```
import pandas as pd
import sqlite3


heart_df = pd.read_csv('heart.csv')
medical_df = pd.read_csv('medical_clean.csv')


heart_df.head()
medical_df.head()
```

	CaseOrder	Customer_id	Interaction	UID	Cit
0	1	C412403	8cd49b13-f45a-4b47-a2bd-173ffa932c2f	3a83ddb66e2ae73798bdf1d705dc0932	Ev
1	2	Z919181	d2450b70-0337-4406-bdbb-bc1037f1734c	176354c5eef714957d486009feabf195	Mariann
2	3	F995323	a2057123-abf5-4a2c-abad-8ffe33512562	e19a0fa00aeda885b8a436757e889bc9	Siou Fal
3	4	A879973	1dec528d-eb34-4079-adce-0d7a40e82205	cd17d7b6d152cb6f23957346d11c3f07	Ne Richlan
4	5	C544523	5885f56b-d6da-43a3-8760-83583af94266	d2f0425877b10ed6bb381f3e2579424a	We Poi

```
testDB = sqlite3.connect('test.db')


heartAge = heart_df['Age']


heartAge.head()
```

```
0    40
1    49
2    37
3    48
```

```
4    54
Name: Age, dtype: int64
```

```
heartSex = heart_df['Sex']
heartHeartDisease = heart_df['HeartDisease']
medicalAge = medical_df['Age']
medicalGender = medical_df['Gender']
medicalHighBlood = medical_df['HighBlood']
medicalIncome = medical_df['Income']
```

```
!pip install ipython-sql
```

```
Requirement already satisfied: ipython-sql in /usr/local/lib/python3.7/dist-packages (0
Requirement already satisfied: sqlparse in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: ipython-genutils>=0.1.0 in /usr/local/lib/python3.7/dist
Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages (from ipyt
Requirement already satisfied: prettytable in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: ipython>=1.0 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: sqlalchemy>=0.6.7 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: pexpect in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: traitlets>=4.2 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: simplegeneric>0.8 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: decorator in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: prompt-toolkit<2.0.0,>=1.0.4 in /usr/local/lib/python3.7
Requirement already satisfied: setuptools>=18.5 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: pygments in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: pickleshare in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: wcwidth in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: importlib-metadata in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: greenlet!=0.4.17 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: typing-extensions>=3.6.4 in /usr/local/lib/python3.7/dis
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: ptyprocess>=0.5 in /usr/local/lib/python3.7/dist-package
```

```
%load_ext sql
```

```
%sql sqlite:///test.db
```

```
'Connected: @test.db'
```

```
heartAge.to_sql('heartAge', testDB)
heartSex.to_sql('heartSex', testDB)
heartHeartDisease.to_sql('heartHeartDisease', testDB)
medicalAge.to_sql('medicalAge', testDB)
medicalGender.to_sql('medicalGender', testDB)
medicalHighBlood.to_sql('medicalHighBlood', testDB)
medicalIncome.to_sql('medicalIncome', testDB)
```

```
%%sql

SELECT * FROM heartAge
LIMIT 10;
```

```
 * sqlite:///test.db
Done.
```

index	Age
0	40
1	49
2	37
3	48
4	54
5	39
6	45
7	54
8	37
9	48

```
%%sql

SELECT COUNT(*) - COUNT(Age) AS missing
FROM heartAge;
```

```
 * sqlite:///test.db
Done.
```

missing
0

```
%%sql

SELECT COUNT(*) - COUNT(Sex) AS missing
FROM heartSex;
```

```
 * sqlite:///test.db
Done.
```

missing
0

```
%%sql

SELECT COUNT(*) - COUNT(HeartDisease) AS missing
FROM heartHeartDisease;
```

```
 * sqlite:///test.db
Done.
```

```
%%sql

SELECT * FROM heartHeartDisease
LIMIT 10;
```

```
 * sqlite:///test.db
Done.
```

index	HeartDisease
0	0
1	1
2	0
3	1
4	0
5	0
6	0
7	0
8	1
9	0

```
%%sql

SELECT * FROM heartAge
LIMIT 10;
```

```
 * sqlite:///test.db
Done.
```

index	Age
0	40
1	49
2	37
3	48
4	54
5	39
6	45
7	54
8	37
9	48

```
%%sql

SELECT * FROM heartSex
LIMIT 10;
```

```
 * sqlite:///test.db
Done.
```

index	Sex
0	M
1	F
2	M
3	F
4	M
5	M
6	F
7	M
8	M

```
%%sql

SELECT * FROM heartHeartDisease
LIMIT 10;
```

```
 * sqlite:///test.db
Done.
```

index	HeartDisease
0	0
1	1
2	0
3	1
4	0
5	0
6	0
7	0
8	1
9	0

```
%%sql

SELECT * FROM medicalAge
LIMIT 10
```

```
 * sqlite:///test.db
Done.
```

index	Age

```
%%sql

SELECT COUNT(*) - COUNT(Age) AS missing
FROM heartAge;
SELECT COUNT(*) - COUNT(Sex) AS missing
FROM heartSex;
SELECT COUNT(*) - COUNT(HeartDisease) AS missing
FROM heartHeartDisease;
```

```
 * sqlite:///test.db
Done.
Done.
Done.
```

missing
0

```
%%sql

SELECT COUNT(*) - COUNT(Age) AS missing
FROM medicalAge;
SELECT COUNT(*) - COUNT(Gender) AS missing
FROM medicalGender;
SELECT COUNT(*) - COUNT(HighBlood) AS missing
FROM medicalHighBlood;
SELECT COUNT(*) - COUNT(Income) AS missing
FROM medicalIncome;
```

```
 * sqlite:///test.db
Done.
Done.
Done.
Done.
```

missing
0

```
%%sql

SELECT heartAge.Age, medicalAge.Age FROM heartAge
INNER JOIN medicalAge
ON heartAge.Age = medicalAge.Age
```

```
%%sql

UPDATE heartSex
SET Sex='Male' WHERE Sex='M';
```

```
 * sqlite:///test.db
0 rows affected.
[]
```

```
%%sql

UPDATE heartSex
SET Sex = 'Female' WHERE Sex = 'F'
```

```
 * sqlite:///test.db
0 rows affected.
[]
```

```
%%sql

SELECT * FROM heartSex
LIMIT 10;
```

```
 * sqlite:///test.db
Done.
```

index	Sex
0	Male
1	Female
2	Male
3	Female
4	Male
5	Male
6	Female
7	Male
8	Male
9	Female

```
%%sql

SELECT * FROM medicalGender
LIMIT 10;
```

```
 * sqlite:///test.db
Done.
```

index	Gender
0	Male

```
%%sql

SELECT COUNT(Age) FROM heartAge
WHERE Age > 50;
```

```
 * sqlite:///test.db
Done.
```

COUNT(Age)
602

```
%%sql

SELECT
  CASE WHEN Income > 75000 THEN '>75k'
  WHEN Income > 50000 THEN '>50k'
  ELSE '<50k'
  END AS incomeBracket
FROM medicalIncome
GROUP BY Income
ORDER BY Income DESC;
```

```
%%sql
/*
SELECT HighBlood, HeartDisease,
  CASE WHEN HighBlood = 'Yes' AND HeartDisease = '1' THEN 'High Risk'
  WHEN HighBlood = 'No' AND HeartDisease = '1' THEN 'Moderate Risk'
  WHEN HighBlood = 'Yes' AND HeartDisease = '0' THEN 'Moderate Risk'
  WHEN HighBlood = 'No' AND HeartDisease = '0' THEN 'Low Risk'
  END AS risk
FROM medicalHighBlood, heartHeartDisease,
GROUP BY risk
ORDER BY risk DESC;
*/
```

```
%%sql

SELECT HighBlood, HeartDisease
FROM medicalHighBlood
INNER JOIN heartHeartDisease
ON medicalHighBlood.HighBlood = heartHeartDisease.HeartDisease
WHERE HighBlood = 'Yes' AND HeartDisease = '1';
```

```
 * sqlite:///test.db
Done.
```

```
%%sql

SELECT medicalHighBlood.HighBlood,
  CASE medicalHighBlood.HighBlood
    WHEN 'Yes' THEN 'High Risk'
    WHEN 'No' THEN 'Low Risk'
  END AS risk,
  COUNT(*) AS c FROM medicalHighBlood GROUP BY risk
```

```
 * sqlite:///test.db
Done.
```

HighBlood	risk	c
Yes	High Risk	4090
No	Low Risk	5910

```
%%sql

SELECT heartHeartDisease.HeartDisease,
  CASE heartHeartDisease.HeartDisease
    WHEN '1' THEN 'High Risk'
    WHEN '0' THEN 'Low Risk'
  END AS risk,
  COUNT(*) AS c FROM heartHeartDisease GROUP BY risk
```

```
 * sqlite:///test.db
Done.
```

HeartDisease	risk	c
1	High Risk	508
0	Low Risk	410

```
%%sql

ALTER TABLE medicalHighBlood
ADD risk varchar(255);
```

```
%%sql

ALTER TABLE heartHeartDisease
ADD risk varchar(255);
```

```
%%sql

UPDATE medicalHighBlood
SET risk = 'High Risk' WHERE HighBlood = 'Yes';
```

```
 * sqlite:///test.db
```

```
      4090 rows affected.
      []
```

```
%%sql

UPDATE medicalHighBlood
SET risk = 'Low Risk' WHERE HighBlood = 'No';
```

```
       * sqlite:///test.db
      5910 rows affected.
      []
```

```
%%sql

UPDATE heartHeartDisease
SET risk = 'Low Risk' WHERE HeartDisease = '0'
```

```
       * sqlite:///test.db
      410 rows affected.
      []
```

```
%%sql

UPDATE heartHeartDisease
SET risk = 'High Risk' WHERE HeartDisease = '1'
```

```
       * sqlite:///test.db
      508 rows affected.
      []
```

```
%%sql

SELECT medicalHighBlood.risk, heartHeartDisease.risk
FROM heartHeartDisease
LEFT JOIN heartHeartDisease ON medicalHighBlood.risk = heartHeartDisease.risk;
```

```
       * sqlite:///test.db
      (sqlite3.OperationalError) no such column: medicalHighBlood.risk
      [SQL: SELECT medicalHighBlood.risk, heartHeartDisease.risk
      FROM heartHeartDisease
      LEFT JOIN heartHeartDisease ON medicalHighBlood.risk = heartHeartDisease.risk;]
      (Background on this error at: https://sqlalche.me/e/14/e3q8)
```

```
%%sql

SELECT * FROM medicalHighBlood
LIMIT 10;
```

```
 * sqlite:///test.db
Done.
```

index	HighBlood	risk
0	Yes	High Risk
1	Yes	High Risk
2	Yes	High Risk
3	No	Low Risk
4	No	Low Risk
5	No	Low Risk
6	Yes	High Risk
7	No	Low Risk
8	No	Low Risk

```
%%sql

SELECT risk FROM medicalHighBlood
UNION ALL SELECT risk FROM heartHeartDisease


%%sql

SELECT risk, COUNT(*) AS c FROM heartHeartDisease GROUP BY risk
```

```
 * sqlite:///test.db
Done.
```

risk	c
High Risk	508
Low Risk	410

```
%%sql

SELECT risk, COUNT(*) AS c FROM medicalHighBlood GROUP BY risk
```

```
 * sqlite:///test.db
Done.
```

risk	c
High Risk	4090
Low Risk	5910

✓ 0s completed at 8:19 PM