Nama : Ahmad Yasir

NPM : 18630171

PRAKTIKUM 211-BASIS DATA 7 - C

PERTEMUAN 2

1. Modifikasi Field
   A. ADD

```
1  ALTER TABLE lokasi ADD alamat VARCHAR(225) NOT NULL AFTER nama_lokasi;
```

#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default	Comment	Collation
1	id	INT	11	☐	☐	☐	AUTO_INCREME...		
2	nama_lokasi	VARCHAR	255	☐	☐	☐	No default		latin1_swedish_ci
3	alamat	VARCHAR	225	☐	☐	☐	No default		latin1_swedish_ci

   B. CHANGE

```
1  ALTER TABLE lokasi CHANGE alamat alamat_gedung TEXT NOT NULL;
```

#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default	Comment	Collation
1	id	INT	11	☐	☐	☐	AUTO_INCREME...		
2	nama_lokasi	VARCHAR	255	☐	☐	☐	No default		latin1_swedish_ci
3	alamat_gedung	TEXT		☐	☐	☐	No default		latin1_swedish_ci

   C. DROP

```
1  ALTER TABLE lokasi DROP alamat_gedung;
```

#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default	Comment	Collation
1	id	INT	11	☐	☐	☐	AUTO_INCREME...		
2	nama_lokasi	VARCHAR	255	☐	☐	☐	No default		latin1_swedish_ci

   D. Select & Where

```
1  SELECT * FROM lokasi WHERE id = 2;
```

lokasi (1r × 2c)

id	nama_lokasi
2	Kota Banjarbaru

   E. Wildcard & Like

```
1  SELECT * FROM lokasi WHERE nama_lokasi LIKE '%kota%';
```

lokasi (2r × 2c)

id	nama_lokasi
1	Kota Banjarmasin
2	Kota Banjarbaru

F. Latihan 2.1

```
1  SELECT * FROM jabatan WHERE nama_jabatan LIKE '%programmer%';
```

jabatan (2r × 5c)

id	nama_jabatan	gapok_jabatan	tunjangan_jabatan	uang_makan_perhari
3	Senior Programmer	2,200,000	400,000	36,000
4	Junior Programmer	2,100,000	350,000	34,000

2. Numeric

```
1  SELECT * FROM jabatan;
```

jabatan (5r × 5c)

id	nama_jabatan	gapok_jabatan	tunjangan_jabatan	uang_makan_perhari
1	System Analyst	2,400,000	500,000	40,000
2	Project Manager	2,300,000	450,000	38,000
3	Senior Programmer	2,200,000	400,000	36,000
4	Junior Programmer	2,100,000	350,000	34,000
5	Magang	1,000,000	100,000	20,000

```
1  SELECT * FROM jabatan WHERE gapok_jabatan > 2200000;
```

jabatan (2r × 5c)

id	nama_jabatan	gapok_jabatan	tunjangan_jabatan	uang_makan_perhari
1	System Analyst	2,400,000	500,000	40,000
2	Project Manager	2,300,000	450,000	38,000

3. Latihan 2.2

```
1  SELECT * FROM jabatan WHERE uang_makan_perhari < 35000;
```

jabatan (2r × 5c)

id	nama_jabatan	gapok_jabatan	tunjangan_jabatan	uang_makan_perhari
4	Junior Programmer	2,100,000	350,000	34,000
5	Magang	1,000,000	100,000	20,000

4. Function

```
1  SELECT * FROM pengguna;
```

pengguna (8r × 5c)

id	username	password	peran	login_terakhir
1	admin	21232f297a57a5a743894a0e4a801fc3	ADMIN	(NULL)
2	user	ee11cbb19052e40b07aac0ca060c23ee	USER	(NULL)
3	johndoe	6579e96f76baa00787a28653876c6127	USER	(NULL)
4	fulanbinfulan	7c232f58f03f6d00c8d28166fbff7a8b	USER	(NULL)
5	mawar	bd117502364227fd8c09098d31e11313	USER	(NULL)
6	melati	27e80ebc907bd57004986be9e6f2dd83	USER	(NULL)
7	dahlia	ee11cbb19052e40b07aac0ca060c23ee	USER	(NULL)
8	lily	89f288757f4d0693c99b007855fc075e	USER	(NULL)

```
1  SELECT * FROM pengguna WHERE username = 'admin' AND PASSWORD = MD5('admin');
```

pengguna (1r × 5c)

id	username	password	peran	login_terakhir
1	admin	21232f297a57a5a743894a0e4a801fc3	ADMIN	(NULL)

```
1  ELECT id, nama_lengkap, tanggal_masuk, TIMESTAMPDIFF(YEAR, tanggal_masuk, CURDATE()) masa_kerja_tahun FROM karyawan;
```

karyawan (8r × 4c)

id	nama_lengkap	tanggal_masuk	masa_kerja_tahun
1	Admin	2011-01-01	10
2	Tes User	2012-02-02	9
3	John Doe	2013-03-03	8
4	Fulan Bin Fulan	2014-04-04	7
5	Mawar Kurniani	2015-05-05	6
6	Melati Rahmawati	2016-06-06	5
7	Dahlia Setiani	2017-07-07	4
8	Lily Handayani	2018-08-08	3

5. Select & Count

```
1  SELECT COUNT(*) kuantitas_lokasi FROM lokasi WHERE nama_lokasi LIKE '%kota%';
```

lokasi (1r × 1c)

kuantitas_lokasi
2

6. Latihan 2.3

```
1  SELECT COUNT(*) kuantitas_jabatan FROM jabatan WHERE nama_jabatan LIKE '%programmer%';
```

jabatan (1r × 1c)

kuantitas_jabatan
2

Pertemuan 3

1. One To Many

```
1  SELECT *FROM bagian;
```

bagian (3r × 4c)

id	nama_bagian	karyawan_id	lokasi_id
1	Autentikasi	5	1
2	Data Science	3	1
3	Backend Developer	6	2

```
1  SELECT b.*, L.nama_lokasi lokasi_bagian FROM bagian B
2  INNER JOIN lokasi L ON B.lokasi_id = L.id
```

bagian (3r × 5c)

id	nama_bagian	karyawan_id	lokasi_id	lokasi_bagian
1	Autentikasi	5	1	Kota Banjarmasin
2	Data Science	3	1	Kota Banjarmasin
3	Backend Developer	6	2	Kota Banjarbaru

2. Latihan 3.1

```
1  SELECT B.*, K.nama_lengkap nama_kepala_bagian, L.nama_lokasi lokasi_bagian
2  FROM bagian B
3  INNER JOIN karyawan K ON B.karyawan_id = K.id
4  INNER JOIN lokasi L ON B.lokasi_id = L.id
```

bagian (3r × 6c)

id	nama_bagian	karyawan_id	lokasi_id	nama_kepala_bagian	lokasi_bagian
1	Autentikasi	5	1	Mawar Kurniani	Kota Banjarmasin
2	Data Science	3	1	John Doe	Kota Banjarmasin
3	Backend Developer	6	2	Melati Rahmawati	Kota Banjarbaru

3. Many To Many

```
1  SELECT * FROM jabatan_karyawan;
```

jabatan_karyawan (17r × 4c)

id	jabatan_id	karyawan_id	tanggal_mulai
1	5	3	2013-03-03
2	4	3	2014-04-01
3	5	4	2014-04-04
4	3	3	2015-05-04
5	4	4	2015-05-05
6	5	5	2015-05-05
7	2	3	2015-06-01
8	3	4	2015-06-02
9	4	5	2015-06-03
10	5	6	2015-06-06
11	1	3	2017-07-01
12	2	4	2017-07-02
13	3	5	2017-07-02
14	4	6	2017-07-02
15	5	7	2017-07-07
16	4	7	2018-08-02
17	5	8	2018-08-08

4. Latihan 3.2

```
SELECT JK.*, J.nama_jabatan nama_jabatan,N.nama_lengkap nama_lengkap FROM jabatan_karyawan JK
INNER JOIN karyawan N ON JK.karyawan_id = N.id
INNER JOIN jabatan J ON JK.jabatan_id = J.id
```

jabatan_karyawan (17r × 6c)

id	jabatan_id	karyawan_id	tanggal_mulai	nama_jabatan	nama_lengkap
1	5	3	2013-03-03	Magang	John Doe
2	4	3	2014-04-01	Junior Programmer	John Doe
3	5	4	2014-04-04	Magang	Fulan Bin Fulan
4	3	3	2015-05-04	Senior Programmer	John Doe
5	4	4	2015-05-05	Junior Programmer	Fulan Bin Fulan
6	5	5	2015-05-05	Magang	Mawar Kurniani
7	2	3	2015-06-01	Project Manager	John Doe
8	3	4	2015-06-02	Senior Programmer	Fulan Bin Fulan
9	4	5	2015-06-03	Junior Programmer	Mawar Kurniani
10	5	6	2015-06-06	Magang	Melati Rahmawati
11	1	3	2017-07-01	System Analyst	John Doe
12	2	4	2017-07-02	Project Manager	Fulan Bin Fulan
13	3	5	2017-07-02	Senior Programmer	Mawar Kurniani
14	4	6	2017-07-02	Junior Programmer	Melati Rahmawati
15	5	7	2017-07-07	Magang	Dahlia Setiani
16	4	7	2018-08-02	Junior Programmer	Dahlia Setiani
17	5	8	2018-08-08	Magang	Lily Handayani

5. Menampilkan data riwayat jabatan per karyawan

```
SELECT K.id, K.nama_lengkap, J.nama_jabatan,
JK.tanggal_mulai
FROM karyawan K
INNER JOIN jabatan_karyawan JK ON K.id = JK.karyawan_id
INNER JOIN jabatan J ON J.id = JK.jabatan_id
WHERE K.id = 3 ORDER BY JK.tanggal_mulai DESC
```

karyawan (5r × 4c)

id	nama_lengkap	nama_jabatan	tanggal_mulai
3	John Doe	System Analyst	2017-07-01
3	John Doe	Project Manager	2015-06-01
3	John Doe	Senior Programmer	2015-05-04
3	John Doe	Junior Programmer	2014-04-01
3	John Doe	Magang	2013-03-03

6. Menampilkan data karyawan beserta jabatan

```
SELECT K.id, K.nama_lengkap, J.nama_jabatan, JK.tanggal_mulai
FROM karyawan K
INNER JOIN jabatan_karyawan JK ON K.id = JK.karyawan_id
INNER JOIN jabatan J ON J.id = Jk.jabatan_id
ORDER BY K.id
```

karyawan (17r × 4c)

id	nama_lengkap	nama_jabatan	tanggal_mulai
3	John Doe	System Analyst	2017-07-01
3	John Doe	Magang	2013-03-03
3	John Doe	Junior Programmer	2014-04-01
3	John Doe	Senior Programmer	2015-05-04
3	John Doe	Project Manager	2015-06-01
4	Fulan Bin Fulan	Project Manager	2017-07-02
4	Fulan Bin Fulan	Magang	2014-04-04
4	Fulan Bin Fulan	Junior Programmer	2015-05-05
4	Fulan Bin Fulan	Senior Programmer	2015-06-02
5	Mawar Kurniani	Senior Programmer	2017-07-02
5	Mawar Kurniani	Magang	2015-05-05
5	Mawar Kurniani	Junior Programmer	2015-06-03
6	Melati Rahmawati	Junior Programmer	2017-07-02
6	Melati Rahmawati	Magang	2015-06-06
7	Dahlia Setiani	Magang	2017-07-07
7	Dahlia Setiani	Junior Programmer	2018-08-02
8	Lily Handayani	Magang	2018-08-08

7. Menampilkan data karyawan dan jabatan terakhir

```
SELECT K.id, K.nama_lengkap,
(
   SELECT J.nama_jabatan FROM jabatan_karyawan JK
   INNER JOIN jabatan J ON JK.jabatan_id = J.id
   WHERE karyawan_id = K.id ORDER BY JK.tanggal_mulai DESC LIMIT 1
) jabatan_terakhir
FROM karyawan K
```

jabatan_karyawan (8r × 3c)

id	nama_lengkap	jabatan_terakhir
1	Admin	(NULL)
2	Tes User	(NULL)
3	John Doe	System Analyst
4	Fulan Bin Fulan	Project Manager
5	Mawar Kurniani	Senior Programmer
6	Melati Rahmawati	Junior Programmer
7	Dahlia Setiani	Junior Programmer
8	Lily Handayani	Magang

8. Latihan 3.3

```
SELECT K.id, K.nama_lengkap,
(
   SELECT J.nama_jabatan FROM jabatan_karyawan JK
   INNER JOIN jabatan J ON JK.jabatan_id = J.id
   WHERE karyawan_id = K.id ORDER BY JK.tanggal_mulai DESC
   LIMIT 1
) jabatan_terakhir,
(
   SELECT JK.tanggal_mulai FROM jabatan_karyawan JK
   WHERE karyawan_id=K.id ORDER BY JK.tanggal_mulai DESC LIMIT 1
)  tanggal_mulai_terakhir

FROM karyawan K;
```

jabatan_karyawan (8r × 4c)

id	nama_lengkap	jabatan_terakhir	tanggal_mulai_terakhir
1	Admin	(NULL)	(NULL)
2	Tes User	(NULL)	(NULL)
3	John Doe	System Analyst	2017-07-01
4	Fulan Bin Fulan	Project Manager	2017-07-02
5	Mawar Kurniani	Senior Programmer	2017-07-02
6	Melati Rahmawati	Junior Programmer	2017-07-02
7	Dahlia Setiani	Junior Programmer	2018-08-02
8	Lily Handayani	Magang	2018-08-08

9. Menampilkan Seluruh Jabatan Beserta Karyawaan yang menjabat

```sql
SELECT
(
SELECT J.nama_jabatan FROM jabatan_karyawan JK
INNER JOIN jabatan J ON JK.jabatan_id = J.id
WHERE karyawan_id = K.id
ORDER BY JK.tanggal_mulai DESC LIMIT 1
) jabatan_terakhir,
COUNT(*) jumlah
FROM karyawan K
GROUP BY jabatan_terakhir
```

jabatan_karyawan (6r × 2c)

jabatan_terakhir	jumlah
(NULL)	2
Junior Programmer	2
Magang	1
Project Manager	1
Senior Programmer	1
System Analyst	1

10. Latihan 3.4

```sql
SELECT K.id, K.nama_lengkap,
(
SELECT J.nama_jabatan FROM jabatan_karyawan JK
INNER JOIN jabatan J ON JK.jabatan_id = J.id
WHERE karyawan_id = K.id ORDER BY JK.tanggal_mulai DESC
LIMIT 1
) jabatan_terakhir,
(
SELECT JK.tanggal_mulai FROM jabatan_karyawan JK
WHERE karyawan_id=K.id ORDER BY JK.tanggal_mulai DESC LIMIT 1
) tanggal_mulai_terakhir,
(
SELECT B.nama_bagian FROM bagian_karyawan BK
INNER JOIN bagian B ON BK.bagian_id = B.id
WHERE BK.karyawan_id=K.id
) bagian_terakhir,
(
SELECT BK.tanggal_mulai FROM bagian_karyawan BK
WHERE BK.karyawan_id=K.id
) tanggal_bagian_terakhir
FROM karyawan K;
```

jabatan_karyawan (8r × 6c)

id	nama_lengkap	jabatan_terakhir	tanggal_mulai_terakhir	bagian_terakhir	tanggal_bagian_terakhir
1	Admin	(NULL)	(NULL)	(NULL)	(NULL)
2	Tes User	(NULL)	(NULL)	(NULL)	(NULL)
3	John Doe	System Analyst	2017-07-01	Data Science	2018-04-01
4	Fulan Bin Fulan	Project Manager	2017-07-02	Autentikasi	2018-03-03
5	Mawar Kurniani	Senior Programmer	2017-07-02	Autentikasi	2018-03-03
6	Melati Rahmawati	Junior Programmer	2017-07-02	Backend Developer	2018-04-04
7	Dahlia Setiani	Junior Programmer	2018-08-02	Data Science	2018-04-01
8	Lily Handayani	Magang	2018-08-08	Backend Developer	2018-04-04

```sql
SELECT
(
SELECT B.nama_bagian FROM bagian_karyawan BK
INNER JOIN bagian B ON BK.bagian_id = B.id
WHERE BK.karyawan_id=K.id
) bagian_terakhir,
COUNT(*) jumlah
FROM karyawan K
GROUP BY bagian_terakhir
```

bagian_karyawan (4r × 2c)

bagian_terakhir	jumlah
(NULL)	2
Autentikasi	2
Backend Developer	2
Data Science	2

Pertemuan 4

1. Penggajian

```
1  SELECT *FROM penggajian
```

penggajian (18r × 7c)

id	karyawan_id	tahun	bulan	gapok	tunjangan	uang_makan
1	3	2020	11	2,400,000	500,000	800,000
2	4	2020	11	2,300,000	450,000	684,000
3	5	2020	11	2,200,000	400,000	684,000
4	6	2020	11	2,100,000	350,000	640,000
5	7	2020	11	2,100,000	350,000	612,000
6	8	2020	11	1,000,000	100,000	380,000
7	3	2020	12	2,400,000	500,000	800,000
8	4	2020	12	2,300,000	450,000	722,000
9	5	2020	12	2,200,000	400,000	720,000
10	6	2020	12	2,100,000	350,000	680,000
11	7	2020	12	2,100,000	350,000	646,000
12	8	2020	12	1,000,000	100,000	360,000
13	3	2021	01	2,400,000	500,000	800,000
14	4	2021	01	2,300,000	450,000	722,000
15	5	2021	01	2,200,000	400,000	612,000
16	6	2021	01	2,100,000	350,000	680,000
17	7	2021	01	2,100,000	350,000	646,000
18	8	2021	01	1,000,000	100,000	340,000

2. Jumlah Gaji yang dibayarkan Per Tahun

```
1  SELECT tahun,
2     SUM(P.gapok) +
3     SUM(P.tunjangan) +
4     SUM(P.uang_makan)
5     jumlah_bayar_gaji
6     FROM penggajian P
7     GROUP BY tahun;
```

Result #1 (2r × 2c)

tahun	jumlah_bayar_gaji
2020	36,228,000
2021	18,050,000

3. Rincian Jumlah Gaji yang dibayarkan Per Tahun

```
SELECT tahun,
   SUM(P.gapok) jumlah_gapok,
   SUM(P.tunjangan) jumlah_tunjangan,
   SUM(P.uang_makan) jumlah_uang_makan
FROM penggajian P
GROUP BY tahun;
```

penggajian (2r × 4c)

tahun	jumlah_gapok	jumlah_tunjangan	jumlah_uang_makan
2020	24,200,000	4,300,000	7,728,000
2021	12,100,000	2,150,000	3,800,000

4. Rincian Jumlah Gaji yang dibayarkan Per Bulan

```
SELECT tahun, bulan,
   SUM(P.gapok) jumlah_gapok,
   SUM(P.tunjangan) jumlah_tunjangan,
   SUM(P.uang_makan) jumlah_uang_makan
FROM penggajian P
GROUP BY tahun, bulan;
```

penggajian (3r × 5c)

tahun	bulan	jumlah_gapok	jumlah_tunjangan	jumlah_uang_makan
2020	11	12,100,000	2,150,000	3,800,000
2020	12	12,100,000	2,150,000	3,928,000
2021	01	12,100,000	2,150,000	3,800,000

5. Rincian Jumlah Gaji yang dibayarkan Per Karyawan dalam 1 tahun

```
SELECT P.tahun,
   P.karyawan_id,
   K.nama_lengkap,
   SUM(P.gapok) jumlah_gapok,
   SUM(P.tunjangan) jumlah_tunjangan,
   SUM(P.uang_makan) jumlah_uang_makan
FROM penggajian P
LEFT JOIN karyawan K ON P.karyawan_id = K.id
WHERE P.tahun = "2020"
GROUP BY P.karyawan_id
```

penggajian (6r × 6c)

tahun	karyawan_id	nama_lengkap	jumlah_gapok	jumlah_tunjangan	jumlah_uang_makan
2020	3	John Doe	4,800,000	1,000,000	1,600,000
2020	4	Fulan Bin Fulan	4,600,000	900,000	1,406,000
2020	5	Mawar Kurniani	4,400,000	800,000	1,404,000
2020	6	Melati Rahmawati	4,200,000	700,000	1,320,000
2020	7	Dahlia Setiani	4,200,000	700,000	1,258,000
2020	8	Lily Handayani	2,000,000	200,000	740,000

6. Rincian Bulanan Jumlah Gaji yang dibayarkan 1 karyawan dalam 1 tahun

```
SELECT P.tahun, P.bulan,
   P.karyawan_id,
   K.nama_lengkap,
   P.gapok,
   P.tunjangan,
   P.uang_makan
FROM penggajian P
LEFT JOIN karyawan K ON P.karyawan_id = K.id
WHERE P.tahun = "2020" AND karyawan_id = 3
```

penggajian (2r × 7c)

tahun	bulan	karyawan_id	nama_lengkap	gapok	tunjangan	uang_makan
2020	11	3	John Doe	2,400,000	500,000	800,000
2020	12	3	John Doe	2,400,000	500,000	800,000

7. Slip Gaji

```
SELECT P.tahun, P.bulan,
   P.karyawan_id,
   K.nama_lengkap,
   P.gapok,
   P.tunjangan,
   P.uang_makan
FROM penggajian P
LEFT JOIN karyawan K ON P.karyawan_id = K.id
WHERE P.tahun = "2020" AND P.bulan = "12" AND karyawan_id = 3
```

penggajian (1r × 7c)

tahun	bulan	karyawan_id	nama_lengkap	gapok	tunjangan	uang_makan
2020	12	3	John Doe	2,400,000	500,000	800,000

8. Presensi

```
SELECT * FROM presensi
```

presensi (552r × 6c)

id	karyawan_id	tanggal	jam_masuk	jam_keluar	keterangan
1	3	2020-11-01	(NULL)	(NULL)	AKHIR PEKAN
2	3	2020-11-02	08:00:00	16:00:00	HADIR
3	3	2020-11-03	08:00:00	16:00:00	HADIR
4	3	2020-11-04	08:00:00	16:00:00	HADIR
5	3	2020-11-05	08:00:00	16:00:00	HADIR
6	3	2020-11-06	08:00:00	16:00:00	HADIR
7	3	2020-11-07	(NULL)	(NULL)	AKHIR PEKAN
8	3	2020-11-08	(NULL)	(NULL)	AKHIR PEKAN
9	3	2020-11-09	08:00:00	16:00:00	HADIR
10	3	2020-11-10	(NULL)	(NULL)	LIBUR NASIONAL

9. Jumlah Presensi Seluruh Karyawan per Keterangan setiap tahun

```
SELECT YEAR(tanggal) tahun,
SUM(case when keterangan = 'HADIR' then 1 else 0 END) jumlah_hadir,
SUM(case when keterangan = 'SAKIT' then 1 else 0 END) jumlah_sakit,
SUM(case when keterangan = 'IZIN' then 1 else 0 END) jumlah_izin,
SUM(case when keterangan = 'CUTI' then 1 else 0 END) jumlah_cuti,
SUM(case when keterangan = 'AKHIR PEKAN' then 1 else 0 END) jumlah_akhir_pekan,
SUM(case when keterangan = 'LIBUR NASIONAL' then 1 ELSE 0 END) jumlah_libur_nasional,
SUM(case when keterangan = 'TANPA KETERANGAN' then 1 ELSE 0 END) jumlah_tanpa_keterangan,
COUNT(*) total
FROM `presensi` GROUP BY tahun
```

presensi (2r × 9c)

tahun	jumlah_hadir	jumlah_sakit	jumlah_izin	jumlah_cuti	jumlah_akhir_pekan	jumlah_libur_nasional	jumlah_tanpa_keterangan	total
2,020	230	5	3	2	102	24	0	366
2,021	112	4	4	0	60	6	0	186

```
SELECT YEAR(tanggal) tahun,
SUM(case when keterangan = 'HADIR' then 1 else 0 END) jumlah_hadir,
SUM(case when keterangan = 'SAKIT' then 1 else 0 END) jumlah_sakit,
SUM(case when keterangan = 'IZIN' then 1 else 0 END) jumlah_izin,
SUM(case when keterangan = 'CUTI' then 1 else 0 END) jumlah_cuti,
SUM(case when keterangan = 'AKHIR PEKAN' then 1 else 0 END) jumlah_akhir_pekan,
SUM(case when keterangan = 'LIBUR NASIONAL' then 1 ELSE 0 END) jumlah_libur_nasional,
SUM(case when keterangan = 'TANPA KETERANGAN' then 1 ELSE 0 END) jumlah_tanpa_keterangan,
COUNT(*) total
FROM `presensi` WHERE YEAR(tanggal) = 2020 GROUP BY tahun
```

presensi (1r × 9c)

tahun	jumlah_hadir	jumlah_sakit	jumlah_izin	jumlah_cuti	jumlah_akhir_pekan	jumlah_libur_nasional	jumlah_tanpa_keterangan	total
2,020	230	5	3	2	102	24	0	366

10. Rincian Bulanan Jumlah Presensi Seluruh Karyawan per Keterangan

```
SELECT YEAR(tanggal) tahun, MONTH(tanggal) bulan,
SUM(case when keterangan = 'HADIR' then 1 else 0 END) jumlah_hadir,
SUM(case when keterangan = 'SAKIT' then 1 else 0 END) jumlah_sakit,
SUM(case when keterangan = 'IZIN' then 1 else 0 END) jumlah_izin,
SUM(case when keterangan = 'CUTI' then 1 else 0 END) jumlah_cuti,
SUM(case when keterangan = 'AKHIR PEKAN' then 1 else 0 END) jumlah_akhir_pekan,
SUM(case when keterangan = 'LIBUR NASIONAL' then 1 ELSE 0 END) jumlah_libur_nasional,
SUM(case when keterangan = 'TANPA KETERANGAN' then 1 ELSE 0 END) jumlah_tanpa_keterangan,
COUNT(*) total
FROM `presensi` WHERE YEAR(tanggal) = 2020 GROUP BY tahun, bulan
```

presensi (2r × 10c)

tahun	bulan	jumlah_hadir	jumlah_sakit	jumlah_izin	jumlah_cuti	jumlah_akhir_pekan	jumlah_libur_nasional	jumlah_tanpa_keterangan	total
2,020	11	114	3	3	0	54	6	0	180
2,020	12	116	2	0	2	48	18	0	186

11. Rincian Bulanan Jumlah Presensi 1 Karyawan per Keterangan

```
SELECT YEAR(tanggal) tahun, MONTH(tanggal) bulan,
SUM(case when keterangan = 'HADIR' then 1 else 0 END) jumlah_hadir,
SUM(case when keterangan = 'SAKIT' then 1 else 0 END) jumlah_sakit,
SUM(case when keterangan = 'IZIN' then 1 else 0 END) jumlah_izin,
SUM(case when keterangan = 'CUTI' then 1 else 0 END) jumlah_cuti,
SUM(case when keterangan = 'AKHIR PEKAN' then 1 else 0 END) jumlah_akhir_pekan,
SUM(case when keterangan = 'LIBUR NASIONAL' then 1 ELSE 0 END) jumlah_libur_nasional,
SUM(case when keterangan = 'TANPA KETERANGAN' then 1 ELSE 0 END) jumlah_tanpa_keterangan,
COUNT(*) total
FROM `presensi` WHERE karyawan_id = 4 AND YEAR(tanggal) = 2020 GROUP BY tahun, bulan
```

presensi (2r × 10c)

tahun	bulan	jumlah_hadir	jumlah_sakit	jumlah_izin	jumlah_cuti	jumlah_akhir_pekan	jumlah_libur_nasional	jumlah_tanpa_keterangan	total
2,020	11	18	1	1	0	9	1	0	30
2,020	12	19	1	0	0	8	3	0	31

12. Rincian 1 Bulan Jumlah Presensi 1 Karyawan per Keterangan

```
SELECT YEAR(tanggal) tahun, MONTH(tanggal) bulan, karyawan_id,
SUM(case when keterangan = 'HADIR' then 1 else 0 END) jumlah_hadir,
SUM(case when keterangan = 'SAKIT' then 1 else 0 END) jumlah_sakit,
SUM(case when keterangan = 'IZIN' then 1 else 0 END) jumlah_izin,
SUM(case when keterangan = 'CUTI' then 1 else 0 END) jumlah_cuti,
SUM(case when keterangan = 'AKHIR PEKAN' then 1 else 0 END) jumlah_akhir_pekan,
SUM(case when keterangan = 'LIBUR NASIONAL' then 1 ELSE 0 END) jumlah_libur_nasional,
SUM(case when keterangan = 'TANPA KETERANGAN' then 1 ELSE 0 END) jumlah_tanpa_keterangan,
COUNT(*) total
FROM `presensi` WHERE YEAR(tanggal) = 2020 AND MONTH(tanggal) = 11 GROUP BY tahun, bulan, karyawan_id
```

presensi (6r × 11c)

tahun	bulan	karyawan_id	jumlah_hadir	jumlah_sakit	jumlah_izin	jumlah_cuti	jumlah_akhir_pekan	jumlah_libur_nasional	jumlah_tanpa_keterangan	total
2,020	11	3	20	0	0	0	9	1	0	30
2,020	11	4	18	1	1	0	9	1	0	30
2,020	11	5	19	1	0	0	9	1	0	30
2,020	11	6	20	0	0	0	9	1	0	30
2,020	11	7	18	1	1	0	9	1	0	30
2,020	11	8	19	0	1	0	9	1	0	30