

Mesterséges intelligencia használata a szoftverfejlesztésben

CadMap Kft.
Szabó József
Szabo.jozsef@cadmap.hu

A CadMap Kft. bemutatása

1996-ban alapították magánszemélyek.

Fő tevékenységi körök:

- Térképek digitalizálása
- Digitális adatok átalakítása konvertálása, transzformálása
- Digitális térképi állományok vizsgálata, minőség-ellenőrzés, javítás
- Technológiák kidolgozása és fejlesztése (pl.: digitalizálás, változásvezetés)
- Adatbázisok tervezése, kialakítása, feltöltése, karbantartása
- Szoftverfejlesztés
-
- ChatGPT
- Bing
- ChatGPT magyarul (Facebook csoport)

Foszforgézu 2023





Midjourney képek



A képeket készítette: Szabó Krisztián szabo.krisztián@cadmap.hu

Foszforgézu 2023





DGN fájl betöltése PostgreSQL adatbázisba Python programmal

Kérlek írd egy olyan programot amely PostgreSQL adatbázisba betölti egy MicroStation DGN fájl tartalmát. A programot Python nyelven írd. A dgn fájlban text, line, linestring és circle elemek lehetnek. Az adatbázisban a különböző elemek külön-külön táblába kerüljenek. Az elemek geometriáján kívül a réteg és a szín adatokat is be kell tölteni. A program az adatbázisban hozza létre a táblákat.

Természetesen! Itt van egy példa program, amely betölti egy MicroStation DGN fájl tartalmát egy PostgreSQL adatbázisba. A program Python nyelven íródott, és a psycopg2 modult használja a PostgreSQL adatbázis kapcsolat létrehozásához és a lekérdezések végrehajtásához. Biztosítani kell, hogy a psycopg2 modul telepítve legyen a Python környezetben a program helyes működéséhez.



DGN fájl betöltése PostgreSQL adatbázisba Python programmal

python

```
import psycopg2
from psycopg2.extensions import AsIs
from pydgnlib import dgnlib

def create_tables(conn):
    cursor = conn.cursor()

    # Text elemek tábla létrehozása
    cursor.execute("""
        CREATE TABLE IF NOT EXISTS text_elements (
            id SERIAL PRIMARY KEY,
            layer TEXT,
            color TEXT,
            text TEXT,
            size FLOAT,
            direction FLOAT,
            geometry GEOMETRY(Point, 2)
        )
    """)
```



DGN fájl betöltése PostgreSQL adatbázisba Python programmal

```
# Line elemek tábla létrehozása
cursor.execute("""
CREATE TABLE IF NOT EXISTS line_elements (
    id SERIAL PRIMARY KEY,
    layer TEXT,
    color TEXT,
    geometry GEOMETRY(LineString, 2)
)
""")

# Linestring elemek tábla létrehozása
cursor.execute("""
CREATE TABLE IF NOT EXISTS linestring_elements (
    id SERIAL PRIMARY KEY,
    layer TEXT,
    color TEXT,
    geometry GEOMETRY(LineString, 2)
)
""")
```



DGN fájl betöltése PostgreSQL adatbázisba Python programmal

```
#  
# Circle elemek tábla létrehozása  
cursor.execute("""  
    CREATE TABLE IF NOT EXISTS circle_elements (  
        id SERIAL PRIMARY KEY,  
        layer TEXT,  
        color TEXT,  
        radius FLOAT,  
        geometry GEOMETRY(Point, 2)  
    )  
""")  
  
cursor.close()  
conn.commit()
```



DGN fájl betöltése PostgreSQL adatbázisba Python programmal

```
def load_dgn_to_postgresql(dgn_file, db_name, db_user, db_password, db_host, db_port):  
    # Kapcsolódás az adatbázishoz  
    conn = psycopg2.connect(database=db_name, user=db_user, password=db_password, host=db_host,  
port=db_port)  
  
    # Táblák létrehozása  
    create_tables(conn)  
  
    cursor = conn.cursor()  
  
    # DGN fájl betöltése  
    dgn = dgnlib.DgnFile()  
    dgn.read(dgn_file)  
  
    for element in dgn.iter_elements():  
        if element.type == 'text':  
            # Text elemek kezelése  
            layer = element.layer  
            color = element.color  
            text = element.text  
            size = element.size  
            direction = element.direction  
            geometry = element.geometry  
  
            # SQL INSERT utasítás végrehajtása a text elemek táblájába  
            cursor.execute("""  
                INSERT INTO text_elements (layer, color, text, size, direction, geometry)  
                VALUES (%s, %s, %s, %s, %s, ST_SetSRID(ST_MakePoint(%s, %s), 23700))  
                """ (layer color text size direction geometry_x geometry_y))
```




DGN fájl betöltése PostgreSQL adatbázisba Python programmal

```
elif element.type == 'line':  
    # Vonal elemek kezelése  
    layer = element.layer  
    color = element.color  
    geometry = element.geometry  
  
    # SQL INSERT utasítás végrehajtása a line elemek táblájába  
    cursor.execute("""  
        INSERT INTO line_elements (layer, color, geometry)  
        VALUES (%s, %s, ST_SetSRID(ST_GeomFromText(%s), 23700))  
        """, (layer, color, geometry.wkt))  
  
elif element.type == 'linestring':  
    # Linestring elemek kezelése  
    layer = element.layer  
    color = element.color  
    geometry = element.geometry  
  
    # SQL INSERT utasítás végrehajtása a linestring elemek táblájába  
    cursor.execute("""  
        INSERT INTO linestring_elements (layer, color, geometry)  
        VALUES (%s, %s, ST_SetSRID(ST_GeomFromText(%s), 23700))  
        """, (layer, color, geometry.wkt))
```



DGN fájl betöltése PostgreSQL adatbázisba Python programmal

```
for element in dgn.iter_elements():
    if element.type == 'circle':
        # Text elemek kezelése
        layer = element.layer
        color = element.color
        radius = element.radius
        geometry = element.geometry

        # SQL INSERT utasítás végrehajtása a circle elemek táblájába
        cursor.execute("""
            INSERT INTO circle_elements (layer, color, radius, geometry)
            VALUES (%s, %s, %s, ST_SetSRID(ST_MakePoint(%s, %s), 23700))
            """, (layer, color, radius, geometry.x, geometry.y))

        # Tranzakciók mentése és kapcsolat lezárása
        conn.commit()
        cursor.close()
        conn.close()

print("A DGN fájl sikeresen betöltve az adatbázisba.")
```

```
# A program futtatása
load_dgn_to_postgresql('path/to/your/dgn/file.dgn', 'your_db_name', 'your_db_user', 'your_db_password',
'your_db_host', 'your_db_port')
```



Bing képek



A képeket készítette: Trudics Gabriella trudics.gabi@cadmap.hu

Foszforgézu 2023





DWG fájl betöltése PostgreSQL adatbázisba Python programmal

Kérlek írd egy olyan programot amely PostgreSQL adatbázisba betölti egy AutoCAD DWG fájl tartalmát. A programot Python nyelven írd. A dwg fájlban text, line és circle elemek lehetnek. Az adatbázisban a különböző elemek külön külön táblába kerüljenek. Az elemek geometriáján kívül a réteg és a szín adatokat is be kell tölteni. A program az adatbázisban hozza létre a táblákat.

Természetesen! Az alábbiakban egy egyszerű Python programot találsz, amely egy AutoCAD DWG fájlt olvas be és az adatokat egy PostgreSQL adatbázisba menti. A program a psycopg2 modult használja a Python és PostgreSQL közötti kapcsolat létrehozásához és az adatok betöltéséhez. Először telepítened kell a psycopg2 modult a pip segítségével:

```
bash
```

```
pip install psycopg2
```

Foszforgézu 2023

Az alábbi kód felhasználja a psycopg2 modult, valamint a





DWG fájl betöltése PostgreSQL adatbázisba Python programmal

```
python
import ezdxf
import psycopg2

def create_tables(conn):
    cursor = conn.cursor()

    # Tábla létrehozása a text elemek számára
    cursor.execute("""
        CREATE TABLE IF NOT EXISTS texts (
            id SERIAL PRIMARY KEY,
            content TEXT,
            layer TEXT,
            color TEXT,
            geometry GEOMETRY(Point, 2)
        )
    """)

    # Tábla létrehozása a line elemek számára
    cursor.execute("""
        CREATE TABLE IF NOT EXISTS lines (
            id SERIAL PRIMARY KEY,
            layer TEXT,
            color TEXT,
            geometry GEOMETRY(LineString, 2)
        )
    """)
```



DWG fájl betöltése PostgreSQL adatbázisba Python programmal

```
# Tábla létrehozása a circle elemek számára
```

```
cursor.execute("""  
    CREATE TABLE IF NOT EXISTS circles (  
        id SERIAL PRIMARY KEY,  
        layer TEXT,  
        color TEXT,  
        geometry GEOMETRY(Point, 2),  
        radius DOUBLE PRECISION  
    )  
""")
```

```
conn.commit()
```

```
def insert_text(conn, content, layer, color, point):
```

```
    cursor = conn.cursor()  
    cursor.execute("""  
        INSERT INTO texts (content, layer, color, geometry)  
        VALUES (%s, %s, %s, ST_SetSRID(ST_MakePoint(%s, %s), 4326))  
        """, (content, layer, color, point[0], point[1]))
```

```
    conn.commit()
```



DWG fájl betöltése PostgreSQL adatbázisba Python programmal

```
def insert_line(conn, layer, color, start_point, end_point):
    cursor = conn.cursor()
    cursor.execute("""
        INSERT INTO lines (layer, color, geometry)
        VALUES (%s, %s, ST_SetSRID(ST_MakeLine(
            ST_MakePoint(%s, %s),
            ST_MakePoint(%s, %s)
        ), 4326))
        """, (layer, color, start_point[0], start_point[1], end_point[0], end_point[1]))

    conn.commit()

def insert_circle(conn, layer, color, center_point, radius):
    cursor = conn.cursor()
    cursor.execute("""
        INSERT INTO circles (layer, color, geometry, radius)
        VALUES (%s, %s, ST_SetSRID(ST_MakePoint(%s, %s), 4326), %s)
        """, (layer, color, center_point[0], center_point[1], radius))

    conn.commit()
```



DWG fájl betöltése PostgreSQL adatbázisba Python programmal

```
def load_dwg_to_postgresql(dwg_file, db_name, db_user, db_password, db_host, db_port):  
    conn = psycopg2.connect(  
        dbname=db_name,  
        user=db_user,  
        password=db_password,  
        host=db_host,  
        port=db_port  
    )  
  
    create_tables(conn)  
  
    doc = ezdxf.readfile(dwg_file)  
  
    for entity in doc.modelspace():  
        if entity.dxftype() == 'TEXT':  
            insert_text(conn, entity.dxf.text, entity.dxf.layer, entity.dxf.color, entity.dxf.insert[:2])  
        elif entity.dxftype() == 'LINE':  
            insert_line(conn, entity.dxf.layer, entity.dxf.color, entity.dxf.start[:2], entity.dxf.end[:2])  
        elif entity.dxftype() == 'CIRCLE':  
            insert_circle(conn, entity.dxf.layer, entity.dxf.color, entity.dxf.center[:2], entity.dxf.radius)  
  
    conn.close()
```




DWG fájl betöltése PostgreSQL adatbázisba Python programmal

A PostgreSQL adatbázis beállításai

```
db_name = 'adatbázis_nev'
```

```
db_user = 'felhasználónev'
```

```
db_password = 'jelszo'
```

```
db_host = 'localhost'
```

```
db_port = '5432'
```

A DWG fájl elérési útvonala

```
dwg_file = 'path/to/your/file.dwg'
```

DWG fájl betöltése a PostgreSQL adatbázisba

```
load_dwg_to_postgresql(dwg_file, db_name, db_user, db_password, db_host, db_port)
```

A fenti kód beolvassa a DWG fájlt az ezdxf modullal, majd az egyes entitásokat (text, line, circle) a megfelelő táblákba tölti a PostgreSQL adatbázisba. Az adatbázis nevét, felhasználónevet, jelszót, hosztot és portot meg kell adnod a kód megfelelő részében. A DWG fájl elérési útvonalát is meg kell adnod a `dwg_file` változóban.



Midjourney képek



A képeket készítette: Szabó Krisztián szabo.krisztián@cadmap.hu

Foszforgézu 2023





Az AI számos területen felhasználható a térinformatikában

A CadMap kft. térinformatikával foglalkozik, azt kérdezem, hogy milyen területeken tudnánk használni a mesterséges intelligencia lehetőségeit?

Térképek és képek automatikus osztályozása és értelmezése

például lehetőség van az erdőterületek, a víztestek, a mezőgazdasági területek stb. automatikus azonosítására és kategorizálására.

Útvonaltervezés és navigáció

az AI alkalmazása lehetővé teszi az útvonaltervezési folyamat javítását és pontosabbá tételét, valamint az útvonalak optimalizálását a forgalom, a sebességhatárok és más tényezők figyelembevételével.



Az AI számos területen felhasználható a térinformatikában

Vészkezelés

az AI lehetővé teszi a válságkezelési folyamatok javítását, például a természeti katasztrófák vagy a járványok során, a kockázatbecslési folyamatok automatizálásával, a riasztási rendszerek javításával és a mentési műveletek koordinálásával.

Jövőbeli előrejelzések és tervezés

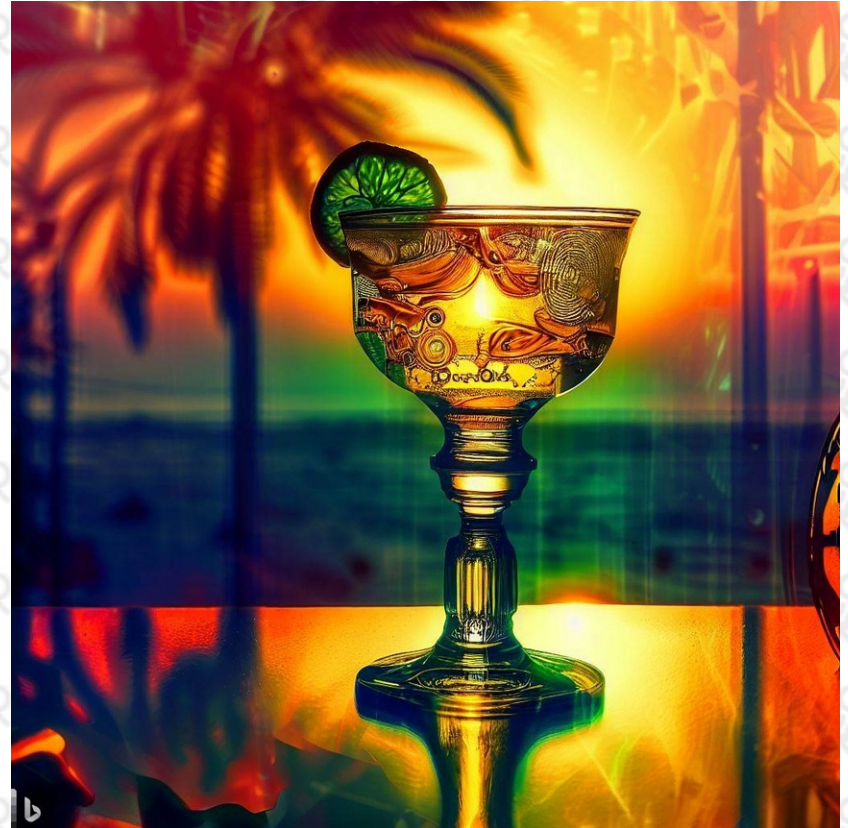
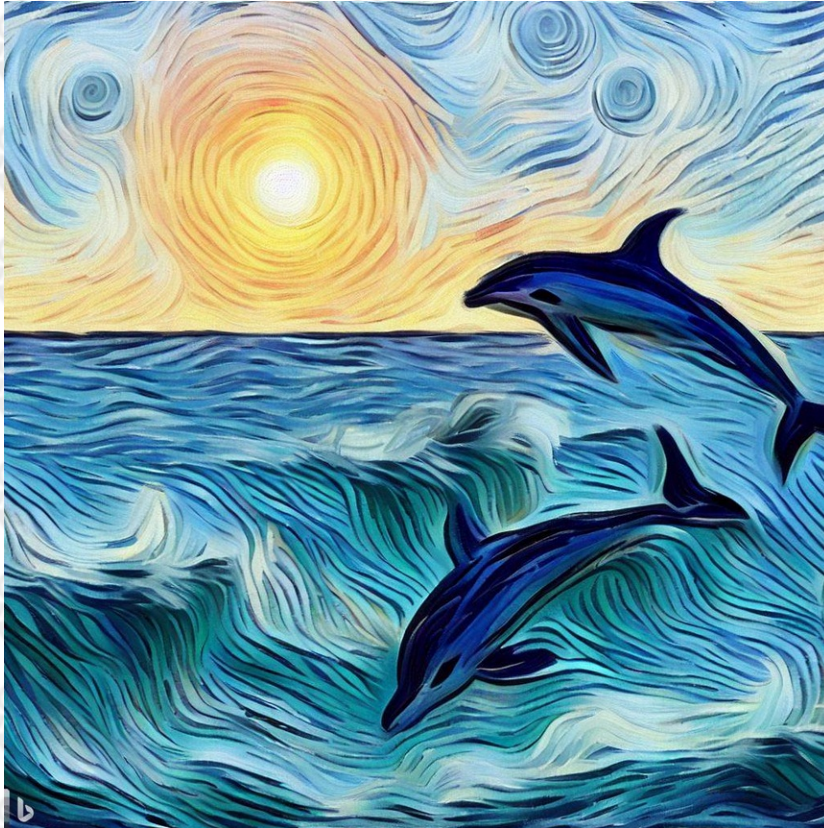
az AI lehetővé teszi a városi fejlesztési tervek, közlekedési infrastruktúra-tervezések és más területek előrejelzését a térinformatikai adatok elemzése alapján.

Automatikus felismerés és korrekció

az AI használata lehetővé teszi a hibák automatikus azonosítását és javítását, valamint az adatok hiányosságainak és pontatlanságainak javítását.



Bing képek



A képeket készítette: Trudics Gabriella trudics.gabi@cadmap.hu

Foszforgézu 2023



Köszönöm a figyelmet!

CadMap Kft.
Szabó József
Szabo.jozsef@cadmap.hu

Foszforgézu 2023