CONNECTING TO THE BAG DATASET IN THE SNOWFLAKE MARKETPLACE

```
        ### Example 1: Query the entire dataset
df_data = session.table('PUBLIC."Verblijfsobject_Bag"')

### Create pandas dataframe from the data
pandas_df_data = df_data.to_pandas()
pandas_df_data.head()

### Example 2: Create custom query for specific number of buildings around the
Tensing office in Waardenburg
df_geo = session.sql('''
    select
        a."postcode" as "postalCode",
        a."huisnummer" as "houseNumber",
        a."huisletter" as "houseLetter",
        a."huisnummertoevoeging" as "houseNumber_add",
        b."naam" as "streetName",
        c."naam" as "townShip" ,
        d."oppervlakte" as "surfaceArea",
        d."tijdstipRegistratie" as "registrationDate",
        st_x(d."geography") as "longitude",
        st_y(d."geography") as "latitude",
        round(st_distance(d."geography", st_makepoint(5.24822729660412,
51.83271006966607)),0) as "distanceFromOffice"
    from public."Nummeraanduiding_Bag" as a
    join public."OpenbareRuimte_Bag" as b
    on a."ligtAan" = b."identificatie"
    left join public."Woonplaats" c
    on coalesce(a."ligtIn", b."ligtIn") = c."identificatie"
    join public."Verblijfsobject_Bag" d
    on a."identificatie" = d."heeftAlsHoofdadres"
    join public."VBO_Gebruiksdoel_Bag" e
    on d."identificatie" = e."identificatie"
    and d."voorkomenidentificatie" = e."voorkomenidentificatie"
    where a."eindGeldigheid" is null
        and b."eindGeldigheid" is null
        and c."eindGeldigheid" is null
        and d."eindGeldigheid" is null
    and e."gebruiksdoel" = 'woonfunctie'
    and c."naam" = 'Waardenburg'
    order by st_distance(d."geography", st_makepoint(5.24822729660412,
51.83271006966607)),  a."postcode", a."huisnummer", a."huisletter"
    limit 5000
''')

### Create pandas dataframe of the queried data
pandas_df_geo = df_geo.to_pandas()
pandas_df_geo
```