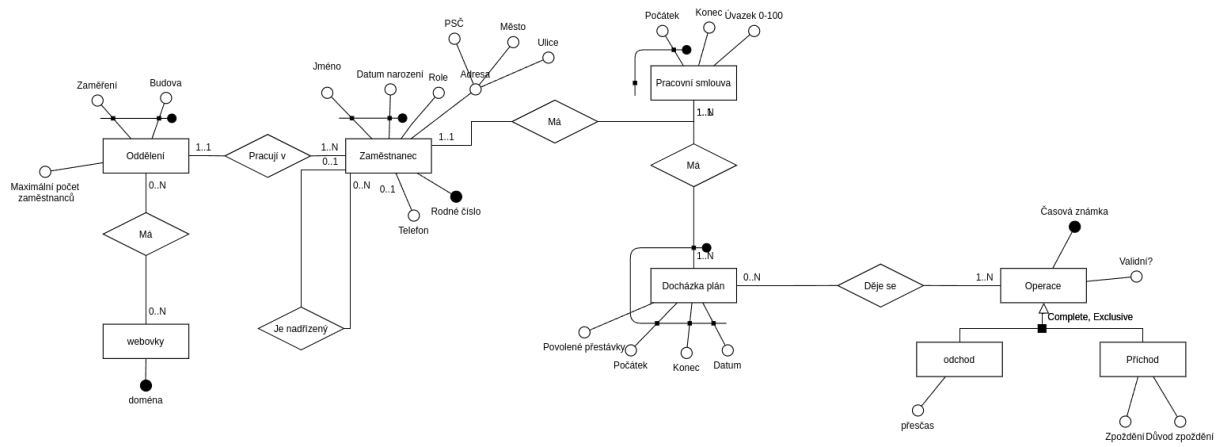
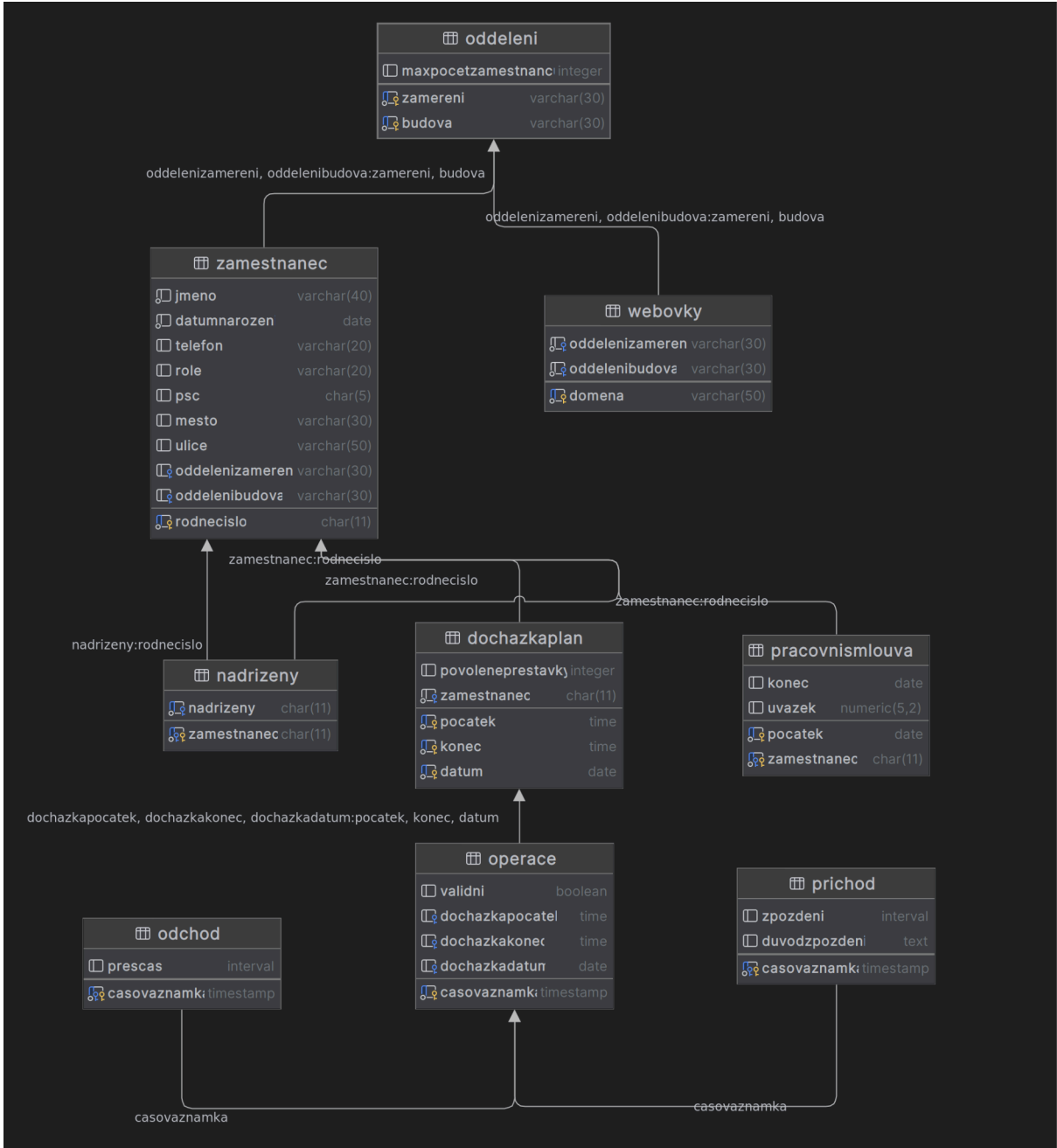


CP3 vondrj47 stolbja2

ER model:



Relační model



Database creation:

```
CREATE TABLE Oddeleni (  
    zamereni VARCHAR(30) NOT NULL,  
    budova VARCHAR(30) NOT NULL,  
    maxPocetZamestnancu INTEGER,  
    PRIMARY KEY (zamereni, budova)  
);
```

```
CREATE TABLE Zamestnanec (  
    rodneCislo CHAR(11) PRIMARY KEY,  
    jmeno VARCHAR(40) NOT NULL,  
    datumNarozeni DATE NOT NULL,  
    telefon VARCHAR(20),  
    role VARCHAR(20),  
    psc CHAR(5),  
    mesto VARCHAR(30),  
    ulice VARCHAR(50),  
    oddeleniZamereni VARCHAR(30),  
    oddeleniBudova VARCHAR(30),  
    CONSTRAINT zamestnanec_fk_odd FOREIGN KEY (oddeleniZamereni,  
oddeleniBudova)  
        REFERENCES Oddeleni(zamereni, budova)  
        ON UPDATE CASCADE  
        ON DELETE SET NULL  
);
```

```
CREATE TABLE Webovky (  
    domena VARCHAR(50) PRIMARY KEY,  
    oddeleniZamereni VARCHAR(30) NOT NULL,  
    oddeleniBudova VARCHAR(30) NOT NULL,  
    CONSTRAINT webovky_fk_odd FOREIGN KEY (oddeleniZamereni,  
oddeleniBudova)  
        REFERENCES Oddeleni(zamereni, budova)  
        ON UPDATE CASCADE  
        ON DELETE CASCADE  
);
```

```
CREATE TABLE Nadrizeny (  
    zamestnanec CHAR(11) PRIMARY KEY,  
    nadrizeny CHAR(11) NOT NULL UNIQUE,  
    CONSTRAINT nadrizeny_fk_zam FOREIGN KEY (zamestnanec)  
        REFERENCES Zamestnanec(rodneCislo)  
        ON UPDATE CASCADE  
        ON DELETE CASCADE,  
    CONSTRAINT nadrizeny_fk_nadr FOREIGN KEY (nadrizeny)  
        REFERENCES Zamestnanec(rodneCislo)  
        ON UPDATE CASCADE
```

```

        ON DELETE CASCADE
    );

CREATE TABLE PracovniSmlouva (
    pocatek DATE,
    zamestnanec CHAR(11) NOT NULL,
    konec DATE,
    uvazek NUMERIC(5,2) CHECK (uvazek >= 0 AND uvazek <= 100),
    PRIMARY KEY (pocatek, zamestnanec),
    CONSTRAINT smlouva_fk_zam FOREIGN KEY (zamestnanec)
        REFERENCES Zamestnanec(rodneCislo)
        ON UPDATE CASCADE
        ON DELETE CASCADE
    );

CREATE TABLE DochazkaPlan (
    pocatek TIME,
    konec TIME,
    datum DATE,
    povolenePrestavky INTEGER,
    zamestnanec CHAR(11) NOT NULL,
    PRIMARY KEY (pocatek, konec, datum),
    CONSTRAINT dochazka_fk_zam FOREIGN KEY (zamestnanec)
        REFERENCES Zamestnanec(rodneCislo)
        ON UPDATE CASCADE
        ON DELETE CASCADE
    );

CREATE TABLE Operace (
    casovaZnamka TIMESTAMP PRIMARY KEY,
    validni BOOLEAN,
    dochazkaPocatek TIME,
    dochazkaKonec TIME,
    dochazkaDatum DATE,
    CONSTRAINT operace_fk_dochazka FOREIGN KEY (dochazkaPocatek,
    dochazkaKonec, dochazkaDatum)
        REFERENCES DochazkaPlan(pocatek, konec, datum)
        ON UPDATE CASCADE
        ON DELETE CASCADE
    );

CREATE TABLE Prichod (
    casovaZnamka TIMESTAMP PRIMARY KEY,
    zpozdeni INTERVAL,
    duvodZpozdeni TEXT,
    CONSTRAINT prichod_fk_operace FOREIGN KEY (casovaZnamka)
        REFERENCES Operace(casovaZnamka)
        ON UPDATE CASCADE

```

```
                ON DELETE CASCADE
);

CREATE TABLE Odchod (
    casovaZnamka TIMESTAMP PRIMARY KEY,
    prescas INTERVAL,
    CONSTRAINT odchod_fk_operace FOREIGN KEY (casovaZnamka)
        REFERENCES Operace(casovaZnamka)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);
```

Privileges:

```
GRANT CONNECT ON DATABASE vondrj47 TO stolbja2;
GRANT USAGE ON SCHEMA public TO stolbja2;
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA public
TO stolbja2;
ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT SELECT, INSERT,
UPDATE, DELETE ON TABLES TO stolbja2;
```

Data queries:

-- Oddeleni

```
INSERT INTO Oddeleni (zamereni, budova, maxPocetZamestnancu) VALUES
    ('IT', 'A1', 10),
    ('HR', 'B1', 5),
    ('Finance', 'C1', 8);
```

-- Zamestnanec

```
INSERT INTO Zamestnanec (rodneCislo, jmeno, datumNarozeni, telefon, role, psc, mesto,
ulice, oddeleniZamereni, oddeleniBudova) VALUES
    ('85010112345', 'Jan Novak', '1985-01-01', '777111222', 'Programator', '11000',
    'Praha', 'Vinohradska 1', 'IT', 'A1'),
    ('86020223456', 'Eva Novakova', '1986-02-02', '777222333', 'HR Specialist', '12000',
    'Brno', 'Masarykova 10', 'HR', 'B1'),
    ('87030334567', 'Petr Svoboda', '1987-03-03', '777333444', 'Ucetni', '13000',
    'Ostrava', 'Hlavni 5', 'Finance', 'C1'),
    ('88040445678', 'Lucie Dvorakova', '1988-04-04', '777444555', 'Programator', '14000',
    'Plzen', 'Klatovska 20', 'IT', 'A1'),
    ('89050556789', 'Karel Prochazka', '1989-05-05', '777555666', 'HR Manager', '15000',
    'Liberec', 'Nerudova 3', 'HR', 'B1');
```

```
INSERT INTO Zamestnanec (rodneCislo, jmeno, datumNarozeni, telefon, role, psc, mesto,
ulice, oddeleniZamereni, oddeleniBudova) VALUES
    ('85010110987', 'Luboš Novotný', '1982-05-22', '734934857', 'Zaměstnanec', '11000',
    'Praha', 'Plzenska 3', 'HR', 'B1');
```

-- PracovniSmlouva

```
INSERT INTO PracovniSmlouva (pocatek, zamestnanec, konec, uvazek) VALUES
    ('2022-01-01', '85010112345', '2025-01-01', 100.0),
    ('2023-03-15', '86020223456', NULL, 80.0),
    ('2023-06-01', '87030334567', NULL, 90.0);
```

-- DochazkaPlan

```
INSERT INTO DochazkaPlan (pocatek, konec, datum, povolenePrestavky, zamestnanec)
VALUES
    ('08:00', '16:00', '2024-04-01', 2, '85010112345'),
    ('09:00', '17:00', '2024-04-01', 1, '86020223456'),
    ('08:30', '16:30', '2024-04-01', 3, '87030334567');
```

-- Operace – testovací vzorek (menší sada + velký generátor)

```
INSERT INTO Operace (casovaZnamka, validni, dochazkaPocatek, dochazkaKonec,
dochazkaDatum) VALUES
    ('2024-04-01 08:03:00', TRUE, '08:00', '16:00', '2024-04-01'),
    ('2024-04-01 16:02:00', TRUE, '08:00', '16:00', '2024-04-01'),
    ('2024-04-01 09:05:00', FALSE, '09:00', '17:00', '2024-04-01'),
    ('2024-04-01 08:35:00', TRUE, '08:30', '16:30', '2024-04-01');
```

-- Prichod

```
INSERT INTO Prichod (casovaZnamka, zpozdeni, duvodZpozdeni) VALUES
    ('2024-04-01 08:03:00', '00:03:00', 'Doprava'),
    ('2024-04-01 09:05:00', '00:05:00', 'Zaspani');
```

-- Odchod

```
INSERT INTO Odchod (casovaZnamka, prescas) VALUES
    ('2024-04-01 16:02:00', '00:02:00');
```

-- Webovky

```
INSERT INTO Webovky (domena, oddeleniZamereni, oddeleniBudova) VALUES
    ('it.firma.cz', 'IT', 'A1'),
    ('hr.firma.cz', 'HR', 'B1');
```

-- Nadrizeny

```
INSERT INTO Nadrizeny (zamestnanec, nadrizeny) VALUES
    ('85010112345', '88040445678'),
    ('86020223456', '89050556789');
```

– **Filling with querie:**

```
INSERT INTO DochazkaPlan (pocatek, konec, datum, povolenePrestavky, zamestnanec)
SELECT
    '08:00',
    '16:00',
    date '2024-01-01' + (n / 480)::int,
    2,
    '85010112345'
FROM generate_series(1, 32000) AS n
GROUP BY date '2024-01-01' + (n / 480)::int;
```

```
INSERT INTO Operace (casovaZnamka, validni, dochazkaPocatek, dochazkaKonec,
dochazkaDatum)
```

```
SELECT
    timestamp '2024-01-01 08:00:00' + (n || ' minutes')::interval,
    (random() > 0.1),
    '08:00',
    '16:00',
    date '2024-01-01' + (n / 480)::int
FROM generate_series(1, 32000) AS n;
```

SQL dotazy: doplnit popisky a obrázky, doplnit dotaz na Definování oprávnění pro druhého ze skupiny (GRANT)

1)

Tento dotaz vypisuje jména a rodná čísla všech zaměstnanců spolu se zaměřením a budovou jejich oddělení. Používá INNER JOIN, což znamená, že vypíše pouze ty zaměstnance, kteří mají přiřazené odpovídající oddělení (na základě kombinace zamereni a budova).

```
SELECT z.jmeno, z.rodneCislo, o.zamereni, o.budova
FROM Zamestnanec z
      INNER JOIN Oddeleni o
      ON z.oddeleniZamereni = o.zamereni AND z.oddeleniBudova = o.budova;
```

jmeno	rodneCislo	zamereni	budova
Lucie Dvorakova	88040445678	IT	A1
Jan Novak	85010112345	IT	A1
Luboš Novotný	85010110987	HR	B1
Karel Prochazka	89050556789	HR	B1
Eva Novakova	86020223456	HR	B1
Petr Svoboda	87030334567	Finance	C1

2)

Dotaz vrací seznam všech zaměstnanců a jejich docházkový plán (datum, začátek, konec). I když zaměstnanec nemá žádný docházkový záznam, zobrazí se (díky LEFT JOIN), přičemž hodnoty z DochazkaPlan budou NULL.

```
SELECT z.jmeno, z.rodneCislo, d.datum, d.pocatek, d.konec
FROM Zamestnanec z
      LEFT JOIN DochazkaPlan d
      ON z.rodneCislo = d.zamestnanec;
```

jmeno	rodneCislo	datum	pocatek	konec
Jan Novak	85010112345	2024-04-01	08:00:00	16:00:00
Eva Novakova	86020223456	2024-04-01	09:00:00	17:00:00
Petr Svoboda	87030334567	2024-04-01	08:30:00	16:30:00
Jan Novak	85010112345	2024-02-29	08:00:00	16:00:00
Jan Novak	85010112345	2024-02-16	08:00:00	16:00:00
Jan Novak	85010112345	2024-01-04	08:00:00	16:00:00
Jan Novak	85010112345	2024-01-01	08:00:00	16:00:00

3)

Jednoduchý dotaz, který vypisuje všechny zaměstnance, kteří mají ve sloupci mesto uvedeno "Praha".

```
SELECT * FROM Zamestnanec
```


WHERE mesto = 'Praha';

rodneCislo	jmeno	datumNarozeni	telefon	role	psc	mesto	ulice	oddeleniZamereni	oddeleniBudova
85010112345	Jan Novak	1985-01-01	777111222	Programator	11000	Praha	Vinohradska 1	IT	A1
85010110987	Luboš Novotný	1982-05-22	734934857	Zaměstnanec	11000	Praha	Plzenska 3	HR	B1

4)

Vrací 10 zaměstnanců od čtvrtého v abecedním pořadí podle jména (OFFSET 3 znamená přeskočení prvních tří řádků). Používá se například pro stránkování v aplikacích.

```
SELECT * FROM Zamestnanec
ORDER BY jmeno
LIMIT 10 OFFSET 3;
```

rodneCislo	jmeno	datumNarozeni	telefon	role	psc	mesto	ulice	oddeleniZamereni	oddeleniBudova
85010110987	Luboš Novotný	1982-05-22	734934857	Zaměstnanec	11000	Praha	Plzenska 3	HR	B1
88040445678	Lucie Dvorakova	1988-04-04	777444555	Programator	14000	Plzen	Klatovska 20	IT	A1
87030334567	Petr Svoboda	1987-03-03	777333444	Ucetni	13000	Ostrava	Hlavní 5	Finance	C1

5)

Vypíše jména zaměstnanců, kteří bydlí v Praze, ale **nepracují** v oddělení s zamereni = 'IT'. Operace EXCEPT vrací rozdíl mezi dvěma množinami.

```
SELECT jmeno FROM Zamestnanec WHERE mesto = 'Praha'
EXCEPT
SELECT jmeno FROM Zamestnanec WHERE oddeleniZamereni = 'IT';
```

jmeno
Luboš Novotný

6)

Dotaz vrací jména zaměstnanců, kteří se účastnili **více než dvou operací** (pravděpodobně pracovních úkonů), přičemž operace se spojují s docházkovými záznamy podle data.

```
SELECT jmeno FROM Zamestnanec
WHERE rodneCislo IN (
    SELECT dochazkaZam FROM (
        SELECT d.zamestnanec AS dochazkaZam, COUNT(o.*) AS operace_count
        FROM DochazkaPlan d
        JOIN Operace o ON o.dochazkaDatum = d.datum
        GROUP BY d.zamestnanec
    ) sub
WHERE operace_count > 2
);
```

jmeno
Jan Novak
Eva Novakova
Petr Svoboda

7)

Vrací seznam oprávnění (SELECT, INSERT, UPDATE apod.), která má role nebo uživatel stolbja2 na konkrétní tabulky. Hodí se pro audit práv v databázi.

```
SELECT grantee, privilege_type, table_schema, table_name
FROM information_schema.role_table_grants
WHERE grantee = 'stolbja2';
```

grantee	privilege_type	table_schema	table_name
stolbja2	INSERT	public	nadrizeny
stolbja2	SELECT	public	nadrizeny
stolbja2	UPDATE	public	nadrizeny
stolbja2	DELETE	public	nadrizeny
stolbja2	INSERT	public	pracovnislouva
stolbja2	SELECT	public	pracovnislouva
stolbja2	UPDATE	public	pracovnislouva
stolbja2	DELETE	public	pracovnislouva
stolbja2	INSERT	public	dochazkaplan
stolbja2	SELECT	public	dochazkaplan
stolbja2	UPDATE	public	dochazkaplan
stolbja2	DELETE	public	dochazkaplan
stolbja2	INSERT	public	operace
stolbja2	SELECT	public	operace