SQL Query

Dashboard

```sql
CREATE TABLE user_info (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  email VARCHAR(255),
  full_name VARCHAR(255),
  birthday DATE,
  last_active_at TIMESTAMP
);

INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a
INSERT INTO user_info (email, full_name, birthday, last_active_a


CREATE TABLE workspace (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  name VARCHAR(255),
  owner_id INT REFERENCES user_info(id)
);

INSERT INTO workspace (name, owner_id)
    VALUES
        ('Glassy Inc',1),
      ('bbv Vietname',3);
```

```sql
CREATE TABLE workspace_member (
  workspace_id INT REFERENCES workspace(id),
  user_id INT REFERENCES user_info(id),
  join_at TIMESTAMP,
  left_at TIMESTAMP
);

INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at) V/
INSERT INTO workspace_member (workspace_id, user_id, join_at,lef

CREATE TABLE channel (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  workspace_id INT REFERENCES workspace(id),
  name varchar(255)
);

INSERT INTO channel (workspace_id, name)
    VALUES
        (1,'team-chat'),
        (1,'project-gizmo'),
        (1,'genz-team');

CREATE TABLE status_message (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  name VARCHAR(255)
```

```sql
);

INSERT INTO status_message (name)
    VALUES
        ('pending'),
        ('sent'),
        ('read'),
        ('error');

CREATE TABLE direct_message (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  user1 INT REFERENCES user_info(id),
  user2 INT REFERENCES user_info(id),
  status_id INT REFERENCES status_message(id)
);

INSERT INTO direct_message (user1, user2, status_id)
    VALUES
        (1,2,2),
        (1,3,2),
        (2,3,3);




CREATE TABLE message (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  sender_id INT REFERENCES user_info(id),
  content text,
  channel_id INT REFERENCES channel(id),
  dms_id INT REFERENCES direct_message(id)
);


INSERT INTO message (sender_id, content, channel_id, dms_id) VAI
INSERT INTO message (sender_id, content, channel_id, dms_id) VAI
```

```sql
INSERT INTO message (sender_id, content, channel_id, dms_id) VAl
INSERT INTO message (sender_id, content, channel_id, dms_id) VAl
INSERT INTO message (sender_id, content, channel_id, dms_id) VAl
INSERT INTO message (sender_id, content, channel_id, dms_id) VAl
INSERT INTO message (sender_id, content, channel_id, dms_id) VAl
INSERT INTO message (sender_id, content, channel_id, dms_id) VAl
INSERT INTO message (sender_id, content, channel_id, dms_id) VAl
INSERT INTO message (sender_id, content, channel_id, dms_id) VAl


CREATE TABLE mention_list (
  message_id INT REFERENCES message(id),
  user_id INT REFERENCES user_info(id)
);

INSERT INTO mention_list (message_id, user_id)
    VALUES
        (9,4),
        (9,5);

CREATE TABLE notification_type (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  name VARCHAR(255)
);

INSERT INTO notification_type (name)
    VALUES
        ('Welcome'),
        ('Meeting'),
        ('Mention');

CREATE TABLE notification_queue (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  user_id INT REFERENCES user_info(id),
  notification_type_id INT REFERENCES notification_type(id),
  content TEXT
```

```sql
);

INSERT INTO notification_queue (user_id, notification_type_id, d
    VALUES
        (1,1,'Welcome to MentorHub'),
        (1,2,'Peter invite you join meeting'),
        (1,3,'Anna mention you in #team-chat');

CREATE TABLE meeting (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  owner_id INT REFERENCES user_info(id),
  name VARCHAR(255),
  start_at TIMESTAMP,
  end_at TIMESTAMP
);

INSERT INTO meeting (owner_id, name, start_at, end_at)
    VALUES
        (2,'Welcome to my program','2024-06-13T09:09:00.000','2(
        (3,'Training','2024-06-13T09:11:00.000','2024-06-13T09:

CREATE TABLE meeting_status (
  id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
  name VARCHAR(255)
);

INSERT INTO meeting_status (name)
    VALUES
        ('accepted'),
        ('declined');

CREATE TABLE meeting_participant (
  meeting_id INT REFERENCES meeting(id),
  user_id INT REFERENCES user_info(id),
  status_id INT REFERENCES meeting_status(id)
);
```

```sql
INSERT INTO meeting_participant (meeting_id, user_id, status_id
    VALUES
        (1,1,1),
        (1,2,1),
        (1,3,1),
      (2,1,2),
        (2,2,1),
        (2,3,1);
```

```sql
-- Query
-- New member
SELECT
    w.name,
    COUNT(user_id) as new_member
FROM workspace_member as wm
JOIN workspace AS w on w.id = wm.workspace_id
WHERE DATE(join_at) = CURRENT_DATE -- current
GROUP BY w.name;

SELECT
    w.name,
    COUNT(user_id) as new_member
FROM workspace_member as wm
JOIN workspace AS w on w.id = wm.workspace_id
WHERE DATE(join_at) = (CURRENT_DATE - INTERVAL'1 day') -- yester
GROUP BY w.name
```