II2250 Manajemen Basis Data
Semester II 2022/2023

Praktikum III: Functions and Stored Procedure

Dipersiapkan oleh:
Asisten Laboratorium Basis Data

Pelaksanaan:
Waktu Mulai: Senin, 13 Maret 2023 – 16.15 WIB
Waktu Selesai: Senin, 13 Maret 2023 – 18.00 WIB

I. Ketentuan

Peserta kuliah akan mengerjakan praktikum mengenai **Functions and Stored Procedure** pada DBMS PostgreSQL menggunakan data hospital.sql yang tersedia di folder praktikum ini. Praktikum ini dikerjakan secara **INDIVIDU**.

Peserta dapat memanfaatkan file yang terdapat pada folder Untuk Peserta dan dokumentasi PostgreSQL dalam pengerjaan soal praktikum.

Deliverables yang harus dikumpulkan untuk Praktikum ini adalah sebagai berikut.

1. File .docx berisikan daftar query beserta screenshot eksekusi tiap query dengan format nama: **K<Kelas>_P03_<NIM>.docx**
   Contoh: K01_P03_18221000.docx

Isi dari dokumen adalah sebagai berikut.

a. Untuk setiap soal.
   i. Perintah atau langkah yang dilakukan untuk menyelesaikan persoalan. Sediakan semua kode dalam bentuk **teks**.
   ii. Query pengecekan dalam bentuk **teks** (jika diminta pada soal).
   iii. Hasil perintah dari *query* tersebut

Deliverables dikumpulkan pada pranala berikut.

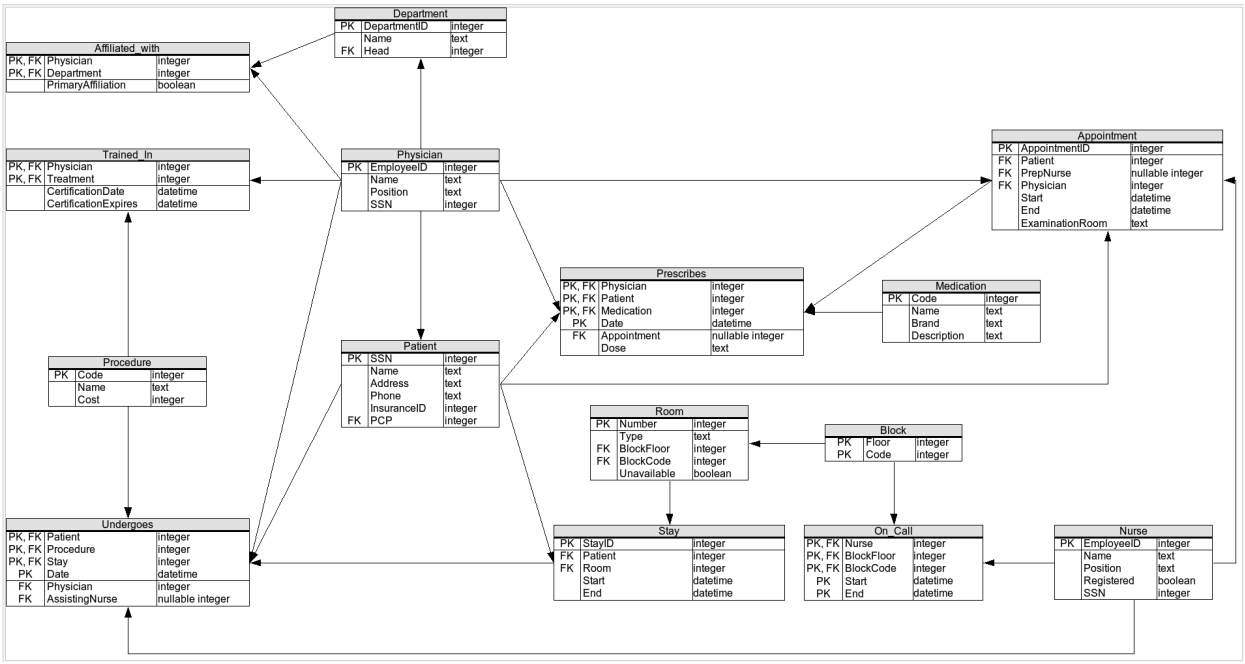
https://s.id/uploader-prak3

Keterlambatan pengumpulan *deliverables* akan mengakibatkan pengurangan nilai. Segala tindak kecurangan akan ditindaklanjuti dan diikuti konsekuensi serius.

~ Selamat mengerjakan! :)) ~

II. Skema Basis Data

Berikut ini adalah skema dari basis data Rumah Sakit Basdat dari file hospital.sql.



Affiliated_with = (Physician, Department, PrimaryAffiliation)

Appointment = (AppointmentID, Patient, PrepNurse, Physician, StartDate, EndDate, ExaminationRoom)

Block = (BlockFloor, BlockCode)

Department = (DepartmentID, Name, Head)

Medication = (Code, Name, Brand, Description)

Nurse = (EmployeeID, Name, Position, Registered, SSN)

On_call = (Nurse, BlockFloor, BlockCode, OnCallStart, OnCallEnd)

Patient = (SSN, Name, Address, Phone, InsuranceID, PCP)

Physician = (EmployeeID, Name, Position, SSN)

Prescribes = (Physician, Patient, Medication, Date, Appointment, Dose)

Procedures = (Code, Name, Cost)

Room = (RoomNumber, RoomType, BlockFloor, BlockCode, Unavailable)

Stay = (StayID, Patient, Room, StayStart, StayEnd)

Trained_in = (Physician, Treatment, CertificationDate, CertificationExpires)

Undergoes = (Patient, Procedures, Stay, DateUndergoes, Physician, AssistingNurse)

III. Soal

(**Note**: Pastikan bahwa pada komputer yang digunakan, telah terdapat sebuah database bernama hospital. Jika belum, buatlah sebuah database bernama **hospital** dan import hospital.sql ke dalam database tersebut!)

1. Buatlah sebuah database bernama **rumahsakit** yang merupakan duplikat dari database **hospital**.
2. Rumah Sakit Basdat memiliki sistem penggajian dimana setiap physician akan mendapatkan pendapatan 20% dari setiap pendapatan prosedur pengobatan yang pernah ia lakukan.

   a. Buatlah sebuah **function** untuk menghitung jumlah pendapatan dari masing-masing physician dari seluruh prosedur pengobatan yang pernah ia lakukan dengan menerima **argumen berupa id physician**.

   Hint : Gunakan coalesce atau isnull untuk memberikan default value jika mereturn null
   Contoh penggunaan select coalesce(NULL,1) akan mereturn 1

   b. Lalu buatlah query untuk menampilkan id, nama, posisi, dan jumlah pendapatan dari **seluruh** physician yang bekerja di rumah sakit.

   Jawaban:

Query Pembuatan Function	`CREATE OR REPLACE FUNCTION total_income(physician_id INT)`
	`  RETURNS REAL`
	`  LANGUAGE plpgsql`
	`AS $$`
	`DECLARE`
	`  total REAL;`
	`BEGIN`
	`  SELECT COALESCE(SUM(0.2 * cost), 0)`
	`  INTO total`
	`  FROM procedures`
	`  WHERE procedures.code = physician_id;`
	
	`  RETURN total;`
	`END;`
	`$$;`

SS Query Pembuatan Function

```
rumahsakit=# CREATE OR REPLACE FUNCTION total_income(physician_id INT)
rumahsakit-#     RETURNS REAL
rumahsakit-#     LANGUAGE plpgsql
rumahsakit-# AS $$
rumahsakit$# DECLARE
rumahsakit$#     total REAL;
rumahsakit$# BEGIN
rumahsakit$#     SELECT COALESCE(SUM(0.2 * cost), 0)
rumahsakit$#     INTO total
rumahsakit$#     FROM procedures
rumahsakit$#     WHERE procedures.code = physician_id;
rumahsakit$#
rumahsakit$#     RETURN total;
rumahsakit$# END;
rumahsakit$# $$;
CREATE FUNCTION
rumahsakit=#
```

Query Penerapan Function

```
SELECT employeeid, name, position,
total_income(employeeid)
FROM physician;
```

SS Hasil Penerapan Function

*boleh SS awal dan akhir row saja jika terlalu panjang

```
rumahsakit=# SELECT employeeid, name, position, total_income(employeeid)
FROM physician;
 employeeid |        name         |            position            | total_income
------------+---------------------+--------------------------------+--------------
          1 | John Dorian         | Staff Internist                |          300
          2 | Elliot Reid         | Attending Physician            |          750
          3 | Christopher Turk    | Surgical Attending Physician   |          900
          4 | Percival Cox        | Senior Attending Physician     |         2000
          5 | Bob Kelso           | Head Chief of Medicine         |        979.8
          6 | Todd Quinlan        | Surgical Attending Physician   |         1120
          7 | John Wen            | Surgical Attending Physician   |            5
          8 | Keith Dudemeister   | MD Resident                    |            0
          9 | Molly Clock         | Attending Psychiatrist         |            0
         10 | Immanuel Gallear    | Nurse                          |            0
         11 | Joshua Geal         | Biostatistician II             |            0
         12 | Kiley Mauser        | Software Engineer II           |            0
         13 | Maxy Sydney         | Senior Financial Analyst       |            0
         14 | Emma Fairfoul       | Administrative Assistant III   |            0
         15 | Dallas Blevin       | Account Representative I       |            0
         16 | Ronny Rulten        | Staff Scientist                |            0
         17 | Karlene Douse       | Accounting Assistant II        |            0
         18 | Baryram Peattie     | Research Associate             |            0
         19 | Judy Copeland       | Administrative Assistant II    |            0
         20 | Hurleigh Croyser    | VP Marketing                   |            0
         21 | Bryan Wigley        | Software Consultant            |            0
         22 | Vasilis Andrioletti | Geological Engineer            |            0
         23 | Marylee Stainer     | Help Desk Operator             |            0
         24 | Gracie Gabrieli     | GIS Technical Architect        |            0
         25 | Shara Girodon       | Help Desk Operator             |            0
         26 | Starr Penddreth     | Cost Accountant                |            0
         27 | Beauregard Bramsom  | Graphic Designer               |            0
         28 | Micki Lapham        | Payment Adjustment Coordinator |            0
         29 | Arri Imlaw          | Sales Representative           |            0
         30 | Beaufort Palumbo    | Environmental Tech             |            0
         31 | Olvan Kluger        | Analyst Programmer             |            0
```

```
 66 | Tabb Fogden          | Compensation Analyst           |          0
 67 | Dore Biesterfeld     | Speech Pathologist             |          0
 68 | Faye Wadeling        | Sales Associate                |          0
 69 | Rodrick Crabtree     | Internal Auditor               |          0
 70 | Jaymee Markussen     | Staff Accountant IV            |          0
 71 | Maddi Giacomelli     | Senior Cost Accountant         |          0
 72 | Bree Leathem         | Sales Representative           |          0
 73 | Chrisy Stanluck      | Legal Assistant                |          0
 74 | Steven Coppard       | Dental Hygienist               |          0
 75 | Kore Danielian       | Cost Accountant                |          0
 76 | Flint Frichley       | Account Coordinator            |          0
 77 | Hesther Eyam         | Marketing Manager              |          0
 78 | Helenka Rattenbury   | Social Worker                  |          0
 79 | Gert Swaden          | Occupational Therapist         |          0
 80 | Ninetta Andres       | Research Assistant II          |          0
 81 | Baryram Gayton       | Information Systems Manager    |          0
 82 | Stan Human           | Sales Representative           |          0
 83 | Hurlee Mathevet      | Pharmacist                     |          0
 84 | Conni Lehrle         | Structural Analysis Engineer   |          0
 85 | Mitchel Kenny        | Staff Accountant II            |          0
 86 | Natalina Overton     | Internal Auditor               |          0
 87 | Wilmar Hallagan      | Environmental Tech             |          0
 88 | Ives Grieveson       | Paralegal                      |          0
 89 | Quintina Bramble     | Graphic Designer               |          0
 90 | Farr Cornfoot        | Desktop Support Technician     |          0
 91 | Joni Hummerston      | Payment Adjustment Coordinator |          0
 92 | Dix Cawthron         | Help Desk Technician           |          0
 93 | Jori Snellman        | Quality Engineer               |          0
 94 | Des Sweedy           | Payment Adjustment Coordinator |          0
 95 | Yoko Anthill         | Safety Technician I            |          0
 96 | Alva Goodlad         | Mechanical Systems Engineer    |          0
 97 | Drew Oglevie         | Product Engineer               |          0
 98 | Casey Gallehock      | Accountant IV                  |          0
 99 | Astrix Gorch         | Financial Analyst              |          0
100 | Aldous Paroni        | Senior Developer               |          0
(100 rows)

rumahsakit=#
```

3. Calon pasien Rumah Sakit Basdat, masih menggunakan cara manual untuk memeriksa ketersediaan jadwal physician yang diinginkan dengan menemuinya secara langsung. Untuk memudahkan calon pasien, akhir Rumah Sakit Basdat menawarkan sistem baru dengan *query* ke basis data.

   Jadwal ketersediaan (availability) seorang physician dilihat dari ada atau tidaknya appointment pada range tanggal yang diberikan

   .

   a. Buatlah sebuah function untuk memeriksa jadwal ketersediaan dari physician Rumah Sakit Basdat yang mengembalikan true jika available dan false jika not available. dengan interface sebagai berikut:

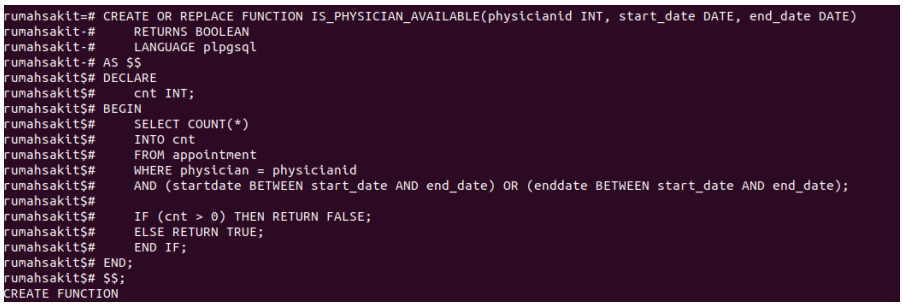
   ```
   function IS_PHYSICIAN_AVAILABLE (physician: INTEGER,
   startDate: DATE, endDate: DATE) -> boolean
   ```

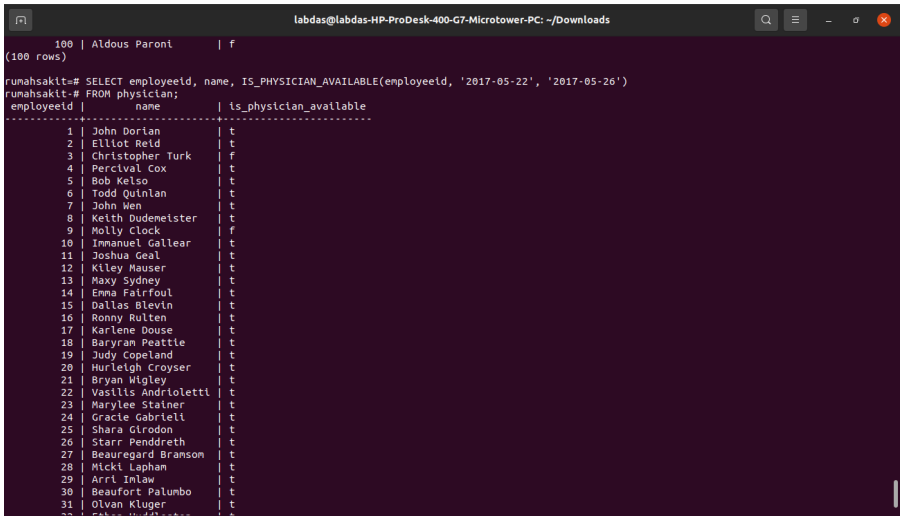
   b. Berikan contoh bebas penggunaan untuk menampilkan physician yang *available* dan yang *tidak available* masing-masing satu. Dengan format: physician_id, nama, *availability*

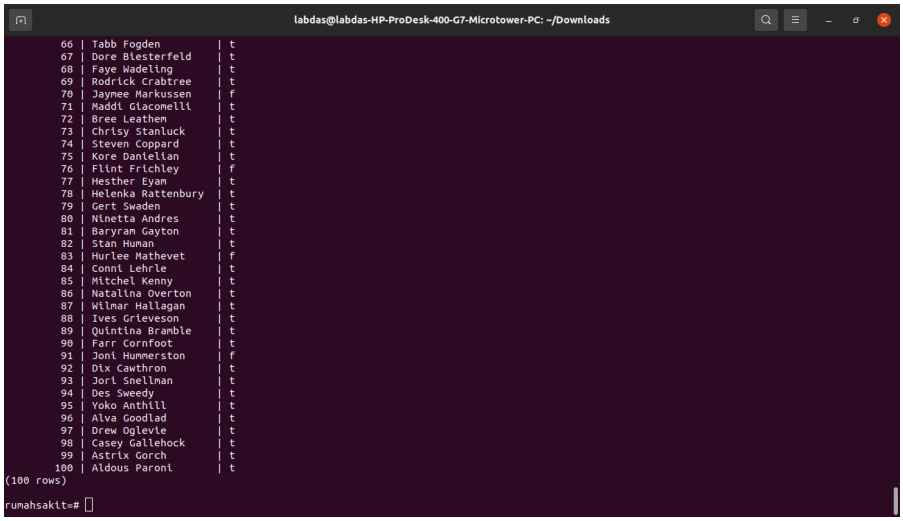
   c. Lalu gunakan function tersebut untuk membuat query yang menampilkan data id dan nama-nama *physician* yang sedang *available* dari tanggal 2017-04-22 hingga 2017-04-23.

Jawaban:

a) Query Pembuatan Function	CREATE OR REPLACE FUNCTION IS_PHYSICIAN_AVAILABLE(physicianid INT, start_date DATE, end_date DATE)
	RETURNS BOOLEAN
	LANGUAGE plpgsql
	AS $$
	DECLARE
	cnt INT;
	BEGIN
	SELECT COUNT(*)
	INTO cnt
	FROM appointment
	WHERE physician = physicianid
	AND (startdate BETWEEN start_date AND end_date) OR (enddate BETWEEN start_date AND end_date);
	
	IF (cnt > 0) THEN RETURN FALSE;
	ELSE RETURN TRUE;
	END IF;
	END;
	$$;

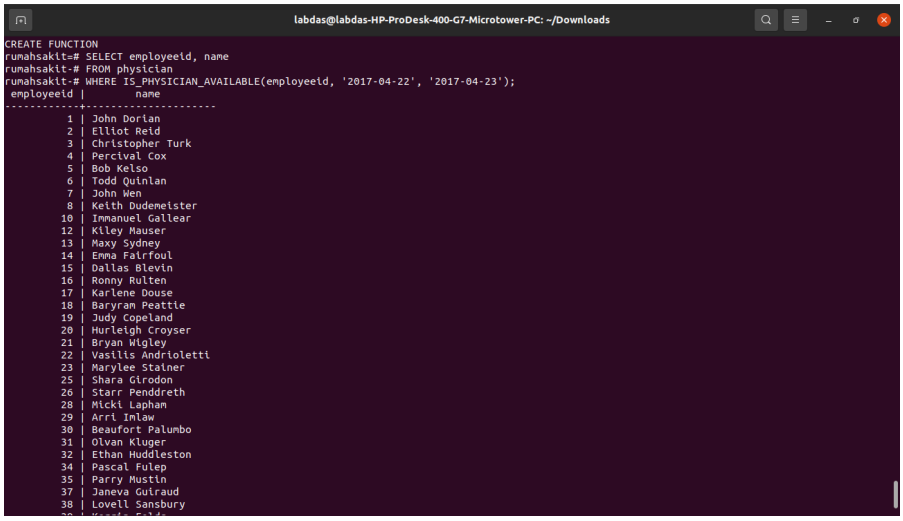
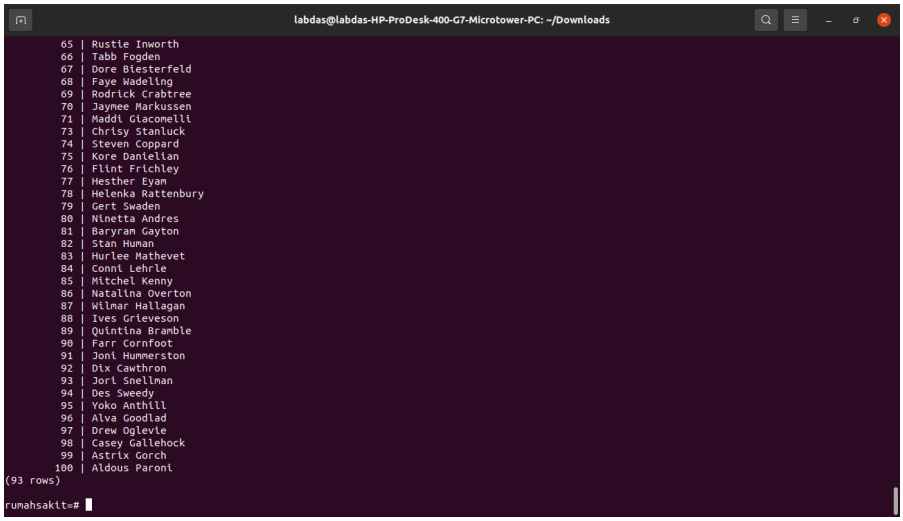
a) SS Query Pembuatan Function	 rumahsakit=# CREATE OR REPLACE FUNCTION IS_PHYSICIAN_AVAILABLE(physicianid INT, start_date DATE, end_date DATE) rumahsakit-# RETURNS BOOLEAN rumahsakit-# LANGUAGE plpgsql rumahsakit-# AS $$ rumahsakit$# DECLARE rumahsakit$# cnt INT; rumahsakit$# BEGIN rumahsakit$# SELECT COUNT(*) rumahsakit$# INTO cnt rumahsakit$# FROM appointment rumahsakit$# WHERE physician = physicianid rumahsakit$# AND (startdate BETWEEN start_date AND end_date) OR (enddate BETWEEN start_date AND end_date); rumahsakit$# rumahsakit$# IF (cnt > 0) THEN RETURN FALSE; rumahsakit$# ELSE RETURN TRUE; rumahsakit$# END IF; rumahsakit$# END; rumahsakit$# $$; CREATE FUNCTION
b) Query dan Screenshot Hasil	SELECT employeeid, name, IS_PHYSICIAN_AVAILABLE(employeeid, '2017-05-22', '2017-05-26') FROM physician;





c) Query Penerapan

```
SELECT employeeid, name
FROM physician
WHERE IS_PHYSICIAN_AVAILABLE(employeeid,
'2017-04-22', '2017-04-23');
```

c) SS Hasil Penerapan Function	 

4. Rumah sakit Basdat bekerja sama dengan pihak asuransi untuk mengetahui seberapa sering pasien mengunjungi rumah sakit untuk menentukan harga premi asuransi yang dibebankan pada pasien. Oleh karena itu,

    A. Buatlah fungsi untuk menghitung jumlah total pasien berkunjung ke rumah sakit basdat dengan menerima argumen berupa nama pasien dan tahun periode kunjungan.

    B. Setelah itu buatlah kueri dengan memanfaatkan fungsi tersebut untuk menampilkan nama pasien, jumlah kunjungan, dan tahun periode kunjungan data diurutkan berdasarkan nama pasien dan tahun secara ascending, tahun kunjungan yang dibutuhkan adalah dari 2015-2019 . **HINT Dapat menggunakan UNION.**
    `extract(year FROM table_name) untuk mendapatkan tahun`

Contoh luaran:

name	total_visit	year
Asisten	0	2015
Asisten	0	2016
Asisten	0	2017
Asisten	0	2018
Asisten	0	2019
Basdat	0	2015
Basdat	0	2016
Basdat	0	2017
Basdat	0	2018
Basdat	0	2019

Query Pembuatan Function	CREATE OR REPLACE FUNCTION total_pasien(nama_pasien VARCHAR(30), year INT)
	RETURNS INT
	LANGUAGE plpgsql
	AS $$
	DECLARE
	total INT;
	BEGIN
	SELECT COUNT(*)
	INTO total
	FROM appointment
	JOIN patient
	ON patient.ssn = appointment.patient
	WHERE patient.name = nama_pasien
	AND EXTRACT(YEAR FROM startdate) BETWEEN 2015 AND 2019;
	
	RETURN total;
	END;
	$$;

Query Penerapan Function	SELECT name, total_pasien(name, CAST(EXTRACT(YEAR FROM startdate) AS INT)) total_visit, EXTRACT(YEAR FROM startdate) AS year FROM patient JOIN appointment ON patient.ssn = appointment.patient ORDER BY name, EXTRACT(YEAR FROM startdate);

SS Query Pembuatan Function

```
rumahsakit=# CREATE OR REPLACE FUNCTION total_pasien(nama_pasien VARCHAR(30), year INT)
rumahsakit-#     RETURNS INT
rumahsakit-#     LANGUAGE plpgsql
rumahsakit-# AS $$
rumahsakit$# DECLARE
rumahsakit$#     total INT;
rumahsakit$# BEGIN
rumahsakit$#     SELECT COUNT(*)
rumahsakit$#     INTO total
rumahsakit$#     FROM appointment
rumahsakit$#     JOIN patient
rumahsakit$#     ON patient.ssn = appointment.patient
rumahsakit$#     WHERE patient.name = nama_pasien
rumahsakit$#     AND EXTRACT(YEAR FROM startdate) BETWEEN 2015 AND 2019;
rumahsakit$#
rumahsakit$#     RETURN total;
rumahsakit$# END;
rumahsakit$# $$;
CREATE FUNCTION
```

SS Hasil Penerapan Function

*boleh SS awal dan akhir row saja jika terlalu panjang

```
rumahsakit=# SELECT name, total_pasien(name, CAST(EXTRACT(YEAR FROM startdate) AS INT)) total_visit, EXTRACT(YEAR FROM startdate) AS year
rumahsakit-# FROM patient
rumahsakit-# JOIN appointment
rumahsakit-# ON patient.ssn = appointment.patient
rumahsakit-# ORDER BY name, EXTRACT(YEAR FROM startdate);
          name              | total_visit | year
----------------------------+-------------+------
 Abra Churchward            |           1 | 2018
 Abrahan Knightsbridge      |           2 | 2017
 Abrahan Knightsbridge      |           2 | 2017
 Adan Arbor                 |           3 | 2017
 Adan Arbor                 |           3 | 2018
 Adan Arbor                 |           3 | 2018
 Adiana Ellph               |           2 | 2017
 Adiana Ellph               |           2 | 2018
 Adore Ganiford             |           1 | 2018
 Adrianna Dovey             |           1 | 2017
 Agace Andrivel             |           1 | 2017
 Agathe Rowe                |           1 | 2017
 Ailsun Bebbell             |           1 | 2017
 Ainslee Cantua             |           2 | 2017
 Ainslee Cantua             |           2 | 2017
 Alano Philippon            |           2 | 2017
 Alano Philippon            |           2 | 2018
 Alasteir Brunt             |           2 | 2017
 Alasteir Brunt             |           2 | 2018
 Alejandro Baldoni          |           1 | 2017
 Alfy Clemendet             |           1 | 2018
 Algernon MacKim            |           1 | 2017
 Alicia Gower               |           2 | 2017
 Alicia Gower               |           2 | 2018
 Alick Waplington           |           3 | 2017
 Alick Waplington           |           3 | 2018
 Alick Waplington           |           3 | 2018
 Almira Pantone             |           4 | 2017
 Almira Pantone             |           4 | 2017
```

```
labdas@labdas-HP-ProDesk-400-G7-Microtower-PC: ~/Downloads
 Vasili Dagger              |           4 | 2018
 Vaughn Ramsell             |           2 | 2017
 Vaughn Ramsell             |           2 | 2017
 Verile Courtenay           |           3 | 2017
 Verile Courtenay           |           3 | 2017
 Verile Courtenay           |           3 | 2018
 Vernor Anglim              |           1 | 2017
 Veronique Newcomen         |           2 | 2017
 Veronique Newcomen         |           2 | 2018
 Vin Reolfo                 |           2 | 2017
 Vin Reolfo                 |           2 | 2017
 Viola Grange               |           1 | 2018
 Virgie Engledow            |           1 | 2017
 Vito Stigger               |           2 | 2017
 Vito Stigger               |           2 | 2018
 Vivien Triggs              |           3 | 2017
 Vivien Triggs              |           3 | 2018
 Vivien Triggs              |           3 | 2018
 Weston Degue               |           3 | 2017
 Weston Degue               |           3 | 2017
 Weston Degue               |           3 | 2017
 West Probin                |           1 | 2018
 Worthington Schofield      |           1 | 2017
 Yanaton Fairburne          |           1 | 2017
 Yovonnda Alenichicov       |           2 | 2018
 Yovonnda Alenichicov       |           2 | 2018
 Ysabel Paolinelli          |           2 | 2017
 Ysabel Paolinelli          |           2 | 2018
 Zaria Andrzejak            |           2 | 2017
 Zaria Andrzejak            |           2 | 2018
 Zaria Poskitt              |           1 | 2017
 Zollie Kuller              |           2 | 2017
 Zollie Kuller              |           2 | 2018
 Zonda Casaro               |           2 | 2018
 Zonda Casaro               |           2 | 2018
(1000 rows)

rumahsakit=#
```

5. Sebagai rumah sakit yang ingin memberikan pelayanan terbaik bagi para pasien, Rumah Sakit Basdat memastikan bahwa seluruh tenaga kerja medisnya memiliki keterampilan dan sertifikasi pelatihan yang dibutuhkan.

A. Buatlah sebuah prosedur yang akan menghapus data sertifikasi dari physician pada relasi trained_in dengan parameter berupa expired_date yang akan menghapus data sertifikasi yang expired sebelum expired_date (expired_date memiliki tipe data DATE) .
B. Lakukan panggilan pada prosedur tersebut untuk expired_date 11 Mei 2019!
C. Buat query untuk menampilkan data dari relasi trained_in dengan *certificationexpires* sebelum 11 Mei 2019 setelah prosedur berhasil dipanggil.

Jawaban:

Query Pembuatan Prosedur	`CREATE OR REPLACE PROCEDURE delete_certification(expired_date DATE)`
	`LANGUAGE SQL`
	`AS $$`
	`  DELETE FROM trained_in`
	`  WHERE certificationexpires < expired_date;`
	`$$;`

SS Query Pembuatan Prosedur	`rumahsakit=# CREATE OR REPLACE PROCEDURE delete_certification(expired_date DATE)`
	`rumahsakit-# LANGUAGE SQL`
	`rumahsakit-# AS $$`
	`rumahsakit$#  DELETE FROM trained_in`
	`rumahsakit$#  WHERE certificationexpires < expired_date;`
	`rumahsakit$# $$;`
	`CREATE PROCEDURE`
	`rumahsakit=#`

Query Pemanggilan Prosedur	`CALL delete_certification('2019-05-11');`

Query untuk Menampilkan Data Akhir & SS Tampilan Data	`SELECT *`
	`FROM trained_in`
	`WHERE certificationexpires < '2019-05-11';`
	`rumahsakit=# CALL delete_certification('2019-05-11');`
	`CALL`
	`rumahsakit=# SELECT *`
	`rumahsakit-# FROM trained_in`
	`rumahsakit-# WHERE certificationexpires < '2019-05-11';`
	` physician | treatment | certificationdate | certificationexpires`
	`----------+----------+----------+----------`
	`(0 rows)`
	
	`rumahsakit=#`

6. Karena adanya investigasi dari WHO, pihak Rumah Sakit ingin mengetahui appointment yang sudah selesai atau tidak
    A. Tambahkan atribut isFinished(boolean pada tabel appointment. Selanjutnya dengan anggapan hari ini adalah tanggal 1 april 2018 , buatlah sebuah procedure untuk memberikan value isFinished apakah appointment tersebut sudah selesai (true) atau belum(false)
    B. Lakukan panggilan pada prosedur tersebut!
    C. Buat query untuk menampilkan data dari tabel appointment setelah prosedur berhasil dipanggil.

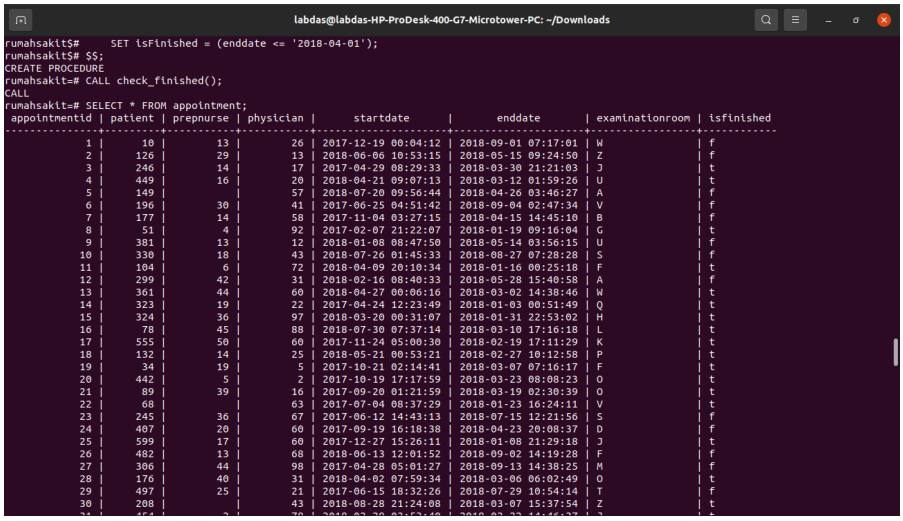
Jawaban:

Query Penambahan kolom	`ALTER TABLE appointment` `ADD COLUMN isFinished BOOLEAN;`

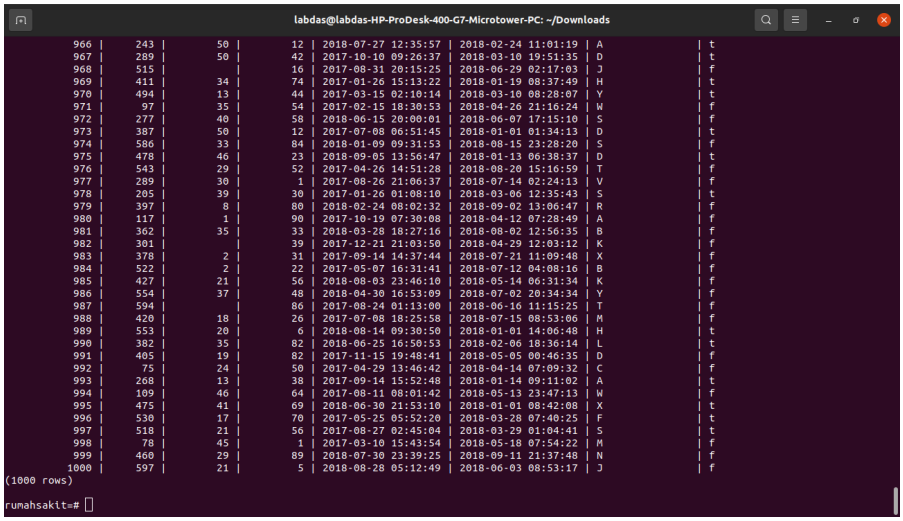
Query Pembuatan Prosedur	`CREATE OR REPLACE PROCEDURE check_finished()`
	`LANGUAGE SQL`
	`AS $$`
	`  UPDATE appointment`
	`  SET isFinished = (enddate <= '2018-04-01');`
	`$$;`

SS Query Pembuatan Prosedur	`rumahsakit=# CREATE OR REPLACE PROCEDURE check_finished()`
	`rumahsakit-# LANGUAGE SQL`
	`rumahsakit-# AS $$`
	`rumahsakit$#  UPDATE appointment`
	`rumahsakit$#  SET isFinished = (enddate <= '2018-04-01');`
	`rumahsakit$# $$;`
	`CREATE PROCEDURE`

Query Pemanggilan Prosedur	`CALL check_finished();`
Query untuk Menampilkan Data & SS	`SELECT * FROM appointment;`



```
labdas@labdas-HP-ProDesk-400-G7-Microtower-PC: ~/Downloads
rumahsakit$#     SET isFinished = (enddate <= '2018-04-01');
rumahsakit$# $$;
CREATE PROCEDURE
rumahsakit=# CALL check_finished();
CALL
rumahsakit=# SELECT * FROM appointment;
 appointmentid | patient | prepnurse | physician |      startdate      |       enddate       | examinationroom | isfinished
---------------+---------+-----------+-----------+---------------------+---------------------+-----------------+------------
             1 |      10 |        13 |        26 | 2017-12-19 00:04:12 | 2018-09-01 07:17:01 | W               | f
             2 |     126 |        29 |        13 | 2018-06-06 10:53:15 | 2018-05-15 09:24:50 | Z               | f
             3 |     246 |        14 |        17 | 2017-04-29 08:29:33 | 2018-03-30 21:21:03 | J               | t
             4 |     449 |        16 |        20 | 2018-04-21 09:07:13 | 2018-03-12 01:59:26 | U               | t
             5 |     149 |           |        57 | 2018-07-20 09:56:44 | 2018-04-26 03:46:27 | A               | f
             6 |     196 |        30 |        41 | 2017-06-25 04:51:42 | 2018-09-04 02:47:34 | V               | f
             7 |     177 |        14 |        58 | 2017-11-04 03:27:15 | 2018-04-15 14:45:10 | B               | f
             8 |      51 |         4 |        92 | 2017-02-07 21:22:07 | 2018-01-19 09:16:04 | G               | t
             9 |     381 |        13 |        12 | 2018-01-08 08:47:50 | 2018-05-14 03:56:15 | U               | f
            10 |     330 |        18 |        43 | 2018-07-26 01:45:33 | 2018-08-27 07:28:28 | S               | f
            11 |     104 |         6 |        72 | 2018-04-09 20:10:34 | 2018-01-16 00:25:18 | F               | t
            12 |     299 |        42 |        31 | 2018-02-16 08:40:33 | 2018-05-28 15:40:58 | A               | f
            13 |     361 |        44 |        60 | 2018-04-27 00:06:16 | 2018-03-02 14:38:46 | W               | t
            14 |     323 |        19 |        22 | 2017-04-24 12:23:49 | 2018-01-03 00:51:49 | Q               | t
            15 |     324 |        36 |        97 | 2018-03-20 00:31:07 | 2018-01-31 22:53:02 | H               | t
            16 |      78 |        45 |        88 | 2018-07-30 07:37:14 | 2018-03-10 17:16:18 | L               | t
            17 |     555 |        50 |        60 | 2017-11-24 05:00:30 | 2018-02-19 17:11:29 | K               | t
            18 |     132 |        14 |        25 | 2018-05-21 00:53:21 | 2018-02-27 10:12:58 | P               | t
            19 |      34 |        19 |         5 | 2017-10-21 02:14:41 | 2018-03-07 07:16:17 | F               | t
            20 |     442 |         5 |         2 | 2017-10-19 17:17:59 | 2018-03-23 08:08:23 | O               | t
            21 |      89 |        39 |        16 | 2017-09-20 01:21:59 | 2018-03-19 02:30:39 | O               | t
            22 |      68 |           |        63 | 2017-07-04 08:37:29 | 2018-01-23 16:24:11 | V               | t
            23 |     245 |        36 |        67 | 2017-06-12 14:43:13 | 2018-07-15 12:21:56 | S               | f
            24 |     407 |        20 |        60 | 2017-09-19 16:18:38 | 2018-04-23 20:08:37 | D               | f
            25 |     599 |        17 |        60 | 2017-12-27 15:26:11 | 2018-01-08 21:29:18 | J               | t
            26 |     482 |        13 |        68 | 2018-06-13 12:01:52 | 2018-09-02 14:19:28 | F               | f
            27 |     306 |        44 |        98 | 2017-04-28 05:01:27 | 2018-09-13 14:38:25 | M               | f
            28 |     176 |        40 |        31 | 2018-04-02 07:59:34 | 2018-03-06 06:02:49 | O               | t
            29 |     497 |        25 |        21 | 2017-06-15 18:32:26 | 2018-07-29 10:54:14 | T               | f
            30 |     208 |           |        43 | 2018-08-28 21:24:08 | 2018-03-07 15:37:54 | Z               | t
```



```
labdas@labdas-HP-ProDesk-400-G7-Microtower-PC: ~/Downloads
           966 |     243 |        50 |        12 | 2018-07-27 12:35:57 | 2018-02-24 11:01:19 | A               | t
           967 |     289 |        50 |        42 | 2017-10-10 09:26:37 | 2018-03-10 19:51:35 | D               | t
           968 |     515 |           |        16 | 2017-08-31 20:15:25 | 2018-06-29 02:17:03 | J               | f
           969 |     411 |        34 |        74 | 2017-01-26 15:13:22 | 2018-01-19 08:37:49 | H               | t
           970 |     494 |        13 |        44 | 2017-03-15 02:10:14 | 2018-03-10 08:28:07 | Y               | t
           971 |      97 |        35 |        54 | 2017-02-15 18:30:53 | 2018-04-26 21:16:24 | W               | f
           972 |     277 |        40 |        58 | 2018-06-15 20:00:01 | 2018-06-07 17:15:10 | S               | f
           973 |     387 |        50 |        12 | 2017-07-08 06:51:45 | 2018-01-01 01:34:13 | D               | t
           974 |     586 |        33 |        84 | 2018-01-09 09:31:53 | 2018-08-15 23:28:20 | S               | f
           975 |     478 |        46 |        23 | 2018-09-05 13:56:47 | 2018-01-13 06:38:37 | D               | t
           976 |     543 |        29 |        52 | 2017-04-26 14:51:28 | 2018-08-20 15:16:59 | T               | f
           977 |     289 |        30 |         1 | 2017-08-26 21:06:37 | 2018-07-14 02:24:13 | V               | f
           978 |     205 |        39 |        30 | 2017-01-26 01:08:10 | 2018-03-06 12:35:43 | S               | t
           979 |     397 |         8 |        80 | 2018-02-24 08:02:32 | 2018-09-02 13:06:47 | R               | f
           980 |     117 |         1 |        90 | 2017-10-19 07:30:08 | 2018-04-12 07:28:49 | A               | f
           981 |     362 |        35 |        33 | 2018-03-28 18:27:16 | 2018-08-02 12:56:35 | B               | f
           982 |     301 |           |        39 | 2017-12-21 21:03:50 | 2018-04-29 12:03:12 | K               | f
           983 |     378 |         2 |        31 | 2017-09-14 14:37:44 | 2018-07-21 11:09:48 | X               | f
           984 |     522 |         2 |        22 | 2017-05-07 16:31:41 | 2018-07-12 04:08:16 | B               | f
           985 |     427 |        21 |        56 | 2018-08-03 23:46:10 | 2018-05-14 06:31:34 | K               | f
           986 |     554 |        37 |        48 | 2018-04-30 16:53:09 | 2018-07-02 20:34:34 | Y               | f
           987 |     594 |           |        86 | 2017-08-24 01:13:00 | 2018-06-16 11:15:25 | T               | f
           988 |     420 |        18 |        26 | 2017-07-08 18:25:58 | 2018-07-15 08:53:06 | M               | f
           989 |     553 |        20 |         6 | 2018-08-14 09:30:50 | 2018-01-01 14:06:48 | H               | t
           990 |     382 |        35 |        82 | 2018-06-25 16:50:53 | 2018-02-06 18:36:14 | L               | t
           991 |     405 |        19 |        82 | 2017-11-15 19:48:41 | 2018-05-05 00:46:35 | D               | f
           992 |      75 |        24 |        50 | 2017-04-29 13:46:42 | 2018-04-14 07:09:32 | C               | f
           993 |     268 |        13 |        38 | 2017-09-14 15:52:48 | 2018-01-14 09:11:02 | A               | t
           994 |     109 |        46 |        64 | 2017-08-11 08:01:42 | 2018-05-13 23:47:13 | W               | f
           995 |     475 |        41 |        69 | 2018-06-30 21:53:10 | 2018-01-01 08:42:08 | X               | t
           996 |     530 |        17 |        70 | 2017-05-25 05:52:20 | 2018-03-28 07:40:25 | F               | t
           997 |     518 |        21 |        56 | 2017-08-27 02:45:04 | 2018-03-29 01:04:41 | S               | t
           998 |      78 |        45 |         1 | 2017-03-10 15:43:54 | 2018-05-18 07:54:22 | M               | f
           999 |     460 |        29 |        89 | 2018-07-30 23:39:25 | 2018-09-11 21:37:48 | N               | f
          1000 |     597 |        21 |         5 | 2018-08-28 05:12:49 | 2018-06-03 08:53:17 | J               | f
(1000 rows)

rumahsakit=#
```

Referensi

[PostgreSQL Functions - javatpoint]

[PostgreSQL: Documentation: 15: CREATE PROCEDURE]

[PostgreSQL: Documentation: 15: CREATE FUNCTION]

[PostgreSQL: Documentation: 15: COALESCE]

Feedback Praktikum

bit.ly/feedback-praktikum