

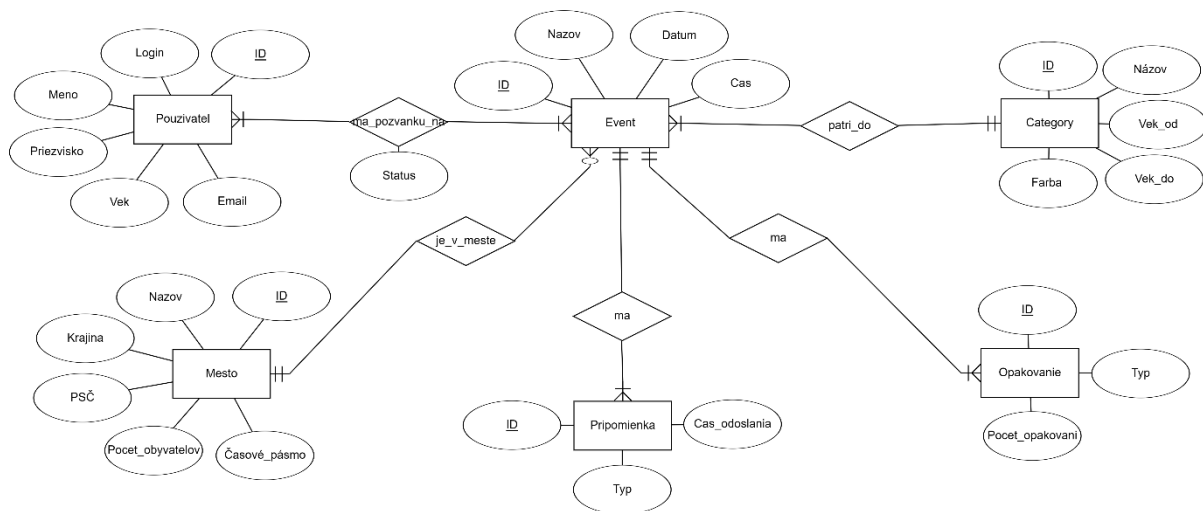
ZADANIE Z PREDMETU DATABÁZOVÉ SYSTÉMY

NÁZOV DOMÉNY: KALENDÁR PLÁNOVANÝCH UDALOSTÍ

MENO A PRIEZVISKO: OLEKSANDR TANCHUK

AK. ROK: 2024/2025

ENTITNO-RELAČNÝ DIAGRAM



OPIS ENTITNÝCH MNOŽÍN

1. **Používateľ** - predstavuje bežného používateľa, ktorý môže byť pozvaný na udalosť.

- ID (PK) – unikátne identifikačné číslo používateľa
- Meno – meno používateľa
- Priezvisko – priezvisko používateľa
- Email – e-mail používateľa
- Vek – vek používateľa
- Login – prihlasovacie meno používateľa

2. **Event** - predstavuje udalosť, na ktorú môže byť používateľ pozvaný.

- ID (PK) – unikátne identifikačné číslo udalosti
- Názov – názov udalosti
- Dátum – dátum konania udalosti
- Čas_začiatku – čas začiatku udalosti

3. **Mesto** - predstavuje mesto, kde sa udalosť koná.

- ID (PK) – unikátne identifikačné číslo mesta
- Názov – názov mesta
- Krajina – krajina, v ktorej sa mesto nachádza
- PSČ – poštové smerovacie číslo mesta
- Pocet_obyvatelov – počet obyvateľov mesta
- Časové_pásma – časové pásma, v ktorom sa mesto nachádza

4. **Kategória** - predstavuje kategóriu udalosti.

- ID (PK) – unikátne identifikačné číslo kategórie
- Názov – názov kategórie
- Farba – farba kategórie
- Vek_od – minimálny vek pre kategóriu
- Vek_do – maximálny vek pre kategóriu

5. **Pripomienka** - predstavuje pripomienku spojenú s udalosťou.
 - ID (PK) – unikátne identifikačné číslo pripomienky
 - Typ – typ pripomienky
 - Cas_odoslania – čas odoslania pripomienky
6. **Opakovanie** - predstavuje opakovanie udalosti.
 - ID (PK) – unikátne identifikačné číslo opakovania
 - Typ – typ opakovania (napr. denne, týždenne, mesačne)
 - Pocet_opakovani – počet opakovaní udalosti

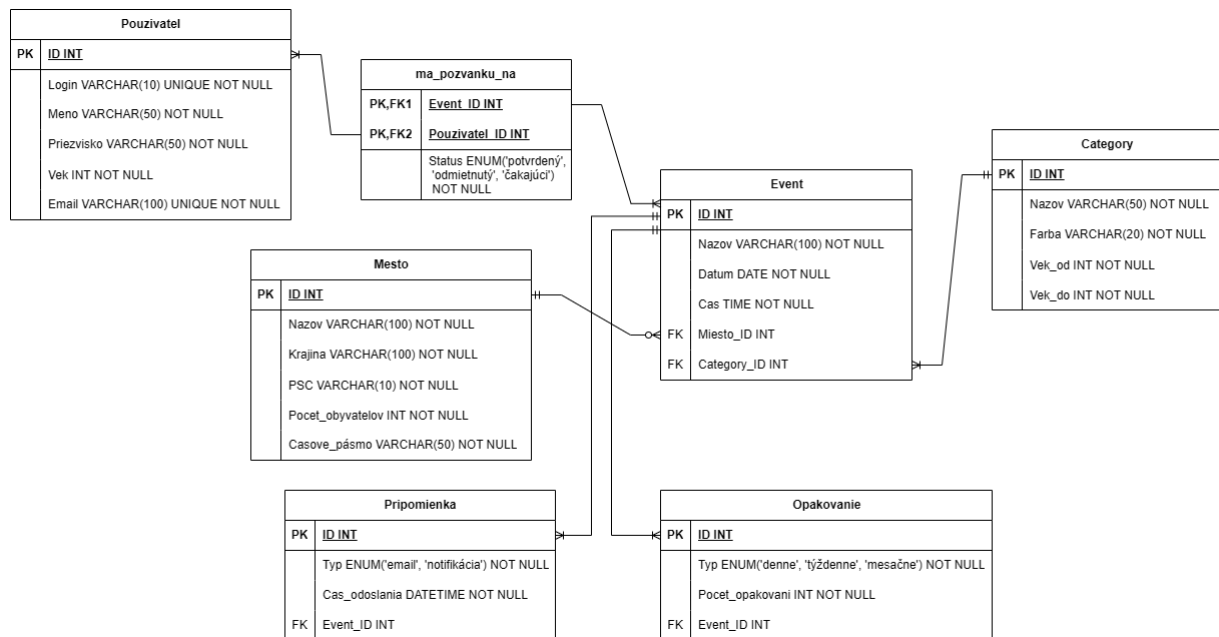
OPIS VZŤAHOV MEDZI ENTITNÝMI MNOŽINAMI

- 1) **ma_pozvanku_na** (Používateľ ↔ Event)
 - Vzťah medzi entitami Používateľ a Event (M:N).
 - Jeden Používateľ môže byť pozvaný na viacero Udalostí.
 - Na jednu Udalosť môže byť pozvaných viacero Používateľov.

Atribút:

 - Status – určuje stav pozvánky (cakajuci, potvrdeny, odmietnuty).
- 2) **je_v_meste** (Event → Mesto)
 - Vzťah medzi entitami Event a Mesto (1:N).
 - Jedna Udalosť sa koná v jednom Meste.
 - V jednom Meste sa môže konať viacero Udalostí.
- 3) **patrí_do** (Event → Category)
 - Vzťah medzi entitami Event a Category (M:1).
 - Jedna Udalosť môže patriť do jedného Kategórií.
 - Jedna Kategória môže obsahovať viacero Udalostí.
- 4) **ma** (Event → Pripomienka)
 - Vzťah medzi entitami Event a Pripomienka (1:N).
 - Jedna Udalosť môže mať viacero Pripomienok.
 - Jedna Pripomienka patrí k jednej Udalosti.
- 5) **ma** (Event → Opakovanie)
 - Vzťah medzi entitami Event a Opakovanie (1:N).
 - Jedna Udalosť môže mať viacero Opakovaní.
 - Jedno Opakovanie patrí k jednej Udalosti.

RELAČNÝ MODEL



PRVÁ ODOVZDÁVKA - SQL SKRIPT NA VYTVORENIE DATABÁZY

```
DROP TABLE IF EXISTS Pripomienka CASCADE;
DROP TABLE IF EXISTS Opakovanie CASCADE;
DROP TABLE IF EXISTS ma_pozvanku_na CASCADE;
DROP TABLE IF EXISTS Event CASCADE;
DROP TABLE IF EXISTS Mesto CASCADE;
DROP TABLE IF EXISTS Category CASCADE;
DROP TABLE IF EXISTS Pouzivatel CASCADE;
```

```
CREATE TABLE Pouzivatel (  
    ID INT PRIMARY KEY,  
    Login VARCHAR(10) UNIQUE NOT NULL,  
    Meno VARCHAR(50) NOT NULL,  
    Priezvisko VARCHAR(50) NOT NULL,  
    Vek INT NOT NULL,  
    Email VARCHAR(100) UNIQUE NOT NULL  
);
```

```
CREATE TABLE Category (
  ID INT PRIMARY KEY,
  Nazov VARCHAR(50) NOT NULL,
  Farba VARCHAR(20) NOT NULL,
  Vek_od INT NOT NULL,
  Vek_do INT NOT NULL
);
```

```
CREATE TABLE Mesto (  
    ID INT PRIMARY KEY,  
    Nazov VARCHAR(100) NOT NULL,  
    Krajina VARCHAR(100) NOT NULL,  
    PSC VARCHAR(10) NOT NULL,  
    Pocet_obyvatelov INT NOT NULL,  
    Casove_pasmo VARCHAR(50) NOT NULL  
);
```

```
CREATE TABLE Event (  
    ID INT PRIMARY KEY,  
    Nazov VARCHAR(100) NOT NULL,  
    Datum DATE NOT NULL,  
    Cas TIME NOT NULL,  
    Miesto_ID INT,  
    Category_ID INT,  
    Author_ID INT,  
    FOREIGN KEY (Miesto_ID) REFERENCES Mesto(ID) ON DELETE SET NULL,  
    FOREIGN KEY (Category_ID) REFERENCES Category(ID) ON DELETE SET NULL,  
    FOREIGN KEY (Author_ID) REFERENCES Pouzivatel(ID) ON DELETE SET NULL  
);
```

```
CREATE TABLE ma_pozvanku_na (  
    Event_ID INT,  
    Pouzivatel_ID INT,  
    Status VARCHAR(20) NOT NULL CHECK (Status IN ('potvrdeny', 'odmietnuty', 'cakajuci')),  
    PRIMARY KEY (Event_ID, Pouzivatel_ID),  
    FOREIGN KEY (Event_ID) REFERENCES Event(ID) ON DELETE CASCADE,  
    FOREIGN KEY (Pouzivatel_ID) REFERENCES Pouzivatel(ID) ON DELETE CASCADE  
);
```

```
CREATE TABLE Opakovanie (  
    ID INT PRIMARY KEY,  
    Typ VARCHAR(20) NOT NULL CHECK (Typ IN ('denne', 'tyzdenne', 'mesacne')),  
    Pocet_opakovani INT NOT NULL,  
    Event_ID INT,  
    FOREIGN KEY (Event_ID) REFERENCES Event(ID) ON DELETE CASCADE  
);
```

```
CREATE TABLE Pripomienka (  
    ID INT PRIMARY KEY,  
    Typ VARCHAR(20) NOT NULL CHECK (Typ IN ('email', 'notifikacia')),  
    Cas_odoslania TIMESTAMP NOT NULL,  
    Event_ID INT,  
    FOREIGN KEY (Event_ID) REFERENCES Event(ID) ON DELETE CASCADE  
);
```

DRUHÁ ODOVZDÁVKA

SKRIPT NA NAPLNENIE DATABÁZY (KAŽDÁ TABUĽKA ASPOŇ 5 ZÁZNAMOV)

--vypln tabulek

```
INSERT INTO Pouzivatel (ID, Login, Meno, Priezvisko, Vek, Email) VALUES
(1, 'spiderPP', 'Tom', 'Holland', 28, 'peter.parker@gmail.com'),
(2, 'RDJironman', 'Tony', 'Stark', 40, 'tony.stark@starkindustries.com'),
(3, 'thor777', 'Chris', 'Hemsworth', 40, 'thor.odinson@asgard.com'),
(4, 'evsm04th', 'Eva', 'Smith', 21, 'eva.smith@student.tuke.sk'),
(5, 'peterjun1', 'Peter', 'Lukac', 24, 'peterjun1@infovek.sk'),
(6, 'maria01', 'Maria', 'Kovac', 20, 'maria01@stuba.sk'),
(7, 'martin_s', 'Martin', 'Kovacik', 19, 'martin_s@trnava.sk'),
(8, 'martina8', 'Martina', 'Kuzmova', 26, 'martina8@infovek.sk'),
(9, 'janek01', 'Janko', 'Rados', 23, 'janek01@infovek.sk'),
(10, 'lenka_02', 'Lenka', 'Hlavackova', 27, 'lenka_02@dpmkcmail.sk'),
(11, 'jakub99', 'Jakub', 'Petrovic', 29, 'jakub99@dpmkcmail.sk'),
(12, 'peter_01', 'Peter', 'Varga', 35, 'peter_01@cmpubianmail.sk'),
(13, 'peter_mi', 'Peter', 'Juris', 31, 'peter.jumic@microsoft.com'),
(14, 'evapetra', 'Eva', 'Kovalska', 31, 'evapetra@infovek.sk'),
(15, 'michal64', 'Michal', 'Varga', 34, 'michal64@infovek.sk'),
(16, 'karol01', 'Karol', 'Novotny', 28, 'karol01@dpmkcmail.sk'),
(17, 'livia_56', 'Livia', 'Jancarova', 32, 'livia_56@dpmkcmail.sk'),
(18, 'milan_78', 'Milan', 'Horvath', 34, 'milan_78@dpmkcmail.sk'),
(19, 'jan_89', 'Jan', 'Kozak', 33, 'jan_89@cmpubianmail.sk'),
(20, 'petersiem', 'Peter', 'Hrubec', 28, 'pet_hru.siem@siemens.sk'),
(21, 'luciasiem', 'Lucia', 'Lukacova', 29, 'luc_luk.siem@siemens.sk'),
(22, 'tomas_84', 'Tomas', 'Benko', 38, 'tomas_84@dpmkcmail.sk'),
(23, 'adam_03', 'Adam', 'Kovac', 35, 'adam_03@cmpubianmail.sk'),
(24, 'veronika4', 'Veronika', 'Siklova', 31, 'veronika4@cmpubianmail.sk'),
(25, 'michalsam', 'Michal', 'Kollar', 28, 'michal.kollar.sam@samsung-slovakia.com'),
(26, 'zuzana_2', 'Zuzana', 'Matuskova', 29, 'zuzana_2@cmpubianmail.sk'),
(27, 'pegooogle', 'Peter', 'Hlava', 31, 'peter.hlava-google@google-eu-sk.com'),
(28, 'ja2oogle', 'Jakub', 'Frolik', 29, 'jakub.frolik-google@google-eu-sk.com'),
(29, 'martopen', 'Martin', 'Blazek', 27, 'martin.blazek.ai@openai-sk.com'),
(30, 'eva_open', 'Eva', 'Dubeckova', 25, 'eva.dubeckova.ai@openai-sk.com'),
(31, 'zuzanasam', 'Zuzana', 'Liskova', 30, 'zuzana.liskova.sam@samsung-slovakia.com'),
(32, 'tomas_sam', 'Tomas', 'Novotny', 34, 'tomas.novotny.sam@samsung-slovakia.com'),
(33, 'lugoogle', 'Lucia', 'Kolarova', 29, 'lucia.kolarova-google@google-eu-sk.com'),
(34, 'jagoogle', 'Jana', 'Miklosova', 33, 'jana.miklosova-google@google-eu-sk.com'),
(35, 'jakubsiem', 'Jakub', 'Toth', 31, 'jak_tot.siem@siemens.sk'),
(36, 'luc646mic', 'Lucia', 'Skrabkova', 27, 'lucia.skmic@microsoft.com'),
(37, 'ov077sp', 'Oleksandr', 'Varga', 22, 'peter.varga@student.tuke.sk'),
(38, 'vs923sp', 'Vladimir', 'Slovak', 23, 'michal.slovak@student.tuke.sk'),
(39, 'zb987rt', 'Zuzana', 'Broskova', 24, 'zuzana.broskova@student.tuke.sk'),
(40, 'jf416no', 'Jana', 'Fialova', 25, 'jana.fialova@student.tuke.sk'),
(41, 'lt384er', 'Luka', 'Tomasov', 21, 'luka.tomasov@student.tuke.sk'),
(42, 'jz220wr', 'Jakub', 'Zelenka', 26, 'jakub.zelenka@student.tuke.sk'),
(43, 'lh416rf', 'Livia', 'Huskovicova', 22, 'livia.huskovicova@student.tuke.sk'),
```

(44, 'iv15no85', 'Ivan', 'Novak', 26, 'ivan_novak@stuba.sk'),
 (45, 'zu54hr24', 'Zuzana', 'Hruskova', 24, 'zuzka_hruskova@stuba.sk'),
 (46, 'to76lu17', 'Tomas', 'Lukac', 25, 'tomas_lukac@stuba.sk'),
 (47, 'januniba', 'Janko', 'Pavlik', 22, 'janko.pavlik.uniba@uniba.sk'),
 (48, 'maruniba', 'Maria', 'Dvorakova', 21, 'maria.dvorakova.uniba@uniba.sk'),
 (49, 'jakuniba', 'Jakub', 'Hruska', 23, 'jakub.hruska.uniba@uniba.sk'),
 (50, 'lucuniba', 'Lucia', 'Kovacova', 20, 'lucia.kovacova.uniba@uniba.sk'),
 (51, 'petuniba', 'Peter', 'Cierny', 22, 'peter.cierny.uniba@uniba.sk'),
 (52, 'mk126er', 'Martin', 'Kocis', 23, 'martin.kocis@student.tuke.sk'),
 (53, 'tm231ty', 'Tomas', 'Malik', 24, 'tomas.malik@student.tuke.sk'),
 (54, 'ek007po', 'Eva', 'Kollarova', 25, 'eva.kollarova@student.tuke.sk'),
 (55, 'rj095ke', 'Radka', 'Jurkova', 22, 'radka.jurkova@student.tuke.sk'),
 (56, 'km119rt', 'Karol', 'Mihalik', 23, 'karol.mihalik@student.tuke.sk'),
 (57, 'ka12fi11', 'Karol', 'Fiala', 24, 'karol_fiala@stuba.sk'),
 (58, 'ev45no09', 'Eva', 'Novotna', 25, 'eva_novotna@stuba.sk'),
 (59, 'ma81su77', 'Martina', 'Sulikova', 21, 'martina_sulikova@stuba.sk'),
 (60, 'john_doe', 'John', 'Doe', 33, 'john_doe@gmail.com'),
 (61, 'marper', 'Maria', 'Perez', 29, 'maria_perez@gmail.com'),
 (62, 'antram', 'Antonio', 'Ramos', 35, 'antonio_amos@gmail.com'),
 (63, '4n0n1m', 'Anonim', 'Anonimov', 99, 'anonim@gmail.com'),
 (64, 'admin1111', 'Admin', 'Adminov', 99, 'admin@gmail.com'),
 (65, 'annshe2ua', 'Anna', 'Shevchenko', 22, 'anna.shevchenko.ua@erasmus-europe.com'),
 (66, 'lucmul3be', 'Lucas', 'Muller', 24, 'lucas.muller.be@erasmus-europe.com'),
 (67, 'mardub2fr', 'Marie', 'Dubois', 23, 'marie.dubois.fr@erasmus-europe.com'),
 (68, 'gioros43i', 'Giovanni', 'Rossi', 25, 'giovanni.rossi.i@erasmus-europe.com'),
 (69, 'viki2006', 'Viktoria', 'Prsiazniuk', 28, 'lena.smith@gmail.com'),
 (70, 'qwerty394', 'Michal', 'Tkac', 27, 'michal.tk@gmail.com');

INSERT INTO Mesto (ID, Nazov, Krajina, PSC, Pocet_obyvatelov, Casove_pasmo) VALUES

('81101', 'Bratislava', 'Slovakia', '81101', 437725, 'CET'),
 ('04001', 'Košice', 'Slovakia', '04001', 240000, 'CET'),
 ('94901', 'Nitra', 'Slovakia', '94901', 78000, 'CET'),
 ('08001', 'Prešov', 'Slovakia', '08001', 91000, 'CET'),
 ('91701', 'Trnava', 'Slovakia', '91701', 67000, 'CET'),
 ('21001', 'Žilina', 'Slovakia', '01001', 82000, 'CET'),
 ('03601', 'Martin', 'Slovakia', '03601', 60000, 'CET'),
 ('91101', 'Trenčín', 'Slovakia', '91101', 58000, 'CET'),
 ('05801', 'Poprad', 'Slovakia', '05801', 55000, 'CET'),
 ('01701', 'Považská Bystrica', 'Slovakia', '01701', 43000, 'CET'),
 ('01051', 'Budapest', 'Hungary', '1051', 1752286, 'CET'),
 ('11000', 'Prague', 'Czech Republic', '11000', 1300000, 'CET'),
 ('01010', 'Vienna', 'Austria', '1010', 1897491, 'CET'),
 ('10115', 'Berlin', 'Germany', '10115', 3769000, 'CET'),
 ('00100', 'Rome', 'Italy', '00100', 2873000, 'CET'),
 ('75001', 'Paris', 'France', '75001', 2148000, 'CET'),
 ('01011', 'Amsterdam', 'Netherlands', '1011', 872680, 'CET'),
 ('01000', 'Brussels', 'Belgium', '1000', 1840000, 'CET'),
 ('00010', 'London', 'United Kingdom', 'EC1A 1BB', 8982000, 'GMT'),
 ('00150', 'Oslo', 'Norway', '0150', 1000000, 'CET'),
 ('01001', 'Kyiv', 'Ukraine', '01001', 2960000, 'EET'),

```
('79000', 'Lviv', 'Ukraine', '79000', 721301, 'EET'),  
('65000', 'Odessa', 'Ukraine', '65000', 1010000, 'EET');
```

```
INSERT INTO Category (ID, Nazov, Farba, Vek_od, Vek_do) VALUES
```

```
(1, 'Conference', 'Blue', 18, 99),  
(2, 'Exhibition', 'Green', 18, 99),  
(3, 'Webinar', 'Gray', 18, 99),  
(4, 'Workshop', 'Orange', 18, 99),  
(5, 'Seminar', 'Yellow', 18, 99),  
(6, 'Innovation Challenge', 'Cyan', 18, 30),  
(7, 'Career Fair', 'Brown', 20, 99);
```

```
INSERT INTO Event (ID, Nazov, Datum, Cas, Miesto_ID, Category_ID, Author_ID) VALUES
```

```
(1, 'International Tech Conference', '2025-05-10', '09:00:00', 79000, 1, 13),  
(2, 'AI & Robotics Exhibition', '2025-06-15', '10:00:00', 00010, 2, 30),  
(3, 'Web Development Webinar', '2025-04-20', '18:00:00', 94901, 3, 44),  
(4, 'Student Hackathon 2025', '2025-07-05', '10:00:00', 08001, 6, 54),  
(5, 'Coding Challenge', '2025-08-15', '14:00:00', 91701, 6, 42),  
(6, 'Robotics Workshop', '2025-05-25', '11:00:00', 10115, 4, 35),  
(7, 'Tech Talk: Future of AI', '2025-06-05', '16:00:00', 04001, 3, 30),  
(8, 'Innovation Challenge 2025', '2025-09-10', '09:30:00', 81101, 1, 29),  
(9, 'Engineering Career Fair', '2025-10-01', '09:00:00', 08001, 7, 25),  
(10, 'Data Science & Big Data Seminar', '2025-11-18', '17:00:00', 00010, 5, 27);
```

```
INSERT INTO Opakovanie (ID, Typ, Pocet_opakovani, Event_ID) VALUES
```

```
(101, 'mesacne', 5, 1),  
(102, 'tyzdenne', 1, 2),  
(103, 'denne', 1, 3),  
(104, 'denne', 2, 4),  
(105, 'tyzdenne', 12, 5),  
(106, 'mesacne', 1, 6),  
(107, 'denne', 1, 7),  
(108, 'mesacne', 12, 8),  
(109, 'mesacne', 9, 9),  
(110, 'tyzdenne', 2, 10);
```

```
INSERT INTO Pripomienka (ID, Typ, Cas_odoslania, Event_ID) VALUES
```

```
(101, 'email', '2025-05-09 08:00:00', 1),  
(102, 'notifikacia', '2025-06-14 09:00:00', 2),  
(103, 'email', '2025-04-19 17:30:00', 3),  
(104, 'notifikacia', '2025-07-04 09:00:00', 4),  
(105, 'email', '2025-08-14 13:00:00', 5),  
(106, 'notifikacia', '2025-05-24 10:00:00', 6),  
(107, 'email', '2025-06-04 15:30:00', 7),  
(108, 'notifikacia', '2025-09-09 09:00:00', 8),  
(109, 'email', '2025-09-30 08:00:00', 9),  
(110, 'notifikacia', '2025-11-17 16:00:00', 10);
```

```
INSERT INTO ma_pozvanku_na (Event_ID, Pouzivatel_ID, Status) VALUES
```

```
(1, 63, 'potvrdeny'),
```


(2, 63, 'potvrdeny'),
(3, 63, 'potvrdeny'),
(4, 63, 'potvrdeny'),
(5, 63, 'potvrdeny'),
(6, 63, 'potvrdeny'),
(7, 63, 'potvrdeny'),
(8, 63, 'potvrdeny'),
(9, 63, 'potvrdeny'),
(10, 63, 'potvrdeny'),
(1, 64, 'potvrdeny'),
(2, 64, 'potvrdeny'),
(3, 64, 'potvrdeny'),
(4, 64, 'potvrdeny'),
(5, 64, 'potvrdeny'),
(6, 64, 'potvrdeny'),
(7, 64, 'potvrdeny'),
(8, 64, 'potvrdeny'),
(9, 64, 'potvrdeny'),
(10, 64, 'potvrdeny'),
(1, 13, 'potvrdeny'),
(2, 30, 'potvrdeny'),
(3, 44, 'potvrdeny'),
(4, 54, 'potvrdeny'),
(5, 42, 'potvrdeny'),
(6, 35, 'potvrdeny'),
(7, 30, 'potvrdeny'),
(8, 29, 'potvrdeny'),
(9, 25, 'potvrdeny'),
(10, 27, 'potvrdeny'),
(1, 1, 'odmietnuty'),
(2, 1, 'odmietnuty'),
(3, 1, 'odmietnuty'),
(4, 1, 'odmietnuty'),
(5, 1, 'odmietnuty'),
(6, 1, 'odmietnuty'),
(7, 1, 'odmietnuty'),
(8, 1, 'odmietnuty'),
(9, 1, 'odmietnuty'),
(10, 1, 'odmietnuty'),
(1, 2, 'cakajuci'),
(2, 2, 'cakajuci'),
(3, 2, 'cakajuci'),
(4, 2, 'cakajuci'),
(5, 2, 'cakajuci'),
(6, 2, 'cakajuci'),
(7, 2, 'cakajuci'),
(8, 2, 'cakajuci'),
(9, 2, 'cakajuci'),
(10, 2, 'cakajuci'),
(1, 3, 'odmietnuty'),

(2, 3, 'odmietnuty'),
(3, 3, 'odmietnuty'),
(4, 3, 'odmietnuty'),
(5, 3, 'odmietnuty'),
(6, 3, 'odmietnuty'),
(7, 3, 'odmietnuty'),
(8, 3, 'odmietnuty'),
(9, 3, 'odmietnuty'),
(10, 3, 'odmietnuty'),
(7, 37, 'potvrdeny'),
(7, 38, 'potvrdeny'),
(7, 39, 'cakajuci'),
(7, 40, 'cakajuci'),
(7, 41, 'potvrdeny'),
(7, 42, 'cakajuci'),
(7, 43, 'potvrdeny'),
(7, 52, 'potvrdeny'),
(7, 53, 'potvrdeny'),
(7, 54, 'cakajuci'),
(7, 55, 'potvrdeny'),
(7, 56, 'cakajuci'),
(8, 44, 'potvrdeny'),
(8, 45, 'cakajuci'),
(8, 46, 'cakajuci'),
(8, 47, 'cakajuci'),
(8, 48, 'potvrdeny'),
(8, 49, 'cakajuci'),
(8, 50, 'potvrdeny'),
(8, 51, 'cakajuci'),
(8, 57, 'cakajuci'),
(8, 58, 'cakajuci'),
(8, 59, 'potvrdeny'),
(8, 65, 'potvrdeny'),
(8, 66, 'potvrdeny'),
(8, 67, 'potvrdeny'),
(8, 68, 'potvrdeny'),
(2, 60, 'potvrdeny'),
(10, 60, 'cakajuci'),
(2, 61, 'cakajuci'),
(10, 61, 'potvrdeny'),
(2, 62, 'potvrdeny'),
(10, 62, 'cakajuci'),
(2, 65, 'cakajuci'),
(10, 65, 'potvrdeny'),
(2, 66, 'potvrdeny'),
(10, 66, 'cakajuci'),
(2, 67, 'cakajuci'),
(10, 67, 'potvrdeny'),
(2, 68, 'potvrdeny'),
(10, 68, 'cakajuci'),

(2, 13, 'cakajuci'),
(10, 13, 'potvrdeny'),
(2, 36, 'potvrdeny'),
(10, 36, 'cakajuci'),
(4, 8, 'cakajuci'),
(4, 4, 'cakajuci'),
(4, 5, 'cakajuci'),
(8, 37, 'cakajuci'),
(3, 37, 'cakajuci'),
(4, 37, 'cakajuci'),
(5, 37, 'cakajuci'),
(9, 37, 'cakajuci'),
(8, 38, 'cakajuci'),
(3, 38, 'cakajuci'),
(4, 38, 'cakajuci'),
(5, 38, 'cakajuci'),
(9, 38, 'cakajuci'),
(8, 39, 'cakajuci'),
(3, 39, 'cakajuci'),
(4, 39, 'cakajuci'),
(5, 39, 'cakajuci'),
(9, 39, 'cakajuci'),
(8, 40, 'cakajuci'),
(3, 40, 'cakajuci'),
(4, 40, 'cakajuci'),
(5, 40, 'cakajuci'),
(9, 40, 'cakajuci'),
(8, 41, 'cakajuci'),
(3, 41, 'cakajuci'),
(4, 41, 'cakajuci'),
(5, 41, 'cakajuci'),
(9, 41, 'cakajuci'),
(8, 42, 'cakajuci'),
(3, 42, 'cakajuci'),
(4, 42, 'cakajuci'),
(9, 42, 'cakajuci'),
(8, 43, 'cakajuci'),
(3, 43, 'cakajuci'),
(4, 43, 'cakajuci'),
(5, 43, 'cakajuci'),
(9, 43, 'cakajuci'),
(8, 52, 'cakajuci'),
(3, 52, 'cakajuci'),
(4, 52, 'cakajuci'),
(5, 52, 'cakajuci'),
(9, 52, 'cakajuci'),
(8, 53, 'cakajuci'),
(3, 53, 'cakajuci'),
(4, 53, 'cakajuci'),
(5, 53, 'cakajuci'),

(9, 53, 'cakajuci'),
(8, 54, 'cakajuci'),
(3, 54, 'cakajuci'),
(5, 54, 'cakajuci'),
(9, 54, 'cakajuci'),
(8, 55, 'cakajuci'),
(3, 55, 'cakajuci'),
(4, 55, 'cakajuci'),
(5, 55, 'cakajuci'),
(9, 55, 'cakajuci'),
(8, 56, 'cakajuci'),
(3, 56, 'cakajuci'),
(4, 56, 'cakajuci'),
(5, 56, 'cakajuci'),
(9, 56, 'cakajuci'),
(7, 6, 'potvrdeny'),
(3, 6, 'potvrdeny'),
(4, 6, 'potvrdeny'),
(5, 6, 'potvrdeny'),
(9, 6, 'potvrdeny'),
(7, 44, 'potvrdeny'),
(4, 44, 'potvrdeny'),
(5, 44, 'potvrdeny'),
(9, 44, 'potvrdeny'),
(7, 45, 'potvrdeny'),
(3, 45, 'potvrdeny'),
(4, 45, 'potvrdeny'),
(5, 45, 'potvrdeny'),
(9, 45, 'potvrdeny'),
(7, 46, 'potvrdeny'),
(3, 46, 'potvrdeny'),
(4, 46, 'potvrdeny'),
(5, 46, 'potvrdeny'),
(9, 46, 'potvrdeny'),
(7, 57, 'potvrdeny'),
(3, 57, 'potvrdeny'),
(4, 57, 'potvrdeny'),
(5, 57, 'potvrdeny'),
(9, 57, 'potvrdeny'),
(7, 58, 'potvrdeny'),
(3, 58, 'potvrdeny'),
(4, 58, 'potvrdeny'),
(5, 58, 'potvrdeny'),
(9, 58, 'potvrdeny'),
(7, 59, 'potvrdeny'),
(3, 59, 'potvrdeny'),
(4, 59, 'potvrdeny'),
(5, 59, 'potvrdeny'),
(9, 59, 'potvrdeny'),
(7, 47, 'potvrdeny'),

(3, 47, 'potvrdeny'),
(4, 47, 'potvrdeny'),
(5, 47, 'potvrdeny'),
(9, 47, 'potvrdeny'),
(7, 48, 'potvrdeny'),
(3, 48, 'potvrdeny'),
(4, 48, 'potvrdeny'),
(5, 48, 'potvrdeny'),
(9, 48, 'potvrdeny'),
(7, 49, 'potvrdeny'),
(3, 49, 'potvrdeny'),
(4, 49, 'potvrdeny'),
(5, 49, 'potvrdeny'),
(9, 49, 'potvrdeny'),
(7, 50, 'potvrdeny'),
(3, 50, 'potvrdeny'),
(4, 50, 'potvrdeny'),
(5, 50, 'potvrdeny'),
(9, 50, 'potvrdeny'),
(7, 51, 'potvrdeny'),
(3, 51, 'potvrdeny'),
(4, 51, 'potvrdeny'),
(5, 51, 'potvrdeny'),
(9, 51, 'potvrdeny'),
(7, 20, 'potvrdeny'),
(3, 20, 'potvrdeny'),
(4, 20, 'potvrdeny'),
(5, 20, 'potvrdeny'),
(9, 20, 'potvrdeny'),
(7, 21, 'potvrdeny'),
(3, 21, 'potvrdeny'),
(4, 21, 'potvrdeny'),
(5, 21, 'potvrdeny'),
(9, 21, 'potvrdeny'),
(7, 35, 'potvrdeny'),
(3, 35, 'potvrdeny'),
(4, 35, 'potvrdeny'),
(5, 35, 'potvrdeny'),
(9, 35, 'potvrdeny'),
(7, 5, 'potvrdeny'),
(8, 5, 'potvrdeny'),
(3, 5, 'potvrdeny'),
(5, 5, 'potvrdeny'),
(9, 5, 'potvrdeny'),
(7, 8, 'potvrdeny'),
(8, 8, 'potvrdeny'),
(3, 8, 'potvrdeny'),
(5, 8, 'odmietnuty'),
(9, 8, 'odmietnuty'),
(7, 9, 'potvrdeny'),

(8, 9, 'potvrdeny'),
(3, 9, 'potvrdeny'),
(4, 9, 'odmietnuty'),
(5, 9, 'potvrdeny'),
(9, 9, 'potvrdeny'),
(7, 14, 'potvrdeny'),
(8, 14, 'potvrdeny'),
(3, 14, 'odmietnuty'),
(4, 14, 'potvrdeny'),
(5, 14, 'potvrdeny'),
(9, 14, 'potvrdeny'),
(7, 15, 'odmietnuty'),
(8, 15, 'potvrdeny'),
(3, 15, 'potvrdeny'),
(4, 15, 'odmietnuty'),
(5, 15, 'potvrdeny'),
(9, 15, 'potvrdeny'),
(7, 10, 'potvrdeny'),
(8, 10, 'potvrdeny'),
(3, 10, 'potvrdeny'),
(4, 10, 'potvrdeny'),
(5, 10, 'potvrdeny'),
(9, 10, 'potvrdeny'),
(7, 11, 'odmietnuty'),
(8, 11, 'odmietnuty'),
(3, 11, 'odmietnuty'),
(4, 11, 'odmietnuty'),
(5, 11, 'odmietnuty'),
(9, 11, 'odmietnuty'),
(7, 16, 'potvrdeny'),
(8, 16, 'potvrdeny'),
(3, 16, 'potvrdeny'),
(4, 16, 'potvrdeny'),
(5, 16, 'potvrdeny'),
(9, 16, 'potvrdeny'),
(7, 17, 'odmietnuty'),
(8, 17, 'odmietnuty'),
(3, 17, 'odmietnuty'),
(4, 17, 'odmietnuty'),
(5, 17, 'odmietnuty'),
(9, 17, 'odmietnuty'),
(7, 18, 'potvrdeny'),
(8, 18, 'potvrdeny'),
(3, 18, 'potvrdeny'),
(4, 18, 'potvrdeny'),
(5, 18, 'potvrdeny'),
(9, 18, 'potvrdeny'),
(7, 22, 'odmietnuty'),
(8, 22, 'odmietnuty'),
(3, 22, 'odmietnuty'),

(4, 22, 'odmietnutý'),	(8, 25, 'potvrdený'),	(7, 34, 'potvrdený'),
(5, 22, 'odmietnutý'),	(3, 25, 'potvrdený'),	(8, 34, 'potvrdený'),
(9, 22, 'odmietnutý'),	(4, 25, 'potvrdený'),	(3, 34, 'potvrdený'),
(7, 12, 'potvrdený'),	(5, 25, 'potvrdený'),	(4, 34, 'potvrdený'),
(8, 12, 'potvrdený'),	(7, 31, 'potvrdený'),	(5, 34, 'potvrdený'),
(3, 12, 'potvrdený'),	(8, 31, 'potvrdený'),	(9, 34, 'potvrdený'),
(4, 12, 'potvrdený'),	(3, 31, 'potvrdený'),	(7, 29, 'potvrdený'),
(5, 12, 'potvrdený'),	(4, 31, 'potvrdený'),	(3, 29, 'potvrdený'),
(9, 12, 'potvrdený'),	(5, 31, 'potvrdený'),	(4, 29, 'potvrdený'),
(7, 19, 'odmietnutý'),	(9, 31, 'potvrdený'),	(5, 29, 'potvrdený'),
(8, 19, 'odmietnutý'),	(7, 32, 'potvrdený'),	(9, 29, 'potvrdený'),
(3, 19, 'odmietnutý'),	(8, 32, 'potvrdený'),	(8, 30, 'potvrdený'),
(4, 19, 'odmietnutý'),	(3, 32, 'potvrdený'),	(3, 30, 'potvrdený'),
(5, 19, 'odmietnutý'),	(4, 32, 'potvrdený'),	(4, 30, 'potvrdený'),
(9, 19, 'odmietnutý'),	(5, 32, 'potvrdený'),	(5, 30, 'potvrdený'),
(7, 23, 'potvrdený'),	(9, 32, 'potvrdený'),	(9, 30, 'potvrdený'),
(8, 23, 'potvrdený'),	(7, 27, 'potvrdený'),	(1, 60, 'potvrdený'),
(3, 23, 'potvrdený'),	(8, 27, 'potvrdený'),	(1, 61, 'potvrdený'),
(4, 23, 'potvrdený'),	(3, 27, 'potvrdený'),	(1, 62, 'cakajuci'),
(5, 23, 'potvrdený'),	(4, 27, 'potvrdený'),	(1, 65, 'potvrdený'),
(9, 23, 'potvrdený'),	(5, 27, 'potvrdený'),	(1, 66, 'potvrdený'),
(7, 24, 'potvrdený'),	(9, 27, 'potvrdený'),	(1, 67, 'cakajuci'),
(8, 24, 'potvrdený'),	(7, 28, 'potvrdený'),	(1, 68, 'cakajuci'),
(3, 24, 'potvrdený'),	(8, 28, 'potvrdený'),	(1, 69, 'cakajuci'),
(4, 24, 'potvrdený'),	(3, 28, 'potvrdený'),	(1, 70, 'cakajuci'),
(5, 24, 'potvrdený'),	(4, 28, 'potvrdený'),	(6, 60, 'cakajuci'),
(9, 24, 'potvrdený'),	(5, 28, 'potvrdený'),	(6, 61, 'cakajuci'),
(7, 26, 'odmietnutý'),	(9, 28, 'potvrdený'),	(6, 62, 'potvrdený'),
(8, 26, 'odmietnutý'),	(7, 33, 'potvrdený'),	(6, 65, 'potvrdený'),
(3, 26, 'odmietnutý'),	(8, 33, 'potvrdený'),	(6, 66, 'potvrdený'),
(4, 26, 'odmietnutý'),	(3, 33, 'potvrdený'),	(6, 67, 'cakajuci'),
(5, 26, 'odmietnutý'),	(4, 33, 'potvrdený'),	(6, 68, 'cakajuci'),
(9, 26, 'odmietnutý'),	(5, 33, 'potvrdený'),	(6, 69, 'cakajuci'),
(7, 25, 'potvrdený'),	(9, 33, 'potvrdený'),	(6, 70, 'cakajuci');

2 POHLADY S JEDNODUCHÝM NETRIVIÁLNYM SELECTOM NAD JEDNOU TABUĽKOU

-- Pohľad-1: najdi všetkých používateľov, ktorí majú vek väčší ako 30 rokov

-- ziska všetkých používateľov, ktorí majú vek väčší ako 30.

DROP VIEW IF EXISTS Pouzivatel_nad_30;

CREATE VIEW Pouzivatel_nad_30 AS

SELECT ID, Login, Meno, Priezvisko, Vek, Email

FROM Pouzivatel

WHERE Vek > 30;

-- Pohľad-2: najdi všetky kategórie, ktoré sú určené pre vek od 18 do 30 rokov

-- ziska všetky kategórie, ktoré sú určené pre 18-30 rokov.

DROP VIEW IF EXISTS Kategoria_18_30;

```
CREATE VIEW Kategoria_18_30 AS
SELECT ID, Nazov, Farba, Vek_od, Vek_do
FROM Category
WHERE Vek_od >= 18 AND Vek_do <= 30;
```

3 POHLADY SO SPÁJANÍM TABULIEK (1X SPOJENIE ASPOŇ 2 TABULIEK, 1X SPOJENIE ASPOŇ 3 TABULIEK, 1X OUTER JOIN)

```
-- Pohľad-3: najdi všetky udalosti, ktoré sa konajú v mestach na Slovensku
-- ziska všetky udalosti, ktoré sa konajú v mestach Slovenska.
DROP VIEW IF EXISTS Udalosti_v_Slovensku;
CREATE VIEW Udalosti_v_Slovensku AS
SELECT E.ID, E.Nazov, E.Datum, E.Cas, M.Nazov AS Mesto, C.Nazov AS Kategoria
FROM Event E
JOIN Mesto M ON E.Miesto_ID = M.ID
JOIN Category C ON E.Category_ID = C.ID
WHERE M.Krajina = 'Slovakia';
```

```
-- Pohľad-4: najdi všetky udalosti a ich účastníkov
-- ziska všetky udalosti a ich účastníkov.
DROP VIEW IF EXISTS Udalosti_a_Ucastnici;
CREATE VIEW Udalosti_a_Ucastnici AS
SELECT
    E.ID AS Event_ID,
    E.Nazov AS Event_Nazov,
    P.ID AS Pouzivatel_ID,
    P.Meno AS Pouzivatel_Meno,
    P.Priezvisko AS Pouzivatel_Priezvisko,
    M.Status
FROM Event E
JOIN ma_pozvanku_na M ON E.ID = M.Event_ID
JOIN Pouzivatel P ON M.Pouzivatel_ID = P.ID;
```

```
-- Pohľad-5: najdi všetky mesta a počet udalosti, ktoré sa konajú v týchto mestach (LEFT JOIN)
-- použije LEFT JOIN na získanie všetkých miest a počtu udalosti, ktoré sa tam konajú.
DROP VIEW IF EXISTS Najpopularnejsie_Kategorie;
CREATE VIEW Mesta_a_Pocet_Udalosti AS
SELECT M.Nazov AS Mesto, COUNT(E.ID) AS Pocet_Udalosti
FROM Mesto M
LEFT JOIN Event E ON M.ID = E.Miesto_ID
GROUP BY M.Nazov;
```

2 POHLADY S POUŽITÍM AGREGAČNÝCH FUNKCIÍ A/ALEBO ZOSKUPENIA

```
-- Pohľad-6: najdi najpopulárnejšie kategorie podujati podľa počtu účastníkov
-- ziska kategorie podujati zoradene podľa počtu účastníkov, aby sa určili najpopulárnejšie kategorie.
DROP VIEW IF EXISTS Mesta_a_Pocet_Udalosti;
CREATE VIEW Najpopularnejsie_Kategorie AS
SELECT C.Nazov AS Kategoria, COUNT(M.Pouzivatel_ID) AS Pocet_Ucastnikov
FROM Event E
JOIN Category C ON E.Category_ID = C.ID
```

```

JOIN ma_pozvanku_na M ON E.ID = M.Event_ID
WHERE M.Status = 'potvrdeny'
GROUP BY C.Nazov
ORDER BY Pocet_Ucastnikov DESC;

```

```

-- Pohľad-7: najdi všetky udalosti a počet účastníkov pre každú udalosť
-- použije agregáciu f-ciu COUNT na získanie počtu účastníkov každej udalosti.
DROP VIEW IF EXISTS Udalosti_s_Pocet_Ucastnikov;
CREATE VIEW Udalosti_s_Pocet_Ucastnikov AS
SELECT E.ID AS Event_ID, E.Nazov, E.Datum, E.Cas, COUNT(M.Pouzivatel_ID) AS Pocet_Ucastnikov
FROM Event E
JOIN ma_pozvanku_na M ON E.ID = M.Event_ID
GROUP BY E.ID;

```

TRETIA ODOVZDÁVKA

1 POHĽAD S POUŽITÍM MNOŽINOVÝCH OPERÁCIÍ

```

-- Pohľad-8:
-- Získa používateľov, ktorí sú buď autormi udalostí, alebo majú pozvánku na udalosť
DROP VIEW IF EXISTS Pouzivatelia_s_Udalostami;
CREATE VIEW Pouzivatelia_s_Udalostami AS
SELECT ID, Meno, Priezvisko FROM Pouzivatel
WHERE ID IN (SELECT Author_ID FROM Event WHERE Author_ID IS NOT NULL)
UNION
-- unikátne výsledky
SELECT ID, Meno, Priezvisko FROM Pouzivatel
WHERE ID IN (SELECT Pouzivatel_ID FROM ma_pozvanku_na);

```

2 POHĽADY S POUŽITÍM VNORENÝCH PODDOPYTOV

```

-- Pohľad-9:
-- Získa udalosti, ktoré majú nadpriemerný počet potvrdených účastníkov
DROP VIEW IF EXISTS Nadpriemerne_Udalosti;
CREATE VIEW Nadpriemerne_Udalosti AS
SELECT E.ID, E.Nazov, E.Datum, COUNT(M.Pouzivatel_ID) AS Pocet_Ucastnikov
FROM Event E
JOIN ma_pozvanku_na M ON E.ID = M.Event_ID
WHERE M.Status = 'potvrdeny'
GROUP BY E.ID, E.Nazov, E.Datum
HAVING COUNT(M.Pouzivatel_ID) > (
    SELECT AVG(Pocet) FROM (
        SELECT COUNT(*) AS Pocet
        FROM ma_pozvanku_na
        WHERE Status = 'potvrdeny'
        GROUP BY Event_ID
    ) AS Poddotaz
);

```

```
-- Pohľad-10:
-- Získa používateľov, ktorí nemajú žiadnu pozvánku
DROP VIEW IF EXISTS Nepozvani_Pouzivatelia;
CREATE VIEW Nepozvani_Pouzivatelia AS
SELECT * FROM Pouzivatel P
WHERE NOT EXISTS (
    SELECT 1 FROM ma_pozvanku_na M
    WHERE M.Pouzivatel_ID = P.ID
);
```

SKRIPT NA VYTVORENIE TRIGGERU/TRIGGEROV, KTORÉ BUDÚ IMPLEMENTOVAŤ AUTOINKREMENTÁCIU UMELÝCH KLÚČOV

```
-- Autoinkrement pre tabuľku Pouzivatel
DROP SEQUENCE IF EXISTS seq_pouzivatel CASCADE;
CREATE SEQUENCE seq_pouzivatel START 71;

DROP FUNCTION IF EXISTS trg_pouzivatel_function() CASCADE;
CREATE OR REPLACE FUNCTION trg_pouzivatel_function()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.ID IS NULL THEN
        NEW.ID := nextval('seq_pouzivatel');
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

DROP TRIGGER IF EXISTS trg_id_pouzivatel ON Pouzivatel CASCADE;
CREATE TRIGGER trg_id_pouzivatel
BEFORE INSERT ON Pouzivatel
FOR EACH ROW
EXECUTE FUNCTION trg_pouzivatel_function();

-- Autoinkrement pre tabuľku Category
DROP SEQUENCE IF EXISTS seq_category CASCADE;
CREATE SEQUENCE seq_category START 8;

DROP FUNCTION IF EXISTS trg_category_function() CASCADE;
CREATE OR REPLACE FUNCTION trg_category_function()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.ID IS NULL THEN
        NEW.ID := nextval('seq_category');
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

DROP TRIGGER IF EXISTS trg_id_category ON Category CASCADE;
```

```

CREATE TRIGGER trg_id_category
BEFORE INSERT ON Category
FOR EACH ROW
EXECUTE FUNCTION trg_category_function();

-- Autoinkrement pre tabuľku Mesto
DROP SEQUENCE IF EXISTS seq_mesto CASCADE;
CREATE SEQUENCE seq_mesto START 24;

DROP FUNCTION IF EXISTS trg_mesto_function() CASCADE;
CREATE OR REPLACE FUNCTION trg_mesto_function()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.ID IS NULL THEN
        NEW.ID := nextval('seq_mesto');
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

DROP TRIGGER IF EXISTS trg_id_mesto ON Mesto CASCADE;
CREATE TRIGGER trg_id_mesto
BEFORE INSERT ON Mesto
FOR EACH ROW
EXECUTE FUNCTION trg_mesto_function();

-- Autoinkrement pre tabuľku Event
DROP SEQUENCE IF EXISTS seq_event CASCADE;
CREATE SEQUENCE seq_event START 11;

DROP FUNCTION IF EXISTS trg_event_function() CASCADE;
CREATE OR REPLACE FUNCTION trg_event_function()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.ID IS NULL THEN
        NEW.ID := nextval('seq_event');
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

DROP TRIGGER IF EXISTS trg_id_event ON Event CASCADE;
CREATE TRIGGER trg_id_event
BEFORE INSERT ON Event
FOR EACH ROW
EXECUTE FUNCTION trg_event_function();

-- Autoinkrement pre tabuľku Opakovanie
DROP SEQUENCE IF EXISTS seq_opakovanie CASCADE;
CREATE SEQUENCE seq_opakovanie START 110;

```

```

DROP FUNCTION IF EXISTS trg_opakovanie_function() CASCADE;
CREATE OR REPLACE FUNCTION trg_opakovanie_function()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.ID IS NULL THEN
        NEW.ID := nextval('seq_opakovanie');
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

```

```

DROP TRIGGER IF EXISTS trg_id_opakovanie ON Opakovanie CASCADE;
CREATE TRIGGER trg_id_opakovanie
BEFORE INSERT ON Opakovanie
FOR EACH ROW
EXECUTE FUNCTION trg_opakovanie_function();

```

```

-- Autoinkrement pre tabuľku Pripomienka
DROP SEQUENCE IF EXISTS seq_pripomienka CASCADE;
CREATE SEQUENCE seq_pripomienka START 110;

```

```

DROP FUNCTION IF EXISTS trg_pripomienka_function() CASCADE;
CREATE OR REPLACE FUNCTION trg_pripomienka_function()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.ID IS NULL THEN
        NEW.ID := nextval('seq_pripomienka');
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

```

```

DROP TRIGGER IF EXISTS trg_id_pripomienka ON Pripomienka CASCADE;
CREATE TRIGGER trg_id_pripomienka
BEFORE INSERT ON Pripomienka
FOR EACH ROW
EXECUTE FUNCTION trg_pripomienka_function();

```

SKRIPT NA ASPOŇ DVA ZMYSLUPLNÉ TRIGGRE (OKREM TRIGGEROV NA AUTOINKREMENTÁCIU)

```

-- Trigger na aktualizáciu dátumu a času udalosti pri zmene v pohľade 'Udalosti_s_Pocet_Ucastnikov'
DROP FUNCTION IF EXISTS trg_update_event_function;
CREATE OR REPLACE FUNCTION trg_update_event_function()
RETURNS TRIGGER AS $$
BEGIN
    UPDATE Event
    SET Datum = NEW.Datum, Cas = NEW.Cas
    WHERE ID = NEW.Event_ID;

```



```

    RETURN NULL;
END;
$$ LANGUAGE plpgsql;
DROP TRIGGER IF EXISTS trg_update_event ON Udalosti_s_Pocet_Ucastnikov;
CREATE TRIGGER trg_update_event
INSTEAD OF UPDATE ON Udalosti_s_Pocet_Ucastnikov
FOR EACH ROW
EXECUTE FUNCTION trg_update_event_function();

-- Trigger na aktualizáciu statusu pozvánky pri zmene v pohľade 'Udalosti_a_Ucastnici'
DROP FUNCTION IF EXISTS trg_update_status_function;
CREATE OR REPLACE FUNCTION trg_update_status_function()
RETURNS TRIGGER AS $$
BEGIN
    UPDATE ma_pozvanku_na
    SET Status = NEW.Status
    WHERE Event_ID = NEW.Event_ID AND Pouzivatel_ID = NEW.Pouzivatel_ID;
    RETURN NULL;
END;
$$ LANGUAGE plpgsql;
DROP TRIGGER IF EXISTS trg_update_status ON Udalosti_a_Ucastnici;
CREATE TRIGGER trg_update_status
INSTEAD OF UPDATE ON Udalosti_a_Ucastnici
FOR EACH ROW
EXECUTE FUNCTION trg_update_status_function();

```

SKRIPT NA ASPOŇ JEDNU STOROVANÚ PROCEDÚRU

```

-- Procedúra na pridanie nového používateľa a vytvorenie udalosti s týmto používateľom ako autorom
DROP PROCEDURE IF EXISTS pridaj_pouzivatela_s_eventom(
    VARCHAR, VARCHAR, VARCHAR, INT, VARCHAR, VARCHAR, DATE, TIME
);

CREATE OR REPLACE PROCEDURE pridaj_pouzivatela_s_eventom(
    IN p_login VARCHAR,
    IN p_meno VARCHAR,
    IN p_priezvisko VARCHAR,
    IN p_vek INT,
    IN p_email VARCHAR,
    IN p_nazov_eventu VARCHAR,
    IN p_datum DATE,
    IN p_cas TIME
)
LANGUAGE plpgsql
AS $$
DECLARE
    new_user_id INT;
BEGIN
    INSERT INTO Pouzivatel (Login, Meno, Priezvisko, Vek, Email)

```

```
VALUES (p_login, p_meno, p_priezvisko, p_vek, p_email)
RETURNING ID INTO new_user_id;
```

```
INSERT INTO Event (Nazov, Datum, Cas, Miesto_ID, Category_ID, Author_ID)
VALUES (p_nazov_eventu, p_datum, p_cas, 4001,1,new_user_id);
END;
$$;
```

SKRIPT NA ASPOŇ JEDNU FUNKCIU

```
DROP FUNCTION IF EXISTS pocet_eventov_pre_pouzivatela(INT);
CREATE OR REPLACE FUNCTION pocet_eventov_pre_pouzivatela(pouzivatel_id INT)
RETURNS INT AS $$
DECLARE
    pocet INT;
BEGIN
    SELECT COUNT(*) INTO pocet
    FROM Event
    WHERE Author_ID = pouzivatel_id;

    RETURN pocet;
END;
$$ LANGUAGE plpgsql;
```