Discord

1. Резултати от създадените **функции**

Функция **fn_UserLastActivity(@UserID)**

Функцията връща последната активност на даден потребител – датата и часа на последното му изпратено съобщение в платформата.

```
CREATE FUNCTION fn_UserLastActivity (@UserID INT)
RETURNS DATETIME2
AS
BEGIN
   DECLARE @LastActivity DATETIME2;

   SELECT @LastActivity = MAX(sent_at)
   FROM Message
   WHERE author_user_id = @UserID;

   RETURN @LastActivity;
END;
```

Пример 1: Проверка на последната активност на един потребител

```
SELECT dbo.fn_UserLastActivity(3) AS LastActivity;
```

Фигура 1. *Резултат от функцията за един потребител*

	LastActivity
1	2025-11-20 12:14:54.0711781

Пример 2: Последна активност за всички потребители

```
SELECT
   u.user_id,
   u.username,
   dbo.fn_UserLastActivity(u.user_id) AS LastActivity
FROM [User] u
ORDER BY LastActivity DESC;
```

Фигура 2. *Резултати от функцията за всички потребители*

	user_id	username	LastActivity
1	2	Bob	2025-11-22 12:19:26.6656050
2	8	Hannah	2025-11-22 12:04:54.0711781
3	7	George	2025-11-22 11:54:54.0711781
4	5	Eugene	2025-11-22 11:34:54.0711781
5	10	Julia	2025-11-22 11:14:54.0711781
6	9	Ivan	2025-11-22 09:14:54.0711781
7	1	Alice	2025-11-21 12:14:54.0711781
8	6	Fiona	2025-11-21 06:14:54.0711781
9	4	Diana	2025-11-20 17:14:54.0711781
10	3	Charlie	2025-11-20 12:14:54.0711781

2. Резултати от създадените **процедури**

Процедура **CreateMessage**

Процедурата добавя ново съобщение в канал, след като провери дали каналът и потребителят съществуват. При некоректни стойности връща грешка.

```
CREATE PROCEDURE CreateMessage
   @ChannelID INT,
   @AuthorUserID INT,
   @Content NVARCHAR(MAX)
AS
BEGIN
   SET NOCOUNT ON;

   IF NOT EXISTS (SELECT 1 FROM Channel WHERE channel_id = @ChannelID)
   BEGIN
      RAISERROR('Channel does not exist.', 16, 1);
      RETURN;
   END;

   IF NOT EXISTS (SELECT 1 FROM [User] WHERE user_id = @AuthorUserID)
   BEGIN
      RAISERROR('User does not exist.', 16, 1);
      RETURN;
   END;

   INSERT INTO Message(channel_id, author_user_id, content)
   VALUES(@ChannelID, @AuthorUserID, @Content);
END;
```

Пример 1: Успешно изпълнение на процедурата

```
EXEC CreateMessage 1, 3, N'Hello from stored procedure!';

SELECT TOP 10 message_id, channel_id, author_user_id, content, sent_at

FROM Message

ORDER BY message_id DESC;
```

Фигура 3. *Успешно добавено съобщение чрез процедурата*

	message_id	channel_id	author_user_id	content	sent_at
1	28	1	3	Hello from stored procedure!	2025-11-22 12:27:47.2801272
2	27	5	2	Hello!	2025-11-22 12:19:26.6656050
3	26	14	8	Looking for internship opportunities in summer.	2025-11-22 12:04:54.0711781
4	25	14	7	Junior backend position open at our company, DM ...	2025-11-22 11:54:54.0711781
5	24	13	5	Nice work! The lighting looks great.	2025-11-22 11:34:54.0711781
6	23	13	10	I made a small game in Unity, feedback appreciated!	2025-11-22 11:14:54.0711781
7	22	12	5	Post the execution plan, we can take a look.	2025-11-22 10:14:54.0711781
8	21	12	9	Why is my query so slow? Any indexing tips?	2025-11-22 09:14:54.0711781
9	20	11	8	Trying to understand Entity Framework migrations.	2025-11-22 08:14:54.0711781
10	19	11	7	I'm building a Discord bot in C#.	2025-11-22 07:14:54.0711781

3. Резултати от създадените **тригери**

Тригер **trg_Message_Insert_UpdateStats**

Тригерът се активира автоматично след всяко ново съобщение в таблица **Message**.
Той актуализира таблицата ChannelStats, като:

- увеличава броя на съобщенията за канала
- обновява последната дата/час на съобщението

```
CREATE TRIGGER trg_Message_Insert_UpdateStats

ON Message

AFTER INSERT

AS

BEGIN

  SET NOCOUNT ON;


  MERGE ChannelStats AS target

  USING (

    SELECT channel_id,

        COUNT(*) AS Cnt,

        MAX(sent_at) AS LastMsg

    FROM inserted

    GROUP BY channel_id
```

```
    ) AS src
    ON target.channel_id = src.channel_id
    WHEN MATCHED THEN
        UPDATE SET
            target.message_count = target.message_count + src.Cnt,
            target.last_message_at = src.LastMsg
    WHEN NOT MATCHED THEN
        INSERT(channel_id, message_count, last_message_at)
        VALUES(src.channel_id, src.Cnt, src.LastMsg);
END;
```

Пример: Актуализирана таблица **ChannelStats**

```
SELECT
    cs.channel_id,
    c.name AS channel_name,
    cs.message_count,
    cs.last_message_at
FROM ChannelStats cs
JOIN Channel c ON c.channel_id = cs.channel_id
ORDER BY cs.channel_id;
```

Фигура 4. *Обновена статистика за каналите след задействане на тригера*

	channel_id	channel_name	message_count	last_message_at
1	1	general	5	2025-11-22 12:27:47.2801272
2	2	lfg	3	2025-11-21 12:14:54.0711781
3	4	announcements	2	2025-11-18 13:14:54.0711781
4	5	homework	5	2025-11-22 12:28:50.9041101
5	6	exams	1	2025-11-20 11:14:54.0711781
6	8	now-playing	3	2025-11-21 09:14:54.0711781
7	9	requests	1	2025-11-21 11:44:54.0711781
8	11	general-dev	3	2025-11-22 08:14:54.0711781
9	12	help	2	2025-11-22 10:14:54.0711781
10	13	showcase	2	2025-11-22 11:34:54.0711781
11	14	jobs	2	2025-11-22 12:04:54.0711781