202003019

## **Functions**

```
1. prescription of a patient id = x
CREATE OR REPLACE FUNCTION prescriptions_patientid(X bigint)
RETURNS SETOF prescription AS $$
DECLARE
     p prescription%rowtype;
BEGIN
     FOR p In Select * FROM prescription WHERE patient_id = X
     Loop
          RETURN NEXT p;
     End Loop;
RETURN;
END;
$$ LANGUAGE plpgsql;
```

## SELECT \* FROM prescriptions\_patientid(926722993945);

าลา	a Output Explair	n Messages I	Notifications					
/u	notiont id	doctor id		from_date	o to_date	morning_dose	noon_dose	night_dose
4	numeric (12)	numeric (12)	integer	date	△ date △	character varying (10)	character varying (10)	character varying (10)
1	926722993945	131407887378		2020-12-04	2020-12-22	7	6	3
2	926722993945	986786537816	4	2020-12-04	2020-12-10	0	9	7
3	926722993945	540258634201	8	2020-12-10	2020-12-20	7	3	0
4	926722993945	131407887378		2020-12-04	2020-12-12	7	2	6

## 2. lab reports of patient id = x

```
CREATE OR REPLACE FUNCTION reports_patientid(X bigint)

RETURNS SETOF lab_reports AS $$

DECLARE

r lab_reports%rowtype;

BEGIN

FOR r In SELECT * FROM lab_reports WHERE patient_id = X

Loop

RETURN NEXT r;

End Loop;

RETURN;

END;

$$ LANGUAGE plpgsql;
```

## SELECT \* FROM reports patientid(983473196869)



3. prescription of a patient id = x, prescribed by doctor\_id = y

CREATE OR REPLACE FUNCTION prescription\_patientid\_doctorid(X bigint, Y bigint)

RETURNS SETOF prescription AS \$\$

#### **DECLARE**

p prescription%rowtype;

#### **BEGIN**

For p in select \* from prescription where patient\_id = X and doctor\_id = Y

Loop

RETURN NEXT p;

End Loop;

## RETURN;

```
END;
```

\$\$ LANGUAGE plpgsql;

## **SELECT \* FROM**

prescription\_patientid\_doctorid(107998063737,596818320348);



4. each operation done by doctor with doctorid= X (function)

CREATE OR REPLACE FUNCTION operations\_doctorid(X bigint)

RETURNS SET OF operation AS \$\$

**DECLARE** 

o operation%rowtype;

#### **BEGIN**

For o in SELECT \* FROM operation WHERE doctor\_id = X

Loop

RETURN NEXT o;

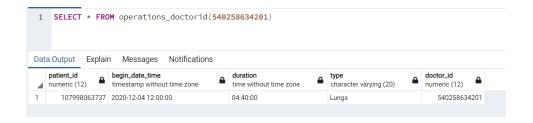
End Loop;

RETURN;

```
END;
```

\$\$ LANGUAGE plpgsql;

SELECT \* FROM operations doctorid(540258634201);



5. operation of a patient with patient\_id = X

CREATE OR REPLACE FUNCTION operations\_patientid(X bigint)
RETURNS SETOF operation AS \$\$

**DECLARE** 

o operation%rowtype;

**BEGIN** 

For o in SELECT \* FROM operation WHERE patient\_id = X Loop

RETURN NEXT o;

End Loop;

RETURN;

## END;

## \$\$ LANGUAGE plpgsql;

# SELECT \* FROM operation\_patientid(713964268158);

1 Dat	SELECT * FRO	om operations_patientid(	713	3964268158)		
4	patient_id numeric (12)	hegin date time	<u></u>	duration time without time zone	type character varying (20)	doctor_id numeric (12)
1	713964268158	2021-04-07 09:00:00		05:24:00	Cancer	657596761672
2	713964268158	2021-04-07 09:00:00		05:24:00	Cancer	131407887378
3	713964268158	2021-04-07 09:00:00		05:24:00	Cancer	220195004338

# 6. total unpaid bill of a patient\_id = X

CREATE OR REPLACE FUNCTION unpaid\_patientid(X bigint)

RETURNS int AS \$\$

RETURN unpaid;

**DECLARE** 

unpaid int;

#### **BEGIN**

```
Select Sum(medicine_charges) + Sum(operation_charges) + Sum(blood_t_charges) + Sum(lab_charges) + Sum(service_charges) into unpaid From bill where patient_id = X and status = false;
```

## END;

\$\$ LANGUAGE plpgsql;

SELECT \* FROM unpaid\_patientid(618290147720);

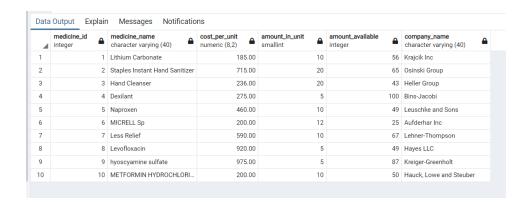


## **Views**

1. available medicines (view for pharmacist)

CREATE VIEW medicines\_available AS

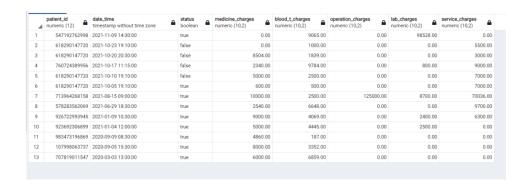
SELECT \* FROM medicines WHERE amount\_available>'0';



2. bill view (each and every bill) (view for receptionist)

### **CREATE VIEW bills AS**

SELECT \* FROM bill ORDER BY date\_time DESC;



3. can get patient list for every room (view for nurses)

CREATE VIEW rooms\_patientids AS

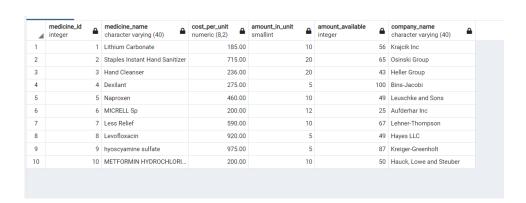
# SELECT \* FROM admitted\_patients\_ids ORDER BY room\_no, date\_of\_admit DESC;

	room_no integer      ▲	date_of_admit date	patient_id numeric (12)
1	1	2020-12-04	926722993945
2	2	2021-07-08	713964268158
3	3	2021-06-01	578283562069
4	5	2021-10-06	760724389956

## 4. medicine view (view for pharmacist)

## CREATE VIEW medicine\_detail AS

## SELECT \* FROM medicines ORDER BY medicine\_id;



5. prescription view (excluding doctor and patient details)(view for pharmacist)

**CREATE VIEW prescriptions medicines AS** 

SELECT medicine\_id, from\_date, to\_date, morning\_dose, noon\_dose, night\_dose FROM prescription ORDER BY medicine\_id, to\_date DESC;



6. can get patient list for every room (view for nurses)

CREATE VIEW rooms\_patientids AS

SELECT \* FROM admitted\_patients\_ids ORDER BY room\_no,
date\_of\_admit DESC;

4	room_no integer	date_of_admit date	patient_id numeric (12)	
1	1	2020-12-04	926722993945	
2	2	2021-07-08	713964268158	
3	3	2021-06-01	578283562069	
4	5	2021-10-06	760724389956	