# BLM442 Keşif aktivitesi-2: SQLite üzerinde veri analizi

**İsim/Soyisim: Sinan ÇALIŞIR**

**Öğrenci Numarası: 170201098**

**E-mail: sinancalisir001@gmail.com**

**Kullanılan dataset adı: House Prices**

**Dataset url:** https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data

Verisetinde 82 sütun olduğu için ekrana sığması amacıyla, yalnızca problemde istenilen sütunları SELECT statementi içine yazdım. Ayrıca bütün veriyle işlem yaptığım sorguları da yine ekrana sığması ve çıktı da bütün veriyi yazmaması için sorgulara LIMIT 20 ekledim.

**İntihal:**

Netten alınacak kısmi kod parçaları önceden kod içinde/raporda belirtilmek ve soru sorulduğunda cevaplanması durumunda sıkıntı çıkarmayacaktır. (i) Internet kaynağını belirtmeyen/açıklayamayan/üzerinde geliştirme yapmayan veya (ii) birbirleriyle benzer/aynı çalışma teslim edenlerin aktiviteleri sıfır üzerinden değerlendirilecektir.

**KA-2 Gönderim şekli**

1-pdf olarak "ogrenciNo.pdf" dokumanını gonderme -calisma dosyasında (File -> Print Preview) sonrası olusan html dosyasını yazdır deyip pdf olarak kaydetme veya -(File -> Download as -> HTML) seklinde indirip pdf'e cevirme http://html2pdf.com/

2-(File -> Download as -> Notebook) indirip "ogrenciNo.ipynb" dokumanını gonderme

suleyman.eken@kocaeli.edu.tr

Problemlerinizin SELECT, WHERE, ORDER BY, SELECT TOP, LIKE, IN, BETWEEN, AS, INTERSECT, GROUP BY, HAVING, COUNT(), MIN() and MAX(), AVG(), SUM(), INNER JOIN, RIGHT (OUTER) JOIN, LEFT (OUTER) JOIN vs. keyword'lerinden birkaçını içermesini sağlayınız.

## Bu notebook Python3 kerneli ile çalıştırıldığı için öncelikle csv dosyasından veritabanı oluşturuyoruz.

Asagida ekledigim kütüphaneleri yalnızca veritabanını olusturmak için ekledim.

In [1]:

```
%load_ext sql
```

In [2]:

```
import sqlite3 # Veritabanını oluşturmak için
import pandas as pd # csv dosyasını okumak ve veritabanına yazmak için

# Veritabanı yoksa olustur, varsa baglan.
db_name = "odev2.db3"
con = sqlite3.connect(db_name)

# Csv dosyasını oku
data = pd.read_csv("./competitions/house-prices-advanced-regression-techniques/train.csv")

# house_prices tablosuna csv dosyasının içeriğini yaz.
table_name = "house_prices"
data.to_sql(table_name, con, index=False)
con.close()
```

In [3]:

```
%sql sqlite:///odev2.db3
```

Out[3]:

```
'Connected: @odev2.db3'
```

**Problem 1: Verinin ilk 10 satırını listeleyin.**

In [4]:

```
%sql SELECT * FROM house_prices LIMIT 10
```

```
 * sqlite:///odev2.db3
Done.
```

Out[4]:

| Id | MSSubClass | MSZoning | LotFrontage | LotArea | Street | Alley | LotShape | LandContour | Utilities | LotConfig | LandSlope | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 60 | RL | 65.0 | 8450 | Pave | None | Reg | Lvl | AllPub | Inside | Gtl | C |
| 2 | 20 | RL | 80.0 | 9600 | Pave | None | Reg | Lvl | AllPub | FR2 | Gtl | V |
| 3 | 60 | RL | 68.0 | 11250 | Pave | None | IR1 | Lvl | AllPub | Inside | Gtl | C |
| 4 | 70 | RL | 60.0 | 9550 | Pave | None | IR1 | Lvl | AllPub | Corner | Gtl | C |
| 5 | 60 | RL | 84.0 | 14260 | Pave | None | IR1 | Lvl | AllPub | FR2 | Gtl | N |
| 6 | 50 | RL | 85.0 | 14115 | Pave | None | IR1 | Lvl | AllPub | Inside | Gtl | N |
| 7 | 20 | RL | 75.0 | 10084 | Pave | None | Reg | Lvl | AllPub | Inside | Gtl | S |
| 8 | 60 | RL | None | 10382 | Pave | None | IR1 | Lvl | AllPub | Corner | Gtl | N |
| 9 | 50 | RM | 51.0 | 6120 | Pave | None | Reg | Lvl | AllPub | Inside | Gtl | C |
| 10 | 190 | RL | 50.0 | 7420 | Pave | None | Reg | Lvl | AllPub | Corner | Gtl | E |

**Problem 2: 2000 yılından sonra yapılmış evleri fiyatlarına göre azalan sırada sıralayın.**

In [5]:

```
%sql SELECT Id, YearBuilt, SalePrice  FROM house_prices WHERE YearBuilt >= 2000 ORDER BY SalePrice
DESC LIMIT 20
```

```
 * sqlite:///odev2.db3
Done.
```

Out[5]:

| Id | YearBuilt | SalePrice |
|---|---|---|
| 899 | 2009 | 611657 |
| 804 | 2008 | 582933 |
| 1047 | 2005 | 556581 |
| 441 | 2008 | 555000 |
| 770 | 2003 | 538000 |
| 179 | 2008 | 501837 |
| 799 | 2008 | 485000 |
| 1374 | 2001 | 466500 |
| 1244 | 2006 | 465000 |
| 592 | 2008 | 451950 |
| 528 | 2008 | 446261 |

| | | |
|---|---|---|
| 528 | 2008 | 446261 |
| 474 | 2006 | 440000 |
| 59 | 2006 | 438780 |
| 350 | 2005 | 437154 |
| 390 | 2007 | 426000 |
| 1143 | 2006 | 424870 |
| 665 | 2005 | 423000 |
| 279 | 2006 | 415298 |
| 162 | 2003 | 412500 |
| 516 | 2009 | 402861 |

**Problem 3: Fiyatı 500k $ dan yüksek toplam ev sayısını bulun.**

In [6]:

```
%sql SELECT COUNT(Id) AS Ttl_HsNum_Abv_500k FROM house_prices WHERE SalePrice >= 500000
```

```
 * sqlite:///odev2.db3
Done.
```

Out[6]:

| Ttl_HsNum_Abv_500k |
|---|
| 9 |

**Problem 4: Fiyatı 100k-120k arasındaki evlerin ortalama kalitesini(OverallQual) bulun.**

In [7]:

```
%sql SELECT Id, OverallQual, SalePrice FROM house_prices WHERE SalePrice BETWEEN 100000 AND 120000
LIMIT 20
```

```
 * sqlite:///odev2.db3
Done.
```

Out[7]:

| Id | OverallQual | SalePrice |
|---|---|---|
| 10 | 5 | 118000 |
| 39 | 5 | 109000 |
| 49 | 4 | 113000 |
| 52 | 6 | 114500 |
| 53 | 5 | 110000 |
| 62 | 5 | 101000 |
| 75 | 3 | 107400 |
| 80 | 5 | 110000 |
| 91 | 4 | 109900 |
| 103 | 5 | 118964 |
| 107 | 4 | 100000 |
| 108 | 5 | 115000 |
| 109 | 5 | 115000 |
| 122 | 4 | 100000 |
| 141 | 4 | 115000 |

| | | |
|---|---|---|
| 147 | 5 | 105000 |
| 150 | 5 | 115000 |
| 157 | 5 | 109500 |
| 164 | 4 | 103200 |
| 180 | 5 | 100000 |

**Problem 5: Ortalama kalitesi 8 ve üzeri olan evlerin ortalama fiyatını bulun.**

In [8]:

```
%sql SELECT AVG(SalePrice) AS Avg_SalePrice FROM house_prices WHERE OverallQual >= 8
```

```
 * sqlite:///odev2.db3
Done.
```

Out[8]:

| Avg_SalePrice |
|---|
| 305035.89956331876 |

http://blogs.lessthandot.com/index.php/datamgmt/datadesign/title-1/

**Problem 6: Garaj tipi(GarageType) sütununun yüzde kaçı NULL değerlerden oluşuyor?**

In [9]:

```
%sql SELECT 100.0 * SUM( \
    CASE WHEN GarageType IS NULL THEN 1 ELSE 0 END) / \
    COUNT(*) AS Grg_Type_NULL_Percent \
    FROM house_prices
```

```
 * sqlite:///odev2.db3
Done.
```

Out[9]:

| Grg_Type_NULL_Percent |
|---|
| 5.5479452054794525 |

**Problem 7: Garaj tipindeki(GarageType) değerlerin toplam sayısını ve yüzdesini bulun.**

In [10]:

```
%sql SELECT GarageType AS Value, COUNT(GarageType) AS ValueCount, 100.0 * COUNT(*) / \
    (SELECT COUNT(*) FROM house_prices ) AS Percentage \
    FROM house_prices GROUP BY GarageType ORDER BY Percentage DESC;
```

```
 * sqlite:///odev2.db3
Done.
```

Out[10]:

| Value | ValueCount | Percentage |
|---|---|---|
| Attchd | 870 | 59.58904109589041 |
| Detchd | 387 | 26.506849315068493 |
| BuiltIn | 88 | 6.027397260273973 |
| None | 0 | 5.5479452054794525 |

| | | |
|---|---|---|
| Basment | 19 | 1.3013698630136987 |
| CarPort | 9 | 0.6164383561643836 |
| 2Types | 6 | 0.410958904109589 |

**Problem 8: Ortalama kalitesi en yüksek olan evlerden fiyatı en düşük olan evi bulun.**

In [11]:

```
%sql SELECT Id, OverallQual, MIN(SalePrice) FROM house_prices \
    WHERE OverallQual == (SELECT MAX(OverallQual) AS max_overall_qual FROM house_prices)
```

```
 * sqlite:///odev2.db3
Done.
```

Out[11]:

| Id | OverallQual | MIN(SalePrice) |
|---|---|---|
| 1299 | 10 | 160000 |

**Problem 9: Satış fiyatının(SalePrice) en küçük, en büyük değerlerini ve ilk ceyrekliğin ortalamasını bulun.**

In [12]:

```
%sql SELECT MIN(SalePrice), \
ROUND((SELECT AVG(SalePrice) FROM \
(SELECT * FROM house_prices ORDER BY SalePrice \
LIMIT (SELECT COUNT(*) * 0.25 FROM house_prices))), 3) AS Quarter_1st, \
MAX(SalePrice) FROM house_prices
```

```
 * sqlite:///odev2.db3
Done.
```

Out[12]:

| MIN(SalePrice) | Quarter_1st | MAX(SalePrice) |
|---|---|---|
| 34900 | 105831.595 | 755000 |

**Problem 10: 2000 yılından sonra yapılmış evlerin toplam sayısını ve ortalama satış fiyatını bulun.**

In [13]:

```
%sql SELECT YearBuilt, COUNT(*) AS Count, ROUND(AVG(SalePrice), 2) AS SalePrice \
FROM house_prices GROUP BY YearBuilt HAVING YearBuilt >= 2000
```

```
 * sqlite:///odev2.db3
Done.
```

Out[13]:

| YearBuilt | Count | SalePrice |
|---|---|---|
| 2000 | 24 | 210766.67 |
| 2001 | 20 | 242630.0 |
| 2002 | 23 | 226869.57 |
| 2003 | 45 | 227408.58 |
| 2004 | 54 | 210347.72 |
| 2005 | 64 | 229680.95 |
| 2006 | 67 | 251775.45 |
| 2007 | 49 | 255362.73 |

| | | |
|---|---|---|
| 2008 | 23 | 348849.13 |
| 2009 | 18 | 269220.0 |
| 2010 | 1 | 394432.0 |

**Problem 11: Oturmaya elverişli(Residential-MSZoning) bölgelerde bulunan evleri satış yıllarına göre sıralayın.**

In [14]:

```
%sql SELECT Id, MsZoning, YrSold, SalePrice FROM house_prices WHERE MSZoning LIKE "R%" ORDER BY Yr
Sold LIMIT 20
```

```
 * sqlite:///odev2.db3
Done.
```

Out[14]:

| Id | MSZoning | YrSold | SalePrice |
|---|---|---|---|
| 4 | RL | 2006 | 140000 |
| 12 | RL | 2006 | 345000 |
| 18 | RL | 2006 | 90000 |
| 21 | RL | 2006 | 325300 |
| 29 | RL | 2006 | 207500 |
| 36 | RL | 2006 | 309000 |
| 41 | RL | 2006 | 160000 |
| 45 | RL | 2006 | 141000 |
| 52 | RM | 2006 | 114500 |
| 54 | RL | 2006 | 385000 |
| 58 | RL | 2006 | 196500 |
| 59 | RL | 2006 | 438780 |
| 61 | RL | 2006 | 158000 |
| 70 | RL | 2006 | 225000 |
| 82 | RM | 2006 | 153500 |
| 86 | RL | 2006 | 260000 |
| 91 | RL | 2006 | 109900 |
| 92 | RL | 2006 | 98600 |
| 97 | RL | 2006 | 214000 |
| 111 | RL | 2006 | 136900 |

**Problem 12: 1stFlrSF(İlk katın büyüklüğü(SF)), 2ndFlrSF (İkinci katın büyüklüğü(SF)) ve LowQualFinSF (Düşük kaliteli ) sütunlarının toplamının GrLivArea(Evin büyüklüğü) sütununa eşit olduğunu gösterin(Veriden yapılan çıkarım sonucu bu bilgiyi elde ettim).**

In [15]:

```
%sql SELECT * FROM (SELECT GrLivArea - Total_Area AS diff FROM \
(SELECT GRLivArea, "1stFlrSF" + "2ndFlrSF" + LowQualFinSF AS Total_Area \
FROM house_prices)) WHERE diff != 0
```

```
 * sqlite:///odev2.db3
Done.
```

Out[15]:

| diff |
|---|

**Problem 13: Kış mevsiminde satılan evlerin aylara göre satış fiyatlarını gösterin.**

In [16]:

```
%sql SELECT MoSold, ROUND(AVG(SalePrice), 3) AS AVG_SalePrice \
FROM house_prices WHERE MoSold IN (12, 1, 2) GROUP BY MoSold
```

```
 * sqlite:///odev2.db3
Done.
```

Out[16]:

| MoSold | AVG_SalePrice |
|---|---|
| 1 | 183256.259 |
| 2 | 177882.0 |
| 12 | 186518.966 |

**Problem 14: Satış fiyatının ortalama(mean) ve ortanca(median) değerlerini bulun.**

In [17]:

```
%sql SELECT DISTINCT (SELECT SalePrice FROM house_prices ORDER BY SalePrice \
LIMIT 1 OFFSET (SELECT COUNT(*) / 2 FROM house_prices)) AS Median, \
ROUND((SELECT AVG(SalePrice) FROM house_prices), 3) AS Mean \
FROM house_prices
```

```
 * sqlite:///odev2.db3
Done.
```

Out[17]:

| Median | Mean |
|---|---|
| 163000 | 180921.196 |

**Problem 15: Bodrum kalitesinin(BsmtQual) yüksekliğe çevrilmiş halini gösterin.**

In [18]:

```
%sql SELECT Id, BsmtQual, \
CASE WHEN BsmtQual=="Ex" THEN '100+' \
WHEN BsmtQual=="Gd" THEN '90-99' \
WHEN BsmtQual=="TA" THEN '80-89' \
WHEN BsmtQual=="Fa" THEN '70-79' \
WHEN BsmtQual=="Po" THEN '<70' \
END AS "Height(inches)" FROM house_prices LIMIT 20
```

```
 * sqlite:///odev2.db3
Done.
```

Out[18]:

| Id | BsmtQual | Height(inches) |
|---|---|---|
| 1 | Gd | 90-99 |
| 2 | Gd | 90-99 |
| 3 | Gd | 90-99 |
| 4 | TA | 80-89 |
| 5 | Gd | 90-99 |
| 6 | Gd | 90-99 |

| | | |
|---|---|---|
| 7 | Ex | 100+ |
| 8 | Gd | 90-99 |
| 9 | TA | 80-89 |
| 10 | TA | 80-89 |
| 11 | TA | 80-89 |
| 12 | Ex | 100+ |
| 13 | TA | 80-89 |
| 14 | Gd | 90-99 |
| 15 | TA | 80-89 |
| 16 | TA | 80-89 |
| 17 | TA | 80-89 |
| 18 | None | None |
| 19 | TA | 80-89 |
| 20 | TA | 80-89 |