Trombini_Quentin_Week6

S06_01

S06_01_a

```
import requests
import sqlite3
import contextlib
import os

db_file = "./Astro/S06/GLARE.db"

if os.path.isfile("./Astro/S06/data.txt") == False:
    with open("./Astro/S06/data.txt","w") as f:
        f.write(str(requests.get("http://glade.elte.hu/GLADE_2.4.txt").content)[2:])

with open("./Astro/S06/data.txt","r") as f:
    lines = f.readline()
lines = lines.split("\\n")

sql_maketable = f"create table GLARE(gnumber integer primary key, npgc integer, gwgc_name text, leda_name text, mass_name text, wise_name t
with contextlib.closing(sqlite3.connect(db_file)) as conn :
    cursor = conn.cursor()
    try:
        cursor.execute(sql_maketable)
    except:
        print("table already existing")
```

S06_01_c

First of all we us the os library to check if the data is already downloaded, if not we downloaded and write it on the data.txt file

once this is done we load all the data into a variable

Once the data preprocessing is done we write ou SQL request in the sql_maketable variable and then we try to create the table.

S06_02

S06_02_a

```
for line in lines:
        line = line.split()
        try:
            gnumber = int(line[0])
        except:
            continue
        try:
            npgc = int(line[1])
        except:
            npgc = 9999999999999999
        try:
            gwgc_name = line[2]
        except:
            gwgc_name = "null"
        try:
            leda_name = line[3]
        except:
            leda_name = "null"
        try:
            mass_name = line[4]
        except:
            mass_name = "null"
        try:
```

```
            wise_name = line[5]
        except:
            wise_name = "null"
        try:
            sdss_name = line[6]
        except:
            sdss_name = "null"
        try:
            flag = line[7]
        except:
            flag = "null"
        try:
            ra = float(line[8])
        except:
            ra = 9999999999999999.99
        try:
            dec = float(line[9])
        except:
            dec = 9999999999999999.99
        try:
            b = float(line[10])
        except:
            b = 9999999999999999.99
        try:
            b_real = float(line[11])
        except:
            b_real = 99999999999999999.99
        try:
            b_err = float(line[12])
        except:
            b_err = 9999999999999999.99
        try:
            b_flag = int(line[13])
        except:
            b_flag = 9999999999999999
        try:
            b_abs = float(line[14])
        except:
            b_abs = 9999999999999999.99
        try:
            j = float(line[15])
        except:
            j = 999999999999999.99
        try:
            j_err = float(line[16])
        except:
            j_err = 9999999999999999.99
        try:
            h = float(line[17])
        except:
            h = 9999999999999999.99
        try:
            h_err = float(line[18])
        except:
            h_err = 9999999999999999.99
        try:
            k = float(line[19])
        except:
            k = 9999999999999999.99
        try:
            k_err = float(line[20])
        except:
            k_err = 9999999999999999.99
        try:
            w1 = float(line[21])
        except:
            w1 = 9999999999999999.99
        try:
            w1_err = float(line[22])
        except:
            w1_err = 9999999999999999.99
        try:
            w2 = float(line[23])
        except:
            w2 = 9999999999999999.99
        try:
            w2_err = float(line[24])
        except:
```

```
            w2_err = 9999999999999999.99
        try:
            w1_flag = int(line[25])
        except:
            w1_flag = 9999999999999
        try:
            b_j = float(line[26])
        except:
            b_j = 9999999999999999.99
        try:
            b_j_err = float(line[27])
        except:
            b_j_err = 99999999999999999.99
        try:
            z_helio = float(line[28])
        except:
            z_helio = 99999999999999999.99
        try:
            z_cmb = float(line[29])
        except:
            z_cmb = 99999999999999999.99
        try:
            z_flag = int(line[30])
        except:
            z_flag = 9999999999999999
        try:
            v_err = float(line[31])
        except:
            v_err = 99999999999999999.99
        try:
            z_err = float(line[32])
        except:
            z_err = 99999999999999999.99
        try:
            d_l = float(line[33])
        except:
            d_l = 9999999999999999.99
        try:
            d_l_err = float(line[34])
        except:
            d_l_err = 99999999999999999.99
        try:
            dist_flag = int(line[35])
        except:
            dist_flag = 9999999999999999
        try:
            mstar = float(line[36])
        except:
            mstar = 99999999999999999.99
        try:
            mstar_err = float(line[37])
        except:
            mstar_err = 99999999999999999.99
        try:
            mstar_flag = int(line[38])
        except:
            mstar_flag = 9999999999999999
        try:
            merger_rate = float(line[39])
        except:
            merger_rate = 99999999999999999.99
        try:
            merger_rate_err = float(line[40])
        except:
            merger_rate_err = 99999999999999999.99

        sql_adddata = f"insert into GLARE values({gnumber},{npgc},'{gwgc_name}','{leda_name}','{mass_name}','{wise_name}','{sdss_name}','{f
        cursor.execute(sql_adddata)
```

S06_02_c

Since the data was divided by line I can just use split() to separate each word in the data and then try to init a variable with each value.

when all the variable are initialized we write another SQL request to insert all the value in the table

S06_03

S06_03_a

```
sql_query = "select * from GLARE where merger_rate > 2;"
cursor.execute(sql_query)
result = cursor.fetchall()
print(result[:5])
conn.commit()
```

S06_03_b

```
(2, 9999999999999999, 'UGC12889', '00000168+4716282', 'null', 'G', '0.007012', '47.274521', 72.6011916674, 1e+16, 0.01673, 1e+16, 999999999
(4, 9999999999999999, 'PGC000004', 'null', 'null', 'G', '0.0144', '23.08764', 66.6813492116, 1e+16, 0.0146025673013, 0.39, 9999999999999999
(6, 9999999999999999, 'PGC000006', '00000214+1552539', 'null', 'G', '0.008953', '15.88166', 87.1007149719, 1e+16, 0.02002, 1e+16, 999999999
(7, 9999999999999999, 'PGC000007', '00000442-0004599', 'null', 'G', '0.018438', '-0.083323', 103.715941639, 1e+16, 0.02377, 1e+16, 99999999
(10, 9999999999999999, 'PGC000010', '00000784-0002264', 'null', 'G', '0.032671', '-0.040668', 103.227223745, 1e+16, 0.02366, 1e+16, 9999999
```

S06_03_c

we select all the line frome the rable where the merger rate is greater than to so the result is a bit long that’s why I only printed the 5 first result