

LAPORAN TUTORIAL LAB 2

BASIS DATA



ALVARO AUSTIN

2106752180

KELAS A

FAKULTAS ILMU KOMPUTER

UNIVERSITAS INDONESIA

DEPOK

2022/2023

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



Latihan 1

1. [SQL] Jalankan SQL Query pada Contoh 1 hingga Contoh 26 di atas dan cantumkan hasilnya pada laporan.

- Screenshot contoh 1

```
alvaro.austin=> SELECT nama
alvaro.austin-> FROM DOKTER
alvaro.austin-> WHERE id_dokter = 'DO04';
      nama
-----
Ronda Hargitt
(1 row)

alvaro.austin=>
```

- Screenshot contoh 2

```
alvaro.austin=> SELECT nama, jenis, kapasitas
alvaro.austin-> FROM KAMAR
alvaro.austin-> ORDER BY harga;
      nama      | jenis | kapasitas
-----+-----+-----
Merpati 1 | Kelas 3 | 6
Merpati 2 | Kelas 3 | 6
Merpati 3 | Kelas 3 | 6
Merpati 4 | Kelas 3 | 6
Merpati 5 | Kelas 3 | 6
Jalak 1 | Kelas 2 | 4
Jalak 2 | Kelas 2 | 4
Jalak 3 | Kelas 2 | 4
Jalak 4 | Kelas 2 | 4
Jalak 5 | Kelas 2 | 4
Murai 1 | Kelas 1 | 2
Murai 2 | Kelas 1 | 2
Murai 3 | Kelas 1 | 2
Murai 4 | Kelas 1 | 2
Murai 5 | Kelas 1 | 2
Merak 1 | VIP B | 1
Merak 2 | VIP B | 1
Merak 3 | VIP B | 1
Merak 4 | VIP B | 1
Merak 5 | VIP B | 1
Merak 6 | VIP B | 1
Merak 7 | VIP B | 1
Merak 8 | VIP B | 1
Merak 9 | VIP B | 1
Merak 10 | VIP B | 1
Merak 11 | VIP B | 1
Merak 12 | VIP B | 1
Merak 13 | VIP B | 1
Merak 14 | VIP B | 1
Merak 15 | VIP B | 1
```

```
Elang 1 | VIP A | 1
Elang 2 | VIP A | 1
Elang 3 | VIP A | 1
Elang 4 | VIP A | 1
Elang 5 | VIP A | 1
Elang 6 | VIP A | 1
Elang 7 | VIP A | 1
Elang 8 | VIP A | 1
Elang 9 | VIP A | 1
Elang 10 | VIP A | 1
Garuda 1 | VVIP | 1
Garuda 2 | VVIP | 1
Garuda 3 | VVIP | 1
Garuda 4 | VVIP | 1
Garuda 5 | VVIP | 1
(45 rows)

alvaro.austin=>
```

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



- Screenshot contoh 3

```
alvaro.austin=> SELECT *
alvaro.austin-> FROM KAMAR
alvaro.austin-> WHERE (kapasitas >= 2) AND (kapasitas <= 4);
id_kamar | nama      | jenis | kapasitas | harga
-----+-----+-----+-----+-----
KA06     | Jalak 1   | Kelas 2 | 4         | 300000
KA07     | Jalak 2   | Kelas 2 | 4         | 300000
KA08     | Jalak 3   | Kelas 2 | 4         | 300000
KA09     | Jalak 4   | Kelas 2 | 4         | 300000
KA10     | Jalak 5   | Kelas 2 | 4         | 300000
KA11     | Murai 1   | Kelas 1 | 2         | 700000
KA12     | Murai 2   | Kelas 1 | 2         | 700000
KA13     | Murai 3   | Kelas 1 | 2         | 700000
KA14     | Murai 4   | Kelas 1 | 2         | 700000
KA15     | Murai 5   | Kelas 1 | 2         | 700000
(10 rows)
alvaro.austin=>
```

```
alvaro.austin=> SELECT *
alvaro.austin-> FROM KAMAR
alvaro.austin-> WHERE (kapasitas BETWEEN 2 AND 4);
id_kamar | nama      | jenis | kapasitas | harga
-----+-----+-----+-----+-----
KA06     | Jalak 1   | Kelas 2 | 4         | 300000
KA07     | Jalak 2   | Kelas 2 | 4         | 300000
KA08     | Jalak 3   | Kelas 2 | 4         | 300000
KA09     | Jalak 4   | Kelas 2 | 4         | 300000
KA10     | Jalak 5   | Kelas 2 | 4         | 300000
KA11     | Murai 1   | Kelas 1 | 2         | 700000
KA12     | Murai 2   | Kelas 1 | 2         | 700000
KA13     | Murai 3   | Kelas 1 | 2         | 700000
KA14     | Murai 4   | Kelas 1 | 2         | 700000
KA15     | Murai 5   | Kelas 1 | 2         | 700000
(10 rows)
alvaro.austin=>
```

- Screenshot contoh 4

```
alvaro.austin=> SELECT nama, email
alvaro.austin-> FROM PERAWAT
alvaro.austin-> WHERE nama LIKE 'Karen%';
      nama      | email
-----+-----
Karen Cotilard | kcotilardk@omniture.com
(1 row)
alvaro.austin=>
```

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



- Screenshot contoh 5

```
alvaro.austin-> (SELECT * FROM DOKTER
alvaro.austin(> WHERE spesialisasi = 'bedah');
id_dokter |      nama      | no_telp |      email      | spesialisasi
-----+-----+-----+-----+-----
DO08      | Den Stollsteimer | 85054750630 | dstollsteimer7@myspace.com | anak
DO09      | Jenda Chaudret  | 87860070111 | jchaudret8@example.com    | anak
DO10      | Moreen Hebblethwaite | 83109586118 | mhebblethwaite9@buzzfeed.com | anak
DO11      | Idalia Caughte  | 88041581978 | icaughtea@samsung.com     | anak
DO12      | Darbie Howle    | 84548778195 | dhowleb@zimbio.com         | anak
DO13      | Meade Munn      | 85430172820 | mmunnnc@si.edu            | anak
DO14      | Josephine Schnieder | 84551802662 | jschniederd@umn.edu       | anak
DO29      | Devlin Earney   | 81003195082 | dearneys@sbwire.com       | bedah
DO30      | Carl Adelsberg  | 87917206718 | cadelsbergt@networksolutions.com | bedah
DO31      | Gearard Stapells | 85738280407 | gstapellsu@hc360.com      | bedah
DO32      | Cassandre Noddles | 81220215324 | cnoddlesv@dell.com        | bedah
DO33      | Genvieve Guillotin | 81315397179 | gguillotinw@theatlantic.com | bedah
DO34      | Tiena Klimmek   | 89104117514 | tklimmekx@ocn.ne.jp       | bedah
DO35      | Garry Aspinell  | 89869528031 | gaspinelly@cargocollective.com | bedah
(14 rows)

alvaro.austin=>
```

- Screenshot contoh 6

```
alvaro.austin=> (SELECT * FROM DOKTER
alvaro.austin(> WHERE nama LIKE 'C%')
alvaro.austin-> INTERSECT
alvaro.austin-> (SELECT * FROM DOKTER
alvaro.austin(> WHERE spesialisasi = 'bedah');
id_dokter |      nama      | no_telp |      email      | spesialisasi
-----+-----+-----+-----+-----
DO30      | Carl Adelsberg  | 87917206718 | cadelsbergt@networksolutions.com | bedah
DO32      | Cassandre Noddles | 81220215324 | cnoddlesv@dell.com        | bedah
(2 rows)

alvaro.austin=>
```

- Screenshot contoh 7

```
alvaro.austin=> (SELECT * FROM DOKTER
alvaro.austin(> WHERE nama LIKE 'C%')
alvaro.austin-> EXCEPT
alvaro.austin-> (SELECT * FROM DOKTER
alvaro.austin(> WHERE spesialisasi = 'bedah');
id_dokter |      nama      | no_telp |      email      | spesialisasi
-----+-----+-----+-----+-----
DO20      | Candy Armingier | 84319934032 | carmingierj@abc.net.au    | saraf
(1 row)

alvaro.austin=>
```

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



```
alvaro.austin=> SELECT nama
alvaro.austin-> FROM PASIEN
alvaro.austin-> WHERE EXTRACT(DECADE FROM tgl_lahir) = '199';
      nama
-----
Padraic De Francisci
Friedrick Pina
Sibella Raddin
Virgil Clutterbuck
Hamlen Ryles
Andrew Ginnell
Meade Redington
Maitilde Cundey
Judye Phillip
Genna Mechem
Elena Badgers
Bobbye Elner
(12 rows)

alvaro.austin=> █
```

- Screenshot contoh 9:

```
alvaro.austin=> SELECT P.nama
alvaro.austin-> FROM PASIEN P, KAMAR K, RAWAT_INAP R
alvaro.austin-> WHERE R.id_pasien=P.id_pasien AND
alvaro.austin->          R.id_kamar=K.id_kamar AND
alvaro.austin->          K.nama='Merak 1';
      nama
-----
(0 rows)

alvaro.austin=> █
```

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

- Screenshot contoh 10 (Berdasarkan tutorial tidak perlu melampirkan seluruh screenshot):

alvaro.austin-> FROM PERAWAT CROSS JOIN SHIFT_PERAWAT;										waktu_mulai	tgl_keluar
id_perawat	nama	no_telp	email	jenis_kelamin	id_shift_perawat	id_perawat	id_rawat_inap				
FE01	Asa Huddleston	84578304515	ahuddleston0@businessinsider.com	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE02	Esina Georgiou	85227801894	egeorgiou1@yelp.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE03	Jeanna Basilone	89838398853	jbasilone2@furl.net	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE04	Gannie Stewart	82141133165	gstewart3@mb.com	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE05	Efram Wikey	86587160468	ewikey4@jiggy.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE06	Talbot Dust	84354834510	tdust5@csmonitor.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE07	Sydel Fiveash	84079342810	sfiveash6@yellowbook.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE08	Constantine Hukin	81254798730	chukin7@google.com.au	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE09	Giacinta Graine	81338605844	ggraine8@seesaa.net	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE10	Sayers Windmill	82265157929	swindmill9@csmonitor.com	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE11	Mair Riddich	89376194780	mriddich@google.co.uk	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE12	Stanwood Redhole	81661848974	sredholeb@cafepress.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE13	Noreen Agostini	81599259095	nagostinic@about.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE14	Harli Liscandro	89888092944	hliscandro@irs.gov	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE15	Brannon Sebring	85469789160	bsebringe@usatoday.com	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE16	Charyl Doozbar	86351061660	cdoozbarf@economist.com	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE17	Sanderson Worg	83770914651	sworg@oakley.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE18	Russell Prestedge	87575175601	rprestedgeh@shutterfly.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE19	Matthew Flockhart	82868873219	mflockharti@technorati.com	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE20	Clarabelle Gong	81173374328	cgongj@globo.com	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE21	Karon Cotillard	81302516078	kcotillardk@omniture.com	P	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE22	Clea Ebdin	83017433463	cebdinl@ftc.gov	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE23	Dallis Rummings	81381398278	drummingsm@google.ru	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00
FE24	Jandy Bassingham	88953069688	jbassingham@pen.io	L	SP01	PE02	RI23			2021-12-07 00:00:00	2021-12-07 12:00:00

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA

Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

- Screenshot contoh 11 (Berdasarkan tutorial tidak perlu menunjukkan seluruh tabel):

alvaro.austin=> SELECT *								waktu_mulai	tgl_keluar
alvaro.austin=> FROM PERAWAT P JOIN SHIFT_PERAWAT S									
alvaro.austin=> ON P.id_perawat = S.id_perawat/									
id_perawat nama no_telp email jenis_kelamin id_shift_perawat id_perawat id_rawat_inap									
mulai tgl_keluar									
PE02	Erina Georgiou	85227801894	egeorghiou1@yelp.com	F	SP01	PE02	RI23	2021-12-07 00:00:00	2021-12-07 12:00:00
07 00:00:00	2021-12-07 12:00:00							2022-02-02 00:00:00	2022-02-02 12:00:00
PE05	Efram Wikey	86587160468	ewikey4@jigsys.com	F	SP02	PE05	RI14	2020-09-11 00:00:00	2020-09-11 12:00:00
02 00:00:00	2022-02-02 12:00:00							2022-06-29 00:00:00	2022-06-29 12:00:00
PE39	Mellicent Crawley	89047721632	mcrawley12@google.co.jp	L	SP03	PE39	RI49	2020-12-03 00:00:00	2020-12-03 12:00:00
11 00:00:00	2020-09-11 12:00:00							2022-09-06 00:00:00	2022-09-06 12:00:00
PE27	Jeth Barlow	85529041701	jbarlowq@tinypic.com	F	SP04	PE27	RI42	2021-04-29 00:00:00	2021-04-29 12:00:00
28 00:00:00	2022-06-29 12:00:00							2022-08-13 00:00:00	2022-08-13 12:00:00
PE39	Mellicent Crawley	89047721632	mcrawley12@google.co.jp	L	SP05	PE39	RI10	2021-04-29 12:00:00	2021-04-30 00:00:00
03 00:00:00	2020-12-03 12:00:00							2020-06-05 00:00:00	2020-06-05 12:00:00
PE04	Gannie Stewart	82141133165	gstewart3@mlb.com	L	SP06	PE04	RI36	2020-04-27 00:00:00	2020-04-27 12:00:00
06 00:00:00	2022-09-06 12:00:00							2020-11-13 00:00:00	2020-11-13 12:00:00
PE31	Kelcy Fraschini	84659987329	kfraschiniu@google.it	L	SP07	PE31	RI05	2022-08-13 12:00:00	2022-08-14 00:00:00
29 00:00:00	2021-04-29 12:00:00							2020-03-22 00:00:00	2020-03-22 12:00:00
PE34	Shaughn Bernardotte	84852720894	sbernardottex@huffingtonpost.com	F	SP08	PE34	RI01	2022-03-28 00:00:00	2022-03-28 12:00:00
13 00:00:00	2022-08-13 12:00:00							2020-10-18 00:00:00	2020-10-18 12:00:00
PE47	Andrej Sandland	85005433694	asandlandia@usnews.com	L	SP09	PE47	RI05	2022-01-20 00:00:00	2022-01-20 12:00:00
29 12:00:00	2021-04-30 00:00:00							2022-05-26 00:00:00	2022-05-26 12:00:00
PE34	Shaughn Bernardotte	84852720894	sbernardottex@huffingtonpost.com	F	SP10	PE34	RI28	2022-03-30 00:00:00	2022-03-30 12:00:00
05 00:00:00	2020-06-05 12:00:00							2021-01-16 00:00:00	2021-01-16 12:00:00
PE41	Dasi Mirama	81891579842	dmirama14@wired.com	F	SP11	PE41	RI15	2020-03-02 00:00:00	2020-03-02 12:00:00
27 00:00:00	2020-04-27 12:00:00							2022-05-26 12:00:00	2022-05-27 00:00:00
PE15	Brannon Sebring	85469789160	bsebringe@usatoday.com	L	SP12	PE15	RI04	2020-12-03 12:00:00	2020-12-04 00:00:00
13 00:00:00	2020-11-13 12:00:00							2022-09-06 12:00:00	2022-09-07 00:00:00
PE35	Ellis Arndell	87453241219	earndelly@about.me	F	SP13	PE35	RI01	2021-08-13 00:00:00	2021-08-13 12:00:00
13 12:00:00	2022-08-14 00:00:00							2022-09-07 00:00:00	2022-09-07 12:00:00
PE28	Mitchell Greger	86521338439	mgregerr@chronoengine.com	L	SP14	PE28	RI26	2021-12-28 00:00:00	2021-12-28 12:00:00
22 00:00:00	2020-03-22 12:00:00							2022-01-03 00:00:00	2022-01-03 12:00:00
PE46	Gipsy Gullefant	89545827319	ggullefant19@theguardian.com	L	SP15	PE46	RI16	2022-05-18 00:00:00	2022-05-18 12:00:00
28 00:00:00	2022-03-28 12:00:00							2021-08-01 00:00:00	2021-08-01 12:00:00
PE20	ClaraBelle Gong	81173374328	cgongj@globlo.com	L	SP16	PE20	RI22	2022-05-18 12:00:00	2022-05-19 00:00:00
18 00:00:00	2020-10-18 12:00:00							2020-11-30 00:00:00	2020-11-30 12:00:00
PE27	Jeth Barlow	85529041701	jbarlowq@tinypic.com	F	SP17	PE27	RI13	2021-08-12 00:00:00	2021-08-12 12:00:00
20 00:00:00	2022-01-20 12:00:00							2021-08-01 12:00:00	2021-08-02 00:00:00
PE47	Andrej Sandland	85005433694	asandlandia@usnews.com	L	SP18	PE47	RI24	2022-02-03 00:00:00	2022-02-03 12:00:00
26 00:00:00	2022-05-26 12:00:00							2022-06-29 12:00:00	2022-06-30 00:00:00
PE35	Ellis Arndell	87453241219	earndelly@about.me	F	SP19	PE35	RI45	2021-12-04 00:00:00	2021-12-04 12:00:00
30 00:00:00	2022-03-30 12:00:00							2021-12-04 12:00:00	2021-12-05 00:00:00
PE26	Willow De Cruce	85175931741	wdep@buzzfeed.com	L	SP20	PE26	RI27	2021-08-02 00:00:00	2021-08-02 12:00:00
16 00:00:00	2021-01-16 12:00:00							2021-12-16 00:00:00	2021-12-16 12:00:00
PE33	Lizette Rodolico	85849741984	lrodolico@uiuc.edu	F	SP21	PE33	RI30	2021-08-13 12:00:00	2021-08-14 00:00:00
02 00:00:00	2020-03-02 12:00:00							2021-07-03 00:00:00	2021-07-03 12:00:00
PE24	Jandy Bassingham	88953069688	jbassingham@pen.io	L	SP22	PE24	RI24	2021-12-07 12:00:00	2021-12-08 00:00:00
26 12:00:00	2022-05-27 00:00:00							2022-06-08 00:00:00	2022-06-08 12:00:00
PE05	Efram Wikey	86587160468	ewikey4@jigsys.com	F	SP23	PE05	RI10	2020-06-25 00:00:00	2020-06-25 12:00:00
03 12:00:00	2020-12-04 00:00:00							2021-08-12 12:00:00	2021-08-13 00:00:00

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

- Screenshot contoh 12 (Berdasarkan tutorial tidak perlu menunjukkan seluruh tabel)

```
alvaro.austin-> SELECT *
alvaro.austin-> FROM PERAWAT P LEFT OUTER JOIN SHIFT_PERAWAT S
alvaro.austin-> ON P.id_perawat = S.id_perawat;
```

id_perawat	nama	no_telp	email	jenis_kelamin	id_shift_perawat	id_perawat	id_rawat_inap	waktu_mulai	tgl_keluar
PE02	Erina Georghiou	85227801894	egeorghioul@yelp.com	P	SP01	PE02	RI23	2021-12-07 00:00:00	2021-12-07 12:00:00
PE05	Efrem Wikey	86587160468	ewikey4@jigasy.com	P	SP02	PE05	RI14	2022-02-02 00:00:00	2022-02-02 12:00:00
PE39	Mellicent Crawley	89047721632	mcrawley12@google.co.jp	L	SP03	PE39	RI49	2020-09-11 00:00:00	2020-09-11 12:00:00
PE27	Jeth Barlow	85529041701	jbarlowq@tinypic.com	P	SP04	PE27	RI42	2022-06-29 00:00:00	2022-06-29 12:00:00
PE39	Mellicent Crawley	89047721632	mcrawley12@google.co.jp	L	SP05	PE39	RI10	2020-12-03 00:00:00	2020-12-03 12:00:00
PE04	Gannie Stewart	82141133165	gstewart3@mlb.com	L	SP06	PE04	RI36	2022-09-06 00:00:00	2022-09-06 12:00:00
PE31	Kelcy Franchini	84659987329	kfranchiniu@google.it	L	SP07	PE31	RI05	2021-04-29 00:00:00	2021-04-29 12:00:00
PE34	Shaughn Bernardotte	84852720894	sbernardottex@huffingtonpost.com	P	SP08	PE34	RI01	2022-08-13 00:00:00	2022-08-13 12:00:00
PE47	Andrei Sandland	85005433694	asandlandia@usnews.com	L	SP09	PE47	RI05	2021-04-29 12:00:00	2021-04-30 00:00:00
PE34	Shaughn Bernardotte	84852720894	sbernardottex@huffingtonpost.com	P	SP10	PE34	RI28	2020-06-05 00:00:00	2020-06-05 12:00:00
PE41	Dasi Mirams	81891579842	dmirams14@wired.com	P	SP11	PE41	RI15	2020-04-27 00:00:00	2020-04-27 12:00:00
PE15	Brannon Sebring	85469789160	bsebringe@usatoday.com	L	SP12	PE15	RI04	2020-11-13 00:00:00	2020-11-13 12:00:00
PE35	Ella Arndell	87453241219	earndelly@about.me	P	SP13	PE35	RI01	2022-08-13 12:00:00	2022-08-14 00:00:00
PE28	Mitchell Greger	86521338439	mgregerr@chronoengine.com	L	SP14	PE28	RI26	2020-03-22 00:00:00	2020-03-22 12:00:00
PE46	Gipsy Gullefant	89545827319	ggullefant19@theguardian.com	L	SP15	PE46	RI16	2022-03-28 00:00:00	2022-03-28 12:00:00
PE20	Clarabelle Gong	81173374328	cgongj@globob.com	L	SP16	PE20	RI22	2020-10-18 00:00:00	2020-10-18 12:00:00
PE27	Jeth Barlow	85529041701	jbarlowq@tinypic.com	P	SP17	PE27	RI13	2022-01-20 00:00:00	2022-01-20 12:00:00
PE47	Andrei Sandland	85005433694	asandlandia@usnews.com	L	SP18	PE47	RI24	2022-05-26 00:00:00	2022-05-26 12:00:00
PE35	Ella Arndell	87453241219	earndelly@about.me	P	SP19	PE35	RI45	2022-03-30 00:00:00	2022-03-30 12:00:00
PE26	Willow De Cruce	85175931741	wdep@buzzfeed.com	L	SP20	PE26	RI27	2021-01-16 00:00:00	2021-01-16 12:00:00
PE33	Lizette Rodolico	85849741984	lrodolicow@uiuc.edu	P	SP21	PE33	RI30	2020-03-02 00:00:00	2020-03-02 12:00:00
PE24	Jandy Bassingham	88953069688	jbassingham@pen.io	L	SP22	PE24	RI24	2022-05-26 12:00:00	2022-05-27 00:00:00
PE05	Efrem Wikey	86587160468	ewikey4@jigasy.com	P	SP23	PE05	RI10	2020-12-04 00:00:00	2020-12-04 12:00:00
PE37	Nobe Claye	81253032172	nclayel0@artisteer.com	P					
PE21	Karen Cotilard	81302516078	kcotilardk@omniture.com	P					
PE08	Constantine Hukin	81254798730	chukin7@google.com.au	L					
PE06	Talbot Dust	84354834510	tdust5@csmonitor.com	P					
PE01	Asa Huddlestone	84578304515	ahuddlestone0@businessinsider.com	L					
PE29	Vonny Northway	84801332978	vnorthways@bigcartel.com	P					
PE16	Charyl Doorbar	86351061660	cdoorbarf@economist.com	L					
PE32	Rafa Demare	81681256401	rdemarev@sakura.ne.jp	L					
PE48	Jeremias Crickmore	86551516106	jcrickmorelb@furl.net	P					
PE50	Bella Prall	84027794222	bprallld@squarepace.com	P					

(110 rows)

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



- Screenshot contoh 13 (Berdasarkan tutorial tidak perlu menunjukkan seluruh tabel)

```
alvaro.austin-> SELECT *
alvaro.austin-> FROM SHIFT_PERAWAT S RIGHT OUTER JOIN PERAWAT P
alvaro.austin-> ON P.id_perawat = S.id_perawat;
id_shift_perawat | id_perawat | id_rawat_inap | waktu_mulai | tgl_keluar | id_perawat | nama | no_telp |
mail | jenis_kelamin
```

SP01	PE02	RI23	2021-12-07 00:00:00	2021-12-07 12:00:00	PE02	Erina Georghiou	85227801894	egeorghioul@ye
p.com	P							
SP02	PE05	RI14	2022-02-02 00:00:00	2022-02-02 12:00:00	PE05	Efrem Wikey	86587160468	ewikey4@jigsy.c
om	P							
SP03	PE39	RI49	2020-09-11 00:00:00	2020-09-11 12:00:00	PE39	Mellicent Crawley	89047721632	mccrawley12@goo
le.co.jp	L							
SP04	PE27	RI42	2022-06-29 00:00:00	2022-06-29 12:00:00	PE27	Jeth Barlow	85529041701	jbarlowq@tinyp
c.com	P							
SP05	PE39	RI10	2020-12-03 00:00:00	2020-12-03 12:00:00	PE39	Mellicent Crawley	89047721632	mccrawley12@goo
le.co.jp	L							
SP06	PE04	RI36	2022-09-06 00:00:00	2022-09-06 12:00:00	PE04	Gannie Stewart	82141133165	gstewart3@mlb.c
om	L							
SP07	PE31	RI05	2021-04-29 00:00:00	2021-04-29 12:00:00	PE31	Kelcy Fraschini	84659987329	kfraschini@goo
gle.it	L							
SP08	PE34	RI01	2022-08-13 00:00:00	2022-08-13 12:00:00	PE34	Shaughn Bernardotte	84852720894	sbernardottex@
uffingtonpost.com	P							
SP09	PE47	RI05	2021-04-29 12:00:00	2021-04-30 00:00:00	PE47	Andrei Sandland	85005433694	asandlandia@uss
ews.com	L							
SP10	PE34	RI28	2020-06-05 00:00:00	2020-06-05 12:00:00	PE34	Shaughn Bernardotte	84852720894	sbernardottex@
uffingtonpost.com	P							
SP11	PE41	RI15	2020-04-27 00:00:00	2020-04-27 12:00:00	PE41	Dasi Mirams	81891579842	dmirams14@wire
.com	P							
SP12	PE15	RI04	2020-11-13 00:00:00	2020-11-13 12:00:00	PE15	Brannon Sebring	85469789160	bsebringe@usato
day.com	L							
SP13	PE35	RI01	2022-08-13 12:00:00	2022-08-14 00:00:00	PE35	Ella Arndell	87453241219	earndelly@about
me	P							
SP14	PE28	RI26	2020-03-22 00:00:00	2020-03-22 12:00:00	PE28	Mitchell Greger	86521338439	mgregerr@chron
ngine.com	L							
SP15	PE46	RI16	2022-03-28 00:00:00	2022-03-28 12:00:00	PE46	Gipsy Gullefant	89545827319	ggullefant19@th
eguardian.com	L							
SP16	PE20	RI22	2020-10-18 00:00:00	2020-10-18 12:00:00	PE20	Clarabelle Gong	81173374328	cgongj@globo.co
m	L							
SP17	PE27	RI13	2022-01-20 00:00:00	2022-01-20 12:00:00	PE27	Jeth Barlow	85529041701	jbarlowq@tinyp
c.com	P							
SP18	PE47	RI24	2022-05-26 00:00:00	2022-05-26 12:00:00	PE47	Andrei Sandland	85005433694	asandlandia@uss
ews.com	L							
SP19	PE35	RI45	2022-03-30 00:00:00	2022-03-30 12:00:00	PE35	Ella Arndell	87453241219	earndelly@about
me	P							
SP20	PE26	RI27	2021-01-16 00:00:00	2021-01-16 12:00:00	PE26	Willow De Cruce	85175931741	wdep@buzzfeed.c
om	L							
SP21	PE33	RI30	2020-03-02 00:00:00	2020-03-02 12:00:00	PE33	Lizette Rodolico	85849741984	lrodolicow@uiuc
edu	P							
SP22	PE24	RI24	2022-05-26 12:00:00	2022-05-27 00:00:00	PE24	Jandy Bassingham	88953069688	jbassingham@pe
n.io	L							
SP23	PE05	RI10	2020-12-03 12:00:00	2020-12-04 00:00:00	PE05	Efrem Wikey	86587160468	ewikey4@jigsy.c
om	P							
e.co.uk	L							
					PE37	Nobe Claye	81253032172	ncclayel0@artist
eer.com	P							
					PE21	Karen Cotilard	81302516078	kcotilardk@omni
ture.com	P							
					PE08	Constantine Hukin	81254798730	chukin7@google.
com.au	L							
					PE06	Talbot Dust	84354834510	tdust5@csmonit
r.com	P							
					PE01	Asa Huddlestone	84578304515	ahuddlestone0@h
usinessinsider.com	L							
					PE29	Vonny Northway	84801332978	vnorthways@big
artel.com	P							
					PE16	Charyl Doorbar	86351061660	cdoorbarf@econ
mist.com	L							
					PE32	Rafa Demare	81681256401	rdemarev@sakura
.ne.jp	L							
					PE48	Jeremias Crickmore	86551516106	jcrickmore1b@fu
rl.net	P							
					PE50	Bella Prall	84027794222	bprall1d@square
space.com	P							
(110 rows)								

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



- Screenshot contoh 14 (Berdasarkan tutorial tidak perlu menunjukkan seluruh tabel)

```
alvaro.austin-> SELECT *
FROM OBAT O FULL OUTER JOIN PEMBERIAN_OBAT PO
ON O.id_obat = PO.id_obat;
```

id_obat	id_shift_perawat	id_obat	dosis	nama	id_pemberian
OB01		calcitonin salmon			PO40
	SP32	OB01	7 mg		
OB01		calcitonin salmon			PO11
	SP22	OB01	53 mg		
OB02		Estazolam			PO60
	SP89	OB02	54 mg		
OB02		Estazolam			PO43
	SP15	OB02	50 mg		
OB02		Estazolam			PO22
	SP13	OB02	43 mg		
OB02		Estazolam			PO14
	SP01	OB02	70 mg		
OB03		ARNICA MONTANA - RHUS TOXICODENDRON - RUTA GRAVEOLENS			
OB04		ANTIPYRINE AND BENZOCAINE			PO75
	SP77	OB04	86 mg		
OB04		ANTIPYRINE AND BENZOCAINE			PO32
	SP22	OB04	45 mg		
OB05		Phenobarbital			PO83
	SP34	OB05	66 mg		
OB05		Phenobarbital			PO19
	SP22	OB05	20 mg		
OB06		levofloxacin			PO73
	SP42	OB06	36 mg		
OB06		levofloxacin			PO66
	SP91	OB06	68 mg		
OB06		levofloxacin			PO07
	SP78	OB06	66 mg		
OB07		Albuterol Sulfate			PO71
	SP68	OB07	36 mg		
OB08		RISPERIDONE			PO100
	SP94	OB08	60 mg		
OB08		RISPERIDONE			PO67
	SP62	OB08	42 mg		
OB08		RISPERIDONE			PO49
	SP52	OB08	45 mg		
OB08		RISPERIDONE			PO36
	SP91	OB08	15 mg		
OB08		RISPERIDONE			PO28
	SP13	OB08	94 mg		
OB09		Levothyroxine Sodium			PO63
	SP20	OB09	20 mg		
OB10		calcipotriene 0.005% and betamethasone dipropionate 0.064%			PO93

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

- Screenshot contoh 15

```
alvaro.austin=> SELECT *
FROM PERAWAT NATURAL JOIN SHIFT_PERAWAT;
```

id_perawat	nama	no_telp	email	jenis_kelamin	id_shift_perawat	id_rawat_inap	waktu_mulai	tgl_keluar
FE02	Erina Georghiou	85227801894	egeorghioul@yelp.com	P	SP01	RI23	2021-12-07 12:00:00	
FE05	Efrem Wikey	86587160468	ewikey4@jigsgy.com	P	SP02	RI14	2022-02-02 12:00:00	
FE39	Mellicent Crawley	89047721632	mcrawleyl2@google.co.jp	L	SP03	RI49	2020-09-11 12:00:00	
FE27	Jeth Barlow	85529041701	jbarlowq@tinypic.com	P	SP04	RI42	2022-06-29 12:00:00	
FE39	Mellicent Crawley	89047721632	mcrawleyl2@google.co.jp	L	SP05	RI10	2020-12-03 12:00:00	
FE04	Gannie Stewart	82141133165	gstewart3@mlb.com	L	SP06	RI36	2022-09-06 12:00:00	
FE31	Kelcy Fraschini	84659987329	kfraschiniu@google.it	L	SP07	RI05	2021-04-29 12:00:00	
FE34	Shaughn Bernardotte	84852720894	sbernardottex@huffingtonpost.com	P	SP08	RI01	2022-08-13 12:00:00	
FE47	Andrei Sandland	85005433694	asandlandla@usnews.com	L	SP09	RI05	2021-04-29 12:00:00	
FE34	Shaughn Bernardotte	84852720894	sbernardottex@huffingtonpost.com	P	SP10	RI28	2020-06-05 12:00:00	
FE41	Dasi Mirams	81891579842	dmirams14@wired.com	P	SP11	RI15	2020-04-27 12:00:00	
FE15	Brannon Sebring	85469789160	bsebring@usatoday.com	L	SP12	RI04	2020-11-13 12:00:00	
FE35	Ella Arndell	87453241219	earndelly@about.me	P	SP13	RI01	2022-08-13 12:00:00	
FE28	Mitchell Greger	86521338439	mgregerr@chronoengine.com	L	SP14	RI26	2020-03-22 12:00:00	
FE46	Gipsy Gullefant	89545827319	ggullefant19@theguardian.com	L	SP15	RI16	2022-03-28 12:00:00	
FE20	Clarabelle Gong	81173374328	cgongj@globo.com	L	SP16	RI22	2020-10-18 12:00:00	
FE27	Jeth Barlow	85529041701	jbarlowq@tinypic.com	P	SP17	RI13	2022-01-20 12:00:00	
FE47	Andrei Sandland	85005433694	asandlandla@usnews.com	L	SP18	RI24	2022-05-26 12:00:00	
FE35	Ella Arndell	87453241219	earndelly@about.me	P	SP19	RI45	2022-03-30 12:00:00	
FE26	Willow De Cruce	85175931741	wdep@buzzfeed.com	L	SP20	RI27	2021-01-16 12:00:00	
FE33	Lizette Rodolico	85849741984	lrodolicow@uiuc.edu	P	SP21	RI30	2020-03-02 12:00:00	
FE24	Jandy Bassingham	88953069688	jbassingham@pen.io	L	SP22	RI24	2022-05-26 12:00:00	

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



- **Screenshot contoh 16**

```
alvaro.austin=> SELECT id_pasien
alvaro.austin-> FROM RAWAT_INAP
alvaro.austin-> WHERE tgl_keluar IS NOT NULL;
id_pasien
-----
PA11
PA29
PA02
PA47
PA44
PA33
PA07
PA34
PA12
PA48
PA34
PA22
PA14
PA44
PA08
PA09
PA43
PA20
PA49
PA29
PA12
PA06
PA27
PA28
PA36
PA42
PA22
PA27
PA22
PA02
PA46
PA03
PA05
PA01
PA21
PA01
PA49
PA40
PA28
PA29
PA24
PA30
PA45
PA23
PA39
(45 rows)

alvaro.austin=> █
```

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



- Screenshot contoh 17

```
alvaro.austin=> SELECT id_pasien
alvaro.austin-> FROM RAWAT_INAP
alvaro.austin-> WHERE tgl_keluar IS NULL;
  id_pasien
-----
  PA43
  PA22
  PA19
  PA25
  PA32
(5 rows)

alvaro.austin=> █
```

- Screenshot contoh 18

```
alvaro.austin=> SELECT O.nama
alvaro.austin-> FROM OBAT O
alvaro.austin-> WHERE O.id_obat IN (
alvaro.austin(>   SELECT PO.id_obat
alvaro.austin(>   FROM PEMBERIAN_OBAT PO, SHIFT_PERAWAT SP, PERAWAT P
alvaro.austin(>   WHERE PO.id_shift_perawat = SP.id_shift_perawat AND
alvaro.austin(>         SP.id_perawat = P.id_perawat AND
alvaro.austin(>         P.nama = 'Mitchell Greger'
alvaro.austin(> );
                                nama
-----
  levofloxacin
  RISPERIDONE
  Acetaminophen
  Triclosan
  Codeine Phosphate, Phenylephrine Hydrochloride, Pyrilamine Maleate
(5 rows)

alvaro.austin=> █
```

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



- Screenshot contoh 19

```
alvaro.austin=> SELECT P.id_perawat, P.nama
alvaro.austin-> FROM PERAWAT P
alvaro.austin-> WHERE NOT EXISTS (
alvaro.austin(>   SELECT *
alvaro.austin(>   FROM PEMBERIAN_OBAT PO, SHIFT_PERAWAT SP
alvaro.austin(>   WHERE PO.id_shift_perawat=SP.id_shift_perawat AND
alvaro.austin(>   SP.id_perawat=P.id_perawat
alvaro.austin(> );
  id_perawat |      nama
-----+-----
PE01         | Asa Huddleston
PE04         | Gannie Stewart
PE06         | Talbot Dust
PE08         | Constantine Hukin
PE16         | Charyl Doorbar
PE18         | Russell Prestedge
PE21         | Karen Cotilard
PE23         | Dalis Rummings
PE29         | Vonny Northway
PE32         | Rafa Demare
PE37         | Nobe Claye
PE48         | Jeremias Crickmore
PE50         | Bella Prall
(13 rows)

alvaro.austin=> █
```

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



- Screenshot contoh 20

```
alvaro.austin=> SELECT min(harga)
alvaro.austin-> FROM KAMAR;
      min
-----
170000
(1 row)

alvaro.austin=> █
```

- Screenshot contoh 21

```
alvaro.austin=> SELECT max(harga)
alvaro.austin-> FROM KAMAR;
      max
-----
3100000
(1 row)

alvaro.austin=> █
```

- Screenshot contoh 22

```
alvaro.austin=> SELECT COUNT(*)
alvaro.austin-> FROM KAMAR;
      count
-----
45
(1 row)

alvaro.austin=> █
```

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



- Screenshot contoh 23

```
alvaro.austin=> SELECT sum(kapasitas)
alvaro.austin-> FROM KAMAR;
  sum
-----
   90
(1 row)

alvaro.austin=> █
```

- Screenshot contoh 24

```
alvaro.austin=> SELECT AVG(harga)
alvaro.austin-> FROM KAMAR;
      avg
-----
1196666.666666666667
(1 row)

alvaro.austin=> █
```

- Screenshot contoh 25

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

```
alvaro.austin=> SELECT jenis, SUM(kapasitas)
alvaro.austin-> FROM KAMAR
alvaro.austin-> GROUP BY jenis;
```

jenis	sum
Kelas 1	10
VVIP	5
Kelas 2	20
VIP A	10
Kelas 3	30
VIP B	15

(6 rows)

```
alvaro.austin=>
```

```
alvaro.austin=> SELECT EXTRACT(YEAR FROM P.tgl_lahir) as TAHUN, COUNT(*)
alvaro.austin-> FROM PASIEN P
alvaro.austin-> GROUP BY EXTRACT (YEAR FROM P.tgl_lahir);
```

tahun	count
1979	4
1994	3
1968	1
1969	2
1966	2
1982	3
1960	2
1963	1
1999	1
1964	1
1995	2
1977	1
1974	1
1988	1
1993	1
1981	1
1996	1
1971	1
1962	2
2002	1
1983	1
1992	2
1984	2
1961	2
2004	1
1967	2
2005	1
1990	1
1970	1
1998	1
1972	3

(31 rows)

```
alvaro.austin=>
```

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



- Screenshot contoh 26

```
alvaro.austin=> SELECT K.jenis, COUNT(K.id_kamar)
alvaro.austin-> FROM KAMAR K, RAWAT_INAP RI
alvaro.austin-> WHERE RI.id_kamar = K.id_kamar
alvaro.austin-> GROUP BY K.jenis
alvaro.austin-> HAVING COUNT(K.id_kamar) >= 5;
  jenis  | count
-----+-----
Kelas 1 |      5
Kelas 2 |      9
VIP A    |     13
VIP B    |     18
(4 rows)

alvaro.austin=> █
```

2. [SQL] Tampilkan nama obat yang telah diresepkan oleh perawat wanita dengan menggunakan keyword **IN**.

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



```
alvaro.austin=> SELECT DISTINCT(OBAT>Nama)
alvaro.austin-> FROM PERAWAT, OBAT
alvaro.austin-> WHERE PERAWAT.jenis_kelamin = 'P' AND OBAT>Nama in (
alvaro.austin(> SELECT Obat>Nama
alvaro.austin(> FROM SHIFT_PERAWAT, PEMBERIAN_OBAT
alvaro.austin(> WHERE PERAWAT.Id_perawat = SHIFT_PERAWAT.Id_perawat AND SHIFT_PERAWAT.Id_shift_perawat = PEMBERIAN_OBAT.Id_shift_perawat AND PEMBERIAN_O
AT.Id_obat = OBAT.Id_obat
alvaro.austin(> );

                                nama
-----
Benzalkonium Chloride
Glipizide
ANTIPYRINE AND BENZOCAINE
Titanium Dioxide
sus scrofa small intestine mucosa lymph follicle
Salsalate
calcipotriene 0.005% and betamethasone dipropionate 0.064%
Benzethonium chloride
calcitonin salmon
Penicillium notatum
Estazolam
LOVASTATIN
Cefprozil
calcium carbonate
Albuterol Sulfate
Rivastigmine Tartrate
Nitrogen
RISPERIDONE
Codeine Phosphate, Phenylephrine Hydrochloride, Pyrilamine Maleate
HYDROCODONE BITARTRATE AND ACETAMINOPHEN
azithromycin
Natural Medicine
SILVER NITRATE,AVENA SATIVA FLOWERING TOP,TURNERA DIFFUSA LEAFY TWIG,LYCOPodium CLAVATUM SPORE,PHOSPHORIC ACID, SELENIUM.
Octinoxate, Titanium Dioxide, Zinc Oxide
desvenlafaxine succinate
Progesterone
Salicylic acid
Salicylic Acid
Methylprednisolone
(29 rows)
alvaro.austin=>
```

3. [SQL] Tampilkan nama UNIK obat yang **setidaknya** telah diberikan pada pasien.

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



```
alvaro.austin=> SELECT OBAT.Nama, OBAT.Id_obat
alvaro.austin-> FROM OBAT
alvaro.austin-> WHERE EXISTS (
alvaro.austin(> SELECT *
alvaro.austin(> FROM PEMBERIAN_OBAT
alvaro.austin(> WHERE PEMBERIAN_OBAT.Id_obat = OBAT.Id_obat
alvaro.austin(> );
```

nama	id_obat
calcitonin salmon	OB01
Estazolam	OB02
ANTIPYRINE AND BENZOCAINE	OB04
Phenobarbital	OB05
levofloxacin	OB06
Albuterol Sulfate	OB07
RISPERIDONE	OB08
Levothyroxine Sodium	OB09
calcipotriene 0.005% and betamethasone dipropionate 0.064%	OB10
sus scrofa small intestine mucosa lymph follicle	OB11
Benzalkonium Chloride	OB12
HYDROCODONE BITARTRATE AND ACETAMINOPHEN	OB13
Natural Medicine	OB14
desvenlafaxine succinate	OB15
Nitrogen	OB16
Treatment Set TS350256	OB17
Acetaminophen	OB18
Titanium Dioxide	OB19
Glipizide	OB20
Salsalate	OB21
Triclosan	OB22
Cefprozil	OB23
azithromycin	OB25
Monobasic Sodium Phosphate and Dibasic Sodium Phosphate	OB27
BENZOYL PEROXIDE	OB30
Nabumetone	OB31
OCTINOXATE, OCTISALATE	OB32
LOVASTATIN	OB33
Codeine Phosphate, Phenylephrine Hydrochloride, Pyrillamine Maleate	OB34
benzoyl peroxide	OB35
calcium carbonate	OB36
Methylprednisolone	OB37
Rivastigmine Tartrate	OB39
Penicillium notatum	OB40
TITANIUM DIOXIDE	OB41
pralatrexate	OB42
Salicylic Acid	OB43
Benzethonium chloride	OB44
Progesterone	OB45
SILVER NITRATE,AVENA SATIVA FLOWERING TOP,TURNERA DIFFUSA LEAFY TWIG,LYCOPODIUM CLAVATUM SPORE,PHOSPHORIC ACID, SELENIUM.	OB46
Menthol, Capsicum extract	OB47
Octinoxate, Titanium Dioxide, Zinc Oxide	OB48
Salicylic acid	OB49

(43 rows)

```
alvaro.austin=>
```

4. [SQL] Tampilkan daftar dokter yang tidak pernah merawat pasien rawat inap.

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

```
alvaro.austin=> (SELECT DOKTER.*
alvaro.austin(> FROM DOKTER)
alvaro.austin-> EXCEPT
alvaro.austin-> (SELECT DISTINCT DOKTER.*
alvaro.austin(> FROM DOKTER,DOKTER_RAWAT_INAP
alvaro.austin(> WHERE DOKTER_RAWAT_INAP.id_dokter = DOKTER.id_dokter);
id_dokter |      nama      | no_telp |      email      |      spesialisasi
-----|-----|-----|-----|-----
D007 | Ly Spellessy | 87114027540 | lspellessy6@blogspot.com | penyakit dalam
D040 | Tobias Trusse | 81023710802 | ttrussel13@sun.com | kulit dan kelamin
D050 | Ulrich Denisevich | 85685636571 | udenisevich1d@symantec.com | mata
D043 | Sigismond Stockbridge | 84252661633 | sstockbridge16@psu.edu | kulit dan kelamin
D032 | Cassandre Noddles | 81220215324 | cnoddlesv@edell.com | bedah
D013 | Meade Munn | 85430172820 | mmunnnc@si.edu | anak
D017 | Stewart Filkov | 81438198883 | sfilkovg@fastcompany.com | saraf
D018 | Nero Vanyard | 81684545781 | nvanyardh@technorati.com | saraf
D031 | Gearard Stapells | 85738280407 | gstapellsu@hc360.com | bedah
D037 | Lawry Galiford | 89124483409 | lgaliford10@sfgate.com | kulit dan kelamin
D048 | Lari Jobbings | 81309045282 | ljobbings1b@furl.net | mata
D019 | Shepperd Lichfield | 84437697707 | slichfieldi@tiny.cc | saraf
D008 | Den Stollsteimer | 85054750630 | dstollsteimer7@myspace.com | anak
D041 | Geoff Winthrop | 87843866302 | gwinthrop14@yelp.com | kulit dan kelamin
D021 | Lazaro Morston | 84529491412 | lmorstonk@nymag.com | saraf
D010 | Moreen Hebblethwaite | 83109586118 | mhebblethwaite9@buzzfeed.com | anak
D024 | Berty Normandale | 83088032291 | bnormandalen@list-manage.com | kandungan dan ginekologi
D005 | Frederique Pruce | 83447355115 | fpruce4@nps.gov | penyakit dalam
D015 | Rachael Redmire | 81414872968 | rredmiree@vimeo.com | saraf
(19 rows)

alvaro.austin=> █
```

5. [SQL] Tampilkan nama dokter dengan total jumlah pasien rawat inap yang telah ditugaskan kepada dokter tersebut diurutkan berdasarkan ascending alphabetical order (A-Z) dari namanya, tanpa peduli jika dokter tersebut memiliki pasien rawat inap atau tidak.

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

```
alvaro.austin(> (SELECT D.Nama AS nama_dokter, COUNT(*) AS jumlah_pasien
alvaro.austin(> FROM DOKTER D, DOKTER_RAWAT_INAP DRI
alvaro.austin(> WHERE D.id_dokter = DRI.ID_Dokter
alvaro.austin(> GROUP BY D.Nama)
alvaro.austin(>
alvaro.austin(> UNION
alvaro.austin(> (SELECT D.Nama AS nama_dokter, 0 AS jumlah_pasien
alvaro.austin(> FROM DOKTER D
alvaro.austin(> WHERE D.id_dokter not in (select DRI.ID_Dokter FROM DOKTER_RAWAT_INAP DRI))) AS DOKTER_DAN_PASIEN
alvaro.austin-> ORDER BY nama_dokter ASC;
nama_dokter      | jumlah_pasien
-----
Adams Savage      | 1
Barty Leebeter    | 2
Berty Normandale  | 0
Bev Pieracci      | 1
Brantley Ludlem   | 2
Candy Arminger    | 1
Carl Adelsberg    | 2
Cassandre Noddles | 0
Darbie Howle      | 1
Deana Gilhespy    | 2
Den Stollsteimer  | 0
Devlin Earney     | 1
Emery Brakespear  | 1
Fanechka Maryman  | 4
Frederique Puce   | 0
Garry Aspinell    | 1
Gearard Stapells  | 0
Genvieve Guillotin | 2
Geoff Winthrop    | 0
Gerrianna Mardall | 3
Halsey Ivashchenko | 1
Idalia Caughte    | 1
Jacinta Micco     | 2
Jenda Chaudret    | 5
Josephine Schnieder | 1
Karna Fehners     | 1
Klemens Braybrooks | 1
Lambert Wixey     | 2
Lari Jobbings     | 0
Lawry Galiford    | 0
Lazaro Morston    | 0
Lorry Jervois     | 2
Ly Spellessy      | 0
Malissa Richens   | 2
Meade Munn        | 0
Moreen Hebblethwaite | 0
Nero Vanyard      | 0
Rachael Redmire   | 0
Ronda Hargitt     | 1
Sephira Copsey    | 2
Shandy Stell      | 1
Shepperd Lichfield | 0
Sherry Fiveash    | 1
Sigismund Stockbridge | 0
Stewart Filkov    | 0
Tiena Klimmek     | 1
```

```
Tobias Trusse      | 0
Ulrich Denisevich  | 0
Vanny Brouwer      | 1
Zebulon Blacklawe  | 1
(50 rows)
alvaro.austin=> █
```

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



6. [SQL] Tampilkan jumlah pasien yang lahir pada setiap bulan. Anda disarankan menggunakan EXTRACT

```
alvaro.austin=> SELECT EXTRACT(MONTH FROM PASIEN.tgl_lahir), COUNT(*)
alvaro.austin-> FROM PASIEN
alvaro.austin-> GROUP BY EXTRACT(MONTH FROM PASIEN.tgl_lahir);
 date_part | count
-----+-----
          2 |      3
          3 |      8
          6 |      7
         10 |      3
         12 |      2
          5 |      7
          7 |      3
          9 |      2
          8 |      4
          4 |      5
         11 |      3
          1 |      2
(12 rows)
alvaro.austin=>
```

7. [SQL] Tampilkan **jenis kamar** dan **harga rata-rata** kamar untuk setiap jenis kamar.

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

```
alvaro.austin=> SELECT KAMAR.Jenis, AVG(KAMAR.Harga)
alvaro.austin-> FROM KAMAR
alvaro.austin-> GROUP BY KAMAR.Jenis;
  jenis |
-----+-----
Kelas 1 | 700000.000000000000
VVIP    | 3100000.000000000000
Kelas 2 | 300000.000000000000
VIP A   | 1600000.000000000000
Kelas 3 | 170000.000000000000
VIP B   | 1100000.000000000000
(6 rows)

alvaro.austin=>
```

8. [SQL] Tampilkan **nama pasien**, **id kamar**, dan **nama dokter** yang bertugas untuk setiap pasien yang masih dirawat di rumah sakit.

```
alvaro.austin=> (SELECT P>Nama as nama_pasien, RI.Id_kamar, STRING AGG(D>Nama, ', ') AS nama_dokter
alvaro.austin(> FROM PASIEN P, DOKTER D, DOKTER_RAWAT_INAP DRI, RAWAT_INAP RI
alvaro.austin(> WHERE D.Id_dokter = DRI.Id_dokter AND DRI.Id_rawat_inap = RI.Id_rawat_inap AND P.Id_pasien = RI.Id_pasien AND RI.tgl_keluar is null
alvaro.austin(> GROUP BY P>Nama, RI.Id_Kamar)
alvaro.austin-> UNION
alvaro.austin-> (SELECT P>Nama as nama_pasien, RI.Id_kamar, NULL AS nama_dokter
alvaro.austin(> FROM PASIEN P, RAWAT_INAP RI
alvaro.austin(> WHERE RI.Id_pasien = P.Id_pasien AND RI.id_rawat_inap NOT IN (SELECT id_rawat_inap FROM DOKTER_RAWAT_INAP) AND RI.tgl_keluar is NULL
alvaro.austin(> );
  nama_pasien | id_kamar | nama_dokter
-----+-----+-----
Barry Steljes | KA36    | Zebulon Blacklaw
Donni OHaire  | KA07    | Darbie Howle
Hamlen Ryles  | KA25    | Lambert Wixey
Lee Dot       | KA30    | Genvieve Guillotin, Brantley Ludlem
Virgil Clutterbuck | KA09    |
(5 rows)

alvaro.austin=>
```

9. [SQL] Tampilkan **tanggal masuk** dan **tanggal keluar** setiap pasien yang namanya tidak mengandung huruf E (tidak case sensitive).

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

```
alvaro.austin=> SELECT PASIEN>Nama, RAWAT_INAP.tgl_masuk, RAWAT_INAP.tgl_keluar
alvaro.austin-> FROM PASIEN, RAWAT_INAP
alvaro.austin-> WHERE PASIEN.Id_pasien = RAWAT_INAP.Id_pasien AND PASIEN>Nama NOT LIKE '%e%' AND PASIEN>Nama NOT LIKE '%E%';
      nama      | tgl_masuk | tgl_keluar
-----+-----+-----
 Archibaldo Wanklyn | 2022-05-27 | 2022-05-31
 Archibaldo Wanklyn | 2020-03-01 | 2020-03-22
   Win Piggford    | 2020-08-03 | 2020-08-29
   Thorny Dubbin    | 2022-03-29 | 2022-04-04
 Orson MacGiffin   | 2021-12-13 | 2021-12-31
   Saloma Jacob    | 2020-09-10 | 2020-09-17
(6 rows)

alvaro.austin=>
```

10. [SQL] Tampilkan **nama** dan **nomor telepon** pasien yang tidak pernah menjadi pasien rawat inap.

“Tidak ada Fredra Addey karena di delete pada lab kemarin”

```
alvaro.austin=> (SELECT PASIEN>Nama, PASIEN.no_telp
alvaro.austin(> FROM PASIEN)
alvaro.austin-> EXCEPT
alvaro.austin-> (SELECT PASIEN>Nama, PASIEN.no_telp
alvaro.austin(> FROM PASIEN, RAWAT_INAP
alvaro.austin(> WHERE PASIEN.Id_pasien = RAWAT_INAP.Id_pasien
alvaro.austin(> );
      nama      | no_telp
-----+-----
 Sibella Raddin  | 81800084683
 Tilly Wallwork  | 85620409570
 Fielding Crone  | 85111343784
 Garold Onslow   | 81141510588
 Lynette Conwell | 89434813042
 Judy Phillip    | 81589924398
 Ashlie Elloy    | 81039808440
 Zaccaria Dewsnap | 84860192729
 Andrew Ginnell  | 81546125433
 Haywood Randles | 82242025651
 Bobby Elner     | 86434766742
 Friedrich Pina  | 82880371279
 Genna Mechem    | 82730586822
(13 rows)

alvaro.austin=>
```

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

11. [SQL] Tampilkan daftar pasien yang pernah menjadi pasien rawat inap di kamar jenis VIP atau VVIP. Anda harus menggunakan keyword **UNION**.

```
alvaro.austin=> (SELECT PASIEN.*
alvaro.austin(> FROM PASIEN, RAWAT_INAP, KAMAR
alvaro.austin(> WHERE PASIEN.id_pasien = RAWAT_INAP.id_pasien and RAWAT_INAP.id_kamar = KAMAR.id_kamar AND KAMAR.jenis LIKE 'VIP%')
alvaro.austin=> UNION
alvaro.austin=> (SELECT PASIEN.*
alvaro.austin(> FROM PASIEN, RAWAT_INAP, KAMAR
alvaro.austin(> WHERE PASIEN.id_pasien = RAWAT_INAP.id_pasien and RAWAT_INAP.id_kamar = KAMAR.id_kamar AND KAMAR.jenis LIKE 'VVIP%')
alvaro.austin=> ;
```

id_pasien	nama	alamat	no_telp	email	tgl_lahir	jenis_kelamin
PA01	Rey Ferrieri	Jl Raya PKP, Dki Jakarta	87667076022	rferrieri0@latimes.com	1960-07-29	L
PA02	Englebert Glassard	Jl. Pluit Raya No. 19, Jakarta Raya	83613307057	eglassard18imdb.com	1972-08-09	L
PA05	Cesar Witard	Plaza City View, Building	83060150583	cwitard4@ca.gov	1969-03-12	L
PA06	Jayne Clew	Jl Bisma Raya Bl D-1/22, Dki Jakarta	82731096705	jclew5@exblog.jp	1982-08-18	L
PA07	Rodney Di Biasio	Jl Jatinegara Brt 122, Dki Jakarta	86987105340	rdi6@amazon.de	1966-06-27	L
PA09	Gilemette Gresswell	Jl Adam 103 Sukabumi Utara, Dki Jakarta	86224127948	ggresswell18@amazon.co.uk	1981-05-27	P
PA11	Padraic De Francisci	Jl H Agus Salim 69, Jakarta	87388709917	pdea@slideshare.net	1994-06-03	P
PA14	Clarisse Pennuzzi	Jl Rawagelam IV 7, Dki Jakarta	86444824850	cpennuzziid@pbs.org	1963-04-24	P
PA20	Charlot Huffadine	Jl Mangga Dua Raya Psr Pagi Mangga Dua Bl A/188, Dki Jakarta	81254024034	chuffadine@ebay.com	1972-11-22	L
PA22	Lee Dot	Jl Pembangunan II 15 A/1, Dki Jakarta	89774964683	ldot1@etsy.com	2002-12-18	L
PA25	Hamlen Ryles	Jl Jend A Yani 286 Ged Graha Pangeran Unit 7/C-1 7th Fl, Jawa Timur	89078717955	hryleso@exblog.jp	1995-10-15	P
PA27	Archibaldo Wanklyn	Jl Jelambar Jaya II 8, Dki Jakarta	87948788158	awanklynq@prweb.com	1968-07-21	P
PA28	Stewart Mustard	Jl KH Hasyim Ashari 33-O, Dki Jakarta	88177679007	smustardr@phbb.com	1964-01-10	L
PA29	Bevin Carlinos	Jl Jelambar Brt III/20, Dki Jakarta	87072282157	bcarlinos@paypal.com	1960-05-15	L
PA30	Orson MacGiffin	Per Kecip 12, Dki Jakarta	83721927052	omacgiffin@marriott.com	1977-04-13	P
PA32	Barry Steljes	Jl Kramat Raya 148 F, Dki Jakarta	86378562628	bsteljesv@bigcartel.com	1984-03-28	L
PA33	Meade Redington	Jl Sunan Muria 7 RT 007/015, Dki Jakarta	83145220454	mredingtonw@wikipedia.org	1992-02-15	L
PA34	Maitilde Cundey	Jl Melayu Besar 68, Dki Jakarta	89473689365	mcundeyx@xrea.com	1996-04-06	P
PA36	Karel Stoggeil	Jl Batu Tulis 16, Dki Jakarta	85872264445	kstoggeilz@globo.com	1969-05-04	L
PA40	Yard Fiddler	Jl Cengkoh 16/17, Dki Jakarta	81354446300	yfiddler13@delicious.com	1967-10-10	L
PA42	Fiona Passmore	Jl Tenggilir 2, Dki Jakarta	81858210291	fpassmore15@gmpg.org	1970-11-19	P
PA44	Elena Badgers	Plaza Ciputat Mas Bl C/5 A Kav CDEF, Dki Jakarta	82734861577	ebadgers17@mit.edu	1994-03-15	P
PA45	Welsh Landeaux	Jl Krekot Jaya Bl E/3, Dki Jakarta	81289839793	wlandeaux18@google.de	1974-03-03	L
PA46	Win Piggford	Jl Muara Baru Ujung Bl S, Jakarta	81394260350	wpiggford19@addtoany.com	1967-05-26	P
PA47	Raddy Kizzena	Jl Suryopranoto 2 Kompl Harmoni Plaza Bl E/28, Dki Jakarta	89925774888	rkizzena@live.com	1982-06-13	P
PA48	Tracey Varey	Jl Bekasi Tmr II 30 RT 015/04, Dki Jakarta	81307584054	tvarey1b@arizona.edu	1971-10-17	P
PA49	Morie Davisson	Jl Kuningan Brt 56, Dki Jakarta	88928959829	mdavisson1c@census.gov	1979-04-17	P

```
(27 rows)
alvaro.austin=>
```

12. [SQL] Tampilkan **nama** dan **jenis kelamin** perawat yang merawat semua pasien wanita.

```
alvaro.austin=> SELECT nama, jenis_kelamin
alvaro.austin=> FROM PERAWAT
alvaro.austin=> WHERE id_perawat in (SELECT id_perawat FROM SHIFT_PERAWAT)
alvaro.austin=> AND id_perawat not in (SELECT DISTINCT P.id_perawat FROM SHIFT_PERAWAT SP, PERAWAT P, RAWAT_INAP RI WHERE SP.id_rawat_inap = RI.id_rawat_inap
alvaro.austin(> AND SP.id_perawat = P.id_perawat AND RI.id_pasien = ANY (SELECT id_pasien from PASIEN WHERE 'jenis_kelamin' = 'L')));
```

nama	jenis_kelamin
Jeanna Basilone	L
Stanwood Redhole	P
Brannon Sebring	L
Clarabelle Gong	L
Clea Ebdin	L
Dalis Rummings	L
Kelcy Fraschini	L
Lizette Rodolico	P
Teodor Schlag	P
Thorndike Quiddinton	L
Andrei Sandland	L

```
(11 rows)
alvaro.austin=>
```

13. [Trivia] Apakah kita mungkin mendapatkan data tertentu dari operasi inner join menggunakan keyword **IN**?

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



Benar, kita dapat menggunakan keyword `in` pada operasi inner seperti apabila kita ingin menggabungkan data dengan syarat ada suatu data tersebut pada subquery yang kita inginkan. Contoh kita dapat memuat sebuah relasi `DOKTER` dan `DOKTER_RAWAT_INAP` dengan melakukan operasi inner join. Apabila kita memakai `IN` terhadap `id_dokter` pada relasi `DOKTER` dan `DOKTER_RAWAT_INAP`, maka proses inner join akan mirip seperti cross join dimana setiap pasangan pada join akan masuk kepada sub query `in` dan selalu menghasilkan nilai `True`.

Latihan Soal 2

1. [SQL] Jalankan SQL Query pada Contoh 27 hingga Contoh 35 di atas dan cantumkan hasilnya pada laporan.

- Screenshot contoh 27

```
alvaro.austin=> CREATE VIEW daftar_dokter AS  
alvaro.austin-> SELECT id_dokter, nama  
alvaro.austin-> FROM dokter;  
CREATE VIEW  
alvaro.austin=> █
```

- Screenshot contoh 28

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



UNIVERSITAS
INDONESIA
Veritas, Probitas, Justitia

FACULTY OF
COMPUTER
SCIENCE

```
alvaro.austin=> SELECT * FROM daftar_dokter;
id_dokter |      nama
-----+-----
D001      | Lorry Jervois
D002      | Brantley Ludlem
D003      | Panchika Mayman
D004      | Ronda Hargitt
D005      | Frederique Puce
D006      | Barry Leebeter
D007      | Ly Spellasy
D008      | Den Stollsteimer
D009      | Jenda Chaudret
D010      | Warren Hobbethwaite
D011      | Idalia Caughte
D012      | Darbie Howle
D013      | Meade Muna
D014      | Josephine Schnieder
D015      | Rachael Redmire
D016      | Sherry Fivash
D017      | Stewart Filkov
D018      | Nero Vanyard
D019      | Shepperd Lichfield
D020      | Candy Atminger
D021      | Lazaro Morston
D022      | Klemens Braybrooks
D023      | Shandy Stell
D024      | Bertie Normandale
D025      | Jacinta Micco
D026      | Malissa Richens
D027      | Emery Brakespear
D028      | Halsey Ivashchenko
D029      | Devlin Earney
D030      | Carl Adelabery
D031      | Gearard Stapells
D032      | Cassandra Noddles
D033      | Genevieve Guillotin
D034      | Tiara Klimmek
D035      | Garry Aspinell
D036      | Gerianna Mardall
D037      | Lawry Galford
D038      | Bev Pieracci
D039      | Zebulon Blacklaw
D040      | Tobias Truse
D041      | Geoff Winthrop
D042      | Adams Savage
D043      | Sigmund Stockbridge
D044      | Erika Fehner
D045      | Lambert Wixey
D046      | Vanny Brouwer
D047      | Sephira Copsey
D048      | Lari Jobbings
D049      | Deana Gilheapy
D050      | Ulrich Denisovich
(50 rows)

alvaro.austin=>
```

- Screenshot contoh 29

```
alvaro.austin=> DROP VIEW daftar_dokter;
DROP VIEW
alvaro.austin=>
```

- Screenshot contoh 30

```
alvaro.austin=> CREATE INDEX index_jenis_kamar ON kamar(jenis);
CREATE INDEX
```

- Screenshot contoh 31

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



```
alvaro.austin=> CREATE INDEX index_nama_pasien  
alvaro.austin-> ON pasien USING HASH (nama);  
CREATE INDEX
```

- Screenshot contoh 32

```
alvaro.austin=> CREATE INDEX index_perawat  
alvaro.austin-> ON perawat (id_perawat, nama DESC);  
CREATE INDEX
```

- Screenshot contoh 33

```
alvaro.austin=> CREATE INDEX index_nama_dokter  
alvaro.austin-> ON dokter (nama);  
CREATE INDEX  
alvaro.austin=>  
alvaro.austin=> CREATE INDEX index_spesialisasi_dokte  
alvaro.austin-> ON dokter (spesialisasi);  
CREATE INDEX  
alvaro.austin=> █
```

- Screenshot contoh 34

```
alvaro.austin=> DROP INDEX index_nama_dokter;  
DROP INDEX  
alvaro.austin=> █
```

- Screenshot contoh 35

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



```
alvaro.austin=> EXPLAIN ANALYZE
alvaro.austin-> SELECT *
alvaro.austin-> FROM PASIEN
alvaro.austin-> WHERE jenis_kelamin = 'L';
                                QUERY PLAN
-----
Seq Scan on pasien  (cost=0.00..1.61 rows=24 width=95) (actual time=0.011..0.024 rows=24 loops=1)
  Filter: (jenis_kelamin = 'L'::bpchar)
  Rows Removed by Filter: 25
Planning time: 0.505 ms
Execution time: 0.055 ms
(5 rows)

alvaro.austin=> █
```

2. View

- a. [Trivia] Apa yang akan terjadi jika kita membuat View menggunakan nama yang sama dengan nama tabel yang ada pada database? Jelaskan!

Akan terjadi error dimana error tersebut mengatakan bahwa relasi sudah ada. Hal ini karena view sendiri membuat relasi yang bersifat *unmaterialized* atau tidak terbuat secara nyata pada database.

- b. [Trivia] Apa fungsi TEMP atau TEMPORARY di View?

Fungsi TEMP atau TEMPORARY di View adalah membuat sebuah view yang bersifat sementara atau *temporary* yang artinya view ini akan di drop setelah session selesai.

- c. [SQL] Buatlah View yang menyimpan **nama** beserta **durasi bekerja** yang dilakukan oleh perawat.

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



```
alvaro.austin=> CREATE VIEW perawat_durasi AS
alvaro.austin-> SELECT P>Nama, SUM(SP.waktu_akhir - SP.waktu_mulai) AS durasi_bekerja
alvaro.austin-> FROM PERAWAT P, SHIFT_PERAWAT SP
alvaro.austin-> WHERE P.id_perawat = SP.id_perawat
alvaro.austin-> GROUP BY P>Nama;
CREATE VIEW
alvaro.austin=> SELECT P>Nama, SUM(SP.waktu_akhir - SP.waktu_mulai) AS durasi_bekerja
alvaro.austin-> FROM PERAWAT P, SHIFT_PERAWAT SP
alvaro.austin-> WHERE P.id_perawat = SP.id_perawat
alvaro.austin-> GROUP BY P>Nama;
```

- d. [SQL] Buatlah View yang menyimpan **nama-nama perawat** yang merawat pasien dikelompokkan berdasarkan id pasien dan **nama pasien** yang belum keluar dari rumah sakit (HINT: string_agg)

```
alvaro.austin=> CREATE VIEW PASIEN_PERAWAT_MASIH_DIRAWAT AS
alvaro.austin-> SELECT STRING_AGG(PERAWAT>Nama, ' ') AS nama_perawat
alvaro.austin-> FROM PERAWAT, PASIEN, SHIFT_PERAWAT, RAWAT_INAP
alvaro.austin-> WHERE PERAWAT.id_perawat = SHIFT_PERAWAT.id_perawat AND SHIFT_PERAWAT.id_rawat_inap = RAWAT_INAP.id_rawat_inap AND PASIEN.id_pasien = RAWAT_INAP.id_pasien AND RAWAT_INAP.tgl_keluar IS NULL
alvaro.austin-> GROUP BY PASIEN>Nama, PASIEN.Id_pasien;
CREATE VIEW
alvaro.austin=>
```

3. Indexing and Analyze

Diberikan query berikut.

```
SELECT * FROM kamar ORDER BY harga DESC;
```

```
SELECT * FROM rawat_inap WHERE tgl_keluar IS NULL;
```

```
SELECT * FROM perawat WHERE nama LIKE 'T%';
```

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



```
SELECT * FROM pasien ORDER BY alamat LIMIT 10;
```

- a. [SQL] Jalankan perintah EXPLAIN ANALYZE untuk setiap *query* di atas. *Screenshot* eksekusinya dan tulis hasilnya pada tabel di bawah, sertakan dalam laporan submisi Anda.

- Query 1

```
alvaro.austin=> EXPLAIN ANALYZE
alvaro.austin-> SELECT * FROM kamar ORDER BY harga DESC;
                        QUERY PLAN
-----
Sort  (cost=2.69..2.80 rows=45 width=160) (actual time=0.126..0.134 rows=45 loops=1)
  Sort Key: harga
  Sort Method: quicksort  Memory: 20kB
  -> Seq Scan on kamar  (cost=0.00..1.45 rows=45 width=160) (actual time=0.050..0.060 rows=45 loops=1)
Planning time: 0.489 ms
Execution time: 0.184 ms
(6 rows)

alvaro.austin=> █
```

- Query2

```
alvaro.austin=> EXPLAIN ANALYZE
alvaro.austin-> SELECT * FROM rawat_inap WHERE tgl_keluar IS NULL;
                        QUERY PLAN
-----
Seq Scan on rawat_inap  (cost=0.00..15.30 rows=3 width=122) (actual time=0.024..0.030 rows=5 loops=1)
  Filter: (tgl_keluar IS NULL)
  Rows Removed by Filter: 45
Planning time: 0.054 ms
Execution time: 0.053 ms
(5 rows)

alvaro.austin=> █
```

- Query 3

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



```
alvaro.austin=> EXPLAIN ANALYZE
alvaro.austin-> SELECT * FROM perawat WHERE nama LIKE 'T%';
                                QUERY PLAN
-----
Seq Scan on perawat  (cost=0.00..1.62 rows=3 width=56) (actual time=0.020..0.027 rows=3 loops=1)
  Filter: ((nama)::text ~~ 'T% '::text)
  Rows Removed by Filter: 47
  Planning time: 0.232 ms
  Execution time: 0.049 ms
(5 rows)

alvaro.austin=> █
```

- Query 4

```
alvaro.austin=> EXPLAIN ANALYZE
alvaro.austin-> SELECT * FROM pasien ORDER BY alamat LIMIT 10;
                                QUERY PLAN
-----
Limit  (cost=2.55..2.57 rows=10 width=95) (actual time=0.186..0.193 rows=10 loops=1)
  -> Sort (cost=2.55..2.67 rows=49 width=95) (actual time=0.184..0.187 rows=10 loops=1)
        Sort Key: alamat
        Sort Method: top-N heapsort  Memory: 18kB
        -> Seq Scan on pasien  (cost=0.00..1.49 rows=49 width=95) (actual time=0.038..0.048 rows=49 loops=1)
  Planning time: 0.158 ms
  Execution time: 0.220 ms
(7 rows)

alvaro.austin=> █
```

b. [SQL] Buat *index* berikut (*method* nya terserah Anda):

- index_harga_kamar pada tabel **KAMAR** kolom **harga**.
- index_tgl_keluar_rawat_inap pada tabel **RAWAT_INAP** kolom **tgl_keluar**.
- index_nama_perawat pada tabel **PERAWAT** kolom **nama**.
- index_alamat_pasien pada tabel **PASIEN** kolom **alamat**.

- Query 1

```
alvaro.austin=> create index index_harga_kamar on kamar(Harga);
CREATE INDEX
alvaro.austin=> █
```

- Query 2

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



```
alvaro.austin=> create index index_tgl_keluar_rawat_inap on rawat_inap(tgl_keluar);
CREATE INDEX
alvaro.austin=>
```

- Query 3

```
alvaro.austin=> CREATE INDEX index_nama_perawat ON PERAWAT(nama);
CREATE INDEX
alvaro.austin=>
```

- Query 4

```
alvaro.austin=> create index index_alamat_pasien on PASIEN(alamat);
CREATE INDEX
alvaro.austin=>
```

Tampilkan *query* dan *index* untuk setiap tabel di dalam laporan submisi Anda.

- c. [SQL] Jalankan kembali **setiap query SELECT** di atas dari pertanyaan nomor 3 menggunakan perintah **EXPLAIN ANALYZE**. *Screenshot* eksekusinya dan tulis hasilnya pada tabel di bawah, sertakan dalam laporan submisi Anda.

- Query 1

```
alvaro.austin=> EXPLAIN ANALYZE SELECT * FROM kamar ORDER BY harga DESC;
QUERY PLAN
-----
Sort  (cost=2.69..2.80 rows=45 width=160) (actual time=0.053..0.061 rows=45 loops=1)
  Sort Key: harga
  Sort Method: quicksort  Memory: 20kB
-> Seq Scan on kamar  (cost=0.00..1.45 rows=45 width=160) (actual time=0.015..0.023 rows=45 loops=1)
Planning time: 0.087 ms
Execution time: 0.096 ms
(6 rows)

alvaro.austin=>
```

Laporan Tutorial Lab 2

Basis Data

Semester Ganjil 2022/2023



- Query 2

```
alvaro.austin=> EXPLAIN ANALYZE SELECT * FROM rawat_inap WHERE tgl_keluar IS NULL;
                    QUERY PLAN
-----
Seq Scan on rawat_inap  (cost=0.00..1.50 rows=1 width=122) (actual time=0.010..0.016 rows=5 loops=1)
  Filter: (tgl_keluar IS NULL)
  Rows Removed by Filter: 45
Planning time: 0.064 ms
Execution time: 0.037 ms
(5 rows)

alvaro.austin=>
```

- Query 3

```
alvaro.austin=> EXPLAIN ANALYZE SELECT * FROM perawat WHERE nama LIKE 'T%';
                    QUERY PLAN
-----
Seq Scan on perawat  (cost=0.00..1.62 rows=3 width=56) (actual time=0.014..0.021 rows=3 loops=1)
  Filter: ((nama)::text ~~ 'T%':text)
  Rows Removed by Filter: 47
Planning time: 0.140 ms
Execution time: 0.043 ms
(5 rows)
```

- Query 4

```
alvaro.austin=> EXPLAIN ANALYZE SELECT * FROM pasien ORDER BY alamat LIMIT 10;
                    QUERY PLAN
-----
Limit  (cost=2.55..2.57 rows=10 width=95) (actual time=0.147..0.152 rows=10 loops=1)
-> Sort  (cost=2.55..2.67 rows=49 width=95) (actual time=0.138..0.140 rows=10 loops=1)
   Sort Key: alamat
   Sort Method: top-N heapsort  Memory: 18kB
-> Seq Scan on pasien  (cost=0.00..1.49 rows=49 width=95) (actual time=0.008..0.023 rows=49 loops=1)
Planning time: 0.121 ms
Execution time: 0.182 ms
(7 rows)

alvaro.austin=>
```

- d. [Trivia] Bandingkan *planning time* dan *execution time* (menggunakan tabel di bawah) dari *query* saat tanpa index dan setelah menggunakan *index*. Mana yang lebih baik? Berikan penjelasan!

	Planning Time	Execution Time
Query		

Laporan Tutorial Lab 2
Basis Data
Semester Ganjil 2022/2023



	TANPA INDEX	DENGAN INDEX	TANPA INDEX	DENGAN INDEX
1	0.489ms	0.087ms	0.184ms	0.096ms
2	0.054ms	0.064ms	0.053ms	0.037ms
3	0.232ms	0.140ms	0.049ms	0.043ms
4	0.158ms	0.121ms	0.220ms	0.182ms

Secara keseluruhan, index biasanya akan mempercepat proses execution time dari query yang kita miliki. Dari data yang saya miliki, apabila menggunakan hash maka proses sorting akan menggunakan heap sort yang biasanya memiliki kompleksitas dalam Big-O sebesar $O(n \log n)$ sedangkan apabila tanpa penjelasan method (binary tree) akan menggunakan quick sort. Untuk kesimpulan, terlihat dari query 1, 2, 3, dan 4 bahwa indexing dapat mempercepat proses pengambilan query kita. Oleh karena itu indexing akan sangat berguna untuk mempercepat pengambilan query.