAGE plpgsql
AS $$
BEGIN
    UPDATE OUTPATIENT
    SET next_appointment_date = p_next_date
    WHERE patient_id = p_patient_id;
END;
$$;
```

5. Procedure: Update Patient Disease Info

```
CREATE OR REPLACE PROCEDURE update_disease(p_patient_id INT, p_disease
TEXT)
LANGUAGE plpgsql
AS $$
BEGIN
    UPDATE PATIENT
    SET disease = p_disease
    WHERE patient_id = p_patient_id;
END;
$$;
```