Name: Francisco, Luigi T.
Course and Section: CPE019-CPE32S3
Date Submitted: 07/02/2024
Instructor: Engr. Roman Richard

In [59]:
```
# Code cell 1
import pandas as pd

brainFile = './brainsize.txt'
brainFrame = pd.read_csv(brainFile, delim_whitespace=True)
```

In [60]:
```
brainFrame.head()
```

Out[60]:

	Gender	FSIQ	VIQ	PIQ	Weight	Height	MRI_Count
0	Female	133	132	124	118.0	64.5	816932
1	Male	140	150	124	NaN	72.5	1001121
2	Male	139	123	150	143.0	73.3	1038437
3	Male	133	129	128	172.0	68.8	965353
4	Female	137	132	134	147.0	65.0	951545

In [61]:
```
brainFrame.describe()
```

Out[61]:

	FSIQ	VIQ	PIQ	Weight	Height	MRI_Count
count	40.000000	40.000000	40.00000	38.000000	39.000000	4.000000e+01
mean	113.450000	112.350000	111.02500	151.052632	68.525641	9.087550e+05
std	24.082071	23.616107	22.47105	23.478509	3.994649	7.228205e+04
min	77.000000	71.000000	72.00000	106.000000	62.000000	7.906190e+05
25%	89.750000	90.000000	88.25000	135.250000	66.000000	8.559185e+05
50%	116.500000	113.000000	115.00000	146.500000	68.000000	9.053990e+05
75%	135.500000	129.750000	128.00000	172.000000	70.500000	9.500780e+05
max	144.000000	150.000000	150.00000	192.000000	77.000000	1.079549e+06

In [62]:
```
import numpy as np
import matplotlib.pyplot as plt
```

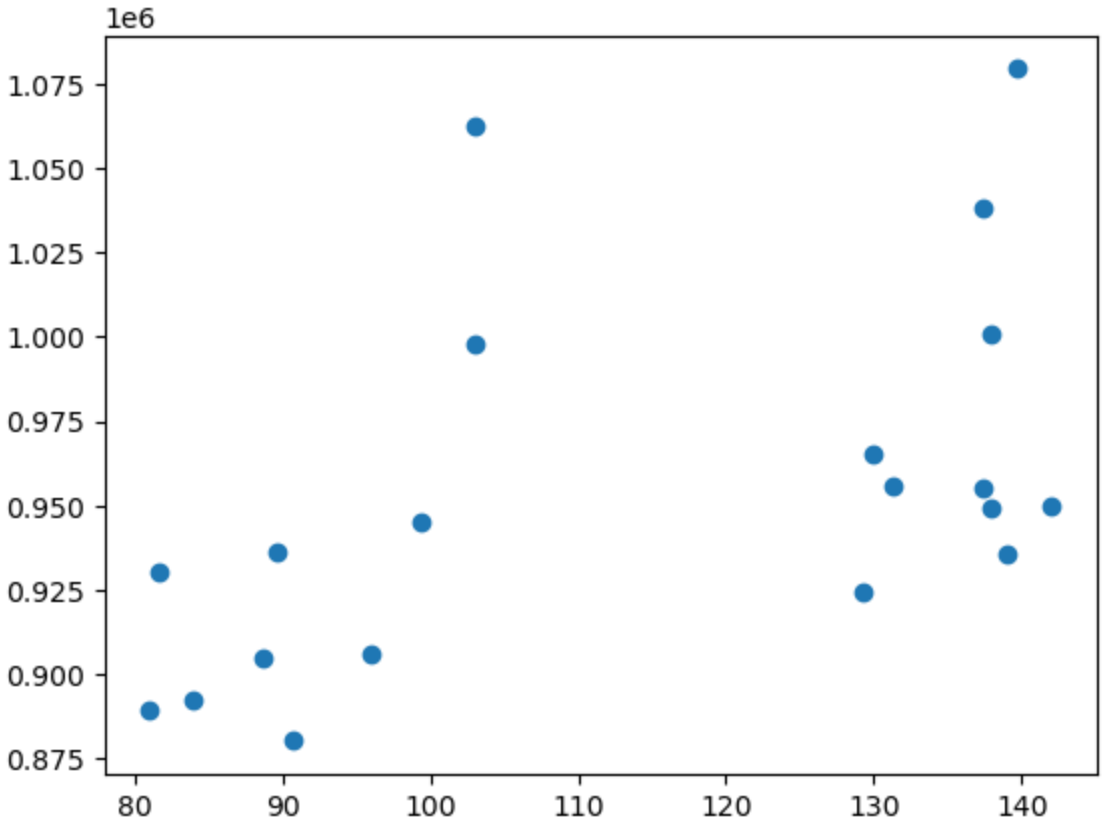
In [63]:
```
#Checker of columns

brainFrame.columns
```

Out[63]:
```
Index(['Gender', 'FSIQ', 'VIQ', 'PIQ', 'Weight', 'Height', 'MRI_Count'], dtype='objec
t')
```

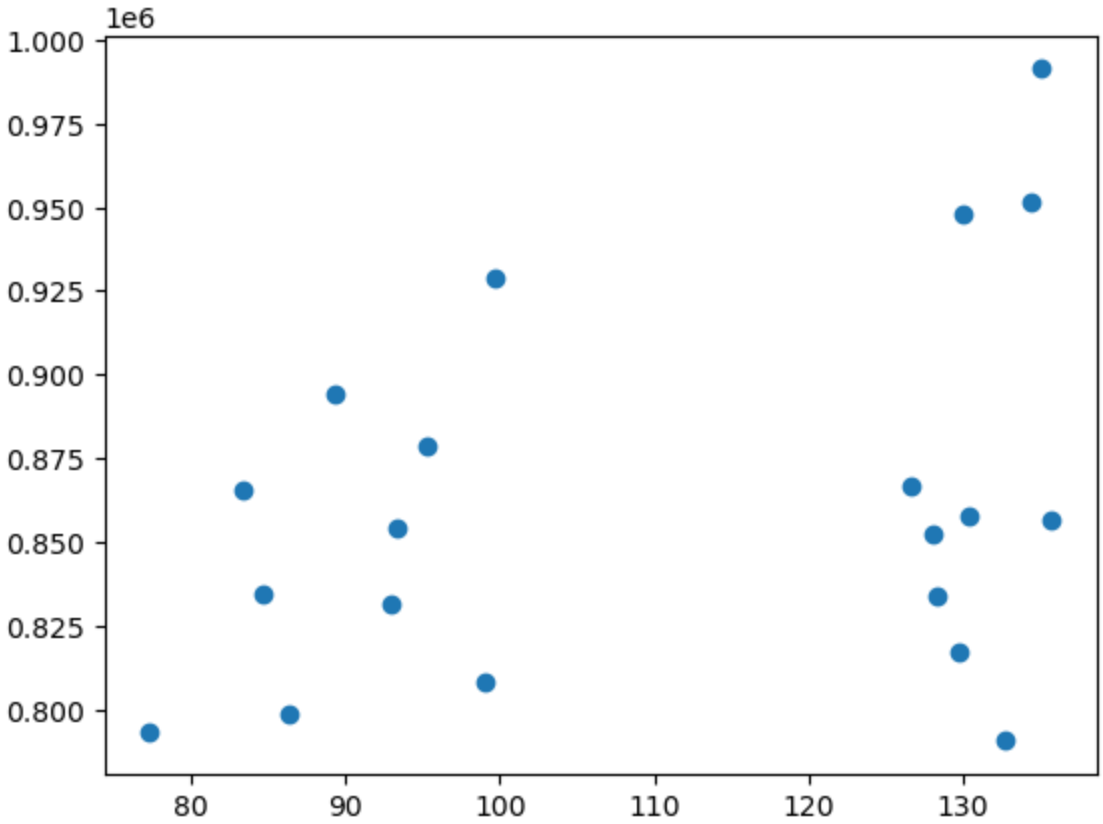
In [65]:
```
# Code cell 5
menDf = brainFrame[(brainFrame.Gender == 'Male')]
```

```
womenDf = brainFrame[(brainFrame.Gender == 'Female')]
```

In [67]:
```
#Plot the graphs! For Men

menMeanSmarts = menDf[["PIQ","FSIQ","VIQ"]].mean(axis=1)
plt.scatter(menMeanSmarts,menDf["MRI_Count"])
%matplotlib inline
```



In [73]:
```
#Plot the graphs! For Women

womenMeanSmarts = womenDf[["PIQ","FSIQ","VIQ"]].mean(axis=1)
plt.scatter(womenMeanSmarts,womenDf["MRI_Count"])
plt.show()

#%matplotlib intline
```



In [74]:
```
# Code cell 8
brainFrame.corr(method='pearson')
```

```
C:\Users\TIPQC\AppData\Local\Temp\ipykernel_8516\2353300390.py:2: FutureWarning: The
default value of numeric_only in DataFrame.corr is deprecated. In a future version, i
t will default to False. Select only valid columns or specify the value of numeric_on
ly to silence this warning.
  brainFrame.corr(method='pearson')
```

Out[74]:

	FSIQ	VIQ	PIQ	Weight	Height	MRI_Count
FSIQ	1.000000	0.946639	0.934125	-0.051483	-0.086002	0.357641
VIQ	0.946639	1.000000	0.778135	-0.076088	-0.071068	0.337478
PIQ	0.934125	0.778135	1.000000	0.002512	-0.076723	0.386817
Weight	-0.051483	-0.076088	0.002512	1.000000	0.699614	0.513378
Height	-0.086002	-0.071068	-0.076723	0.699614	1.000000	0.601712
MRI_Count	0.357641	0.337478	0.386817	0.513378	0.601712	1.000000

Notice at the left-to-right diagonal in the correlation table generated above. Why is the diagonal filled with 1s? Is that a coincidence? Explain.

The diagonal line is filled with one since it correlates the value to itself as such it would always be one. In essence the other values aside from those in diagonals are compared between two distinct variables so its not a coincidence its by design.

Still looking at the correlation table above, notice that the values are mirrored; values below the 1 diagonal have a mirrored counterpart above the 1 diagonal. Is that a coincidence? Explain.

The diagonal line in this case always have a distint variable to be correlated with and after the middle correlation it repeats that certain variable as such it seems to mirror afterwards so i guess its not a coincidence its by design and it would always happen when there are even objects.

In [75]:
```
womenDf.corr(method='pearson')
```

```
C:\Users\TIPQC\AppData\Local\Temp\ipykernel_8516\1249820013.py:1: FutureWarning: The
default value of numeric_only in DataFrame.corr is deprecated. In a future version, i
t will default to False. Select only valid columns or specify the value of numeric_on
ly to silence this warning.
  womenDf.corr(method='pearson')
```

Out[75]:

	FSIQ	VIQ	PIQ	Weight	Height	MRI_Count
FSIQ	1.000000	0.955717	0.939382	0.038192	-0.059011	0.325697
VIQ	0.955717	1.000000	0.802652	-0.021889	-0.146453	0.254933
PIQ	0.939382	0.802652	1.000000	0.113901	-0.001242	0.396157
Weight	0.038192	-0.021889	0.113901	1.000000	0.552357	0.446271
Height	-0.059011	-0.146453	-0.001242	0.552357	1.000000	0.174541
MRI_Count	0.325697	0.254933	0.396157	0.446271	0.174541	1.000000

In [76]:
```
menDf.corr(method='pearson')
```

```
C:\Users\TIPQC\AppData\Local\Temp\ipykernel_8516\2517740925.py:1: FutureWarning: The
default value of numeric_only in DataFrame.corr is deprecated. In a future version, i
t will default to False. Select only valid columns or specify the value of numeric_on
ly to silence this warning.
  menDf.corr(method='pearson')
```

Out[76]:

	FSIQ	VIQ	PIQ	Weight	Height	MRI_Count
FSIQ	1.000000	0.944400	0.930694	-0.278140	-0.356110	0.498369
VIQ	0.944400	1.000000	0.766021	-0.350453	-0.355588	0.413105
PIQ	0.930694	0.766021	1.000000	-0.156863	-0.287676	0.568237
Weight	-0.278140	-0.350453	-0.156863	1.000000	0.406542	-0.076875
Height	-0.356110	-0.355588	-0.287676	0.406542	1.000000	0.301543
MRI_Count	0.498369	0.413105	0.568237	-0.076875	0.301543	1.000000

In [77]:
```
!pip install seaborn
```

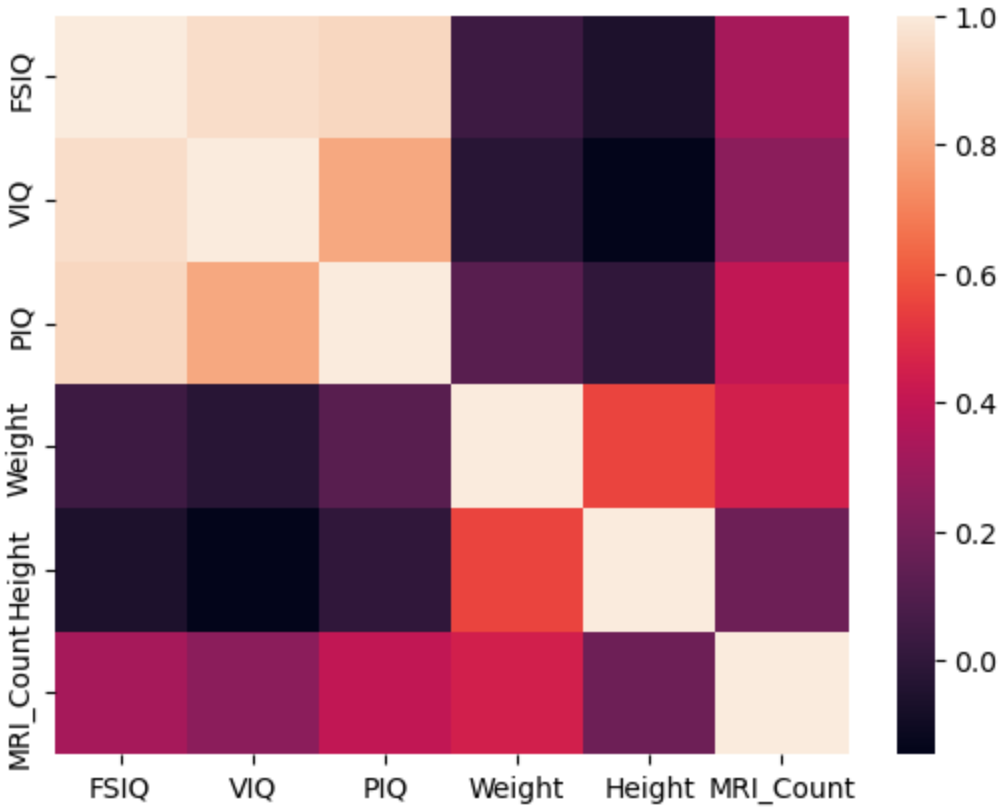
```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: seaborn in c:\programdata\anaconda3\lib\site-packages
(0.12.2)
Requirement already satisfied: numpy!=1.24.0,>=1.17 in c:\programdata\anaconda3\lib\s
ite-packages (from seaborn) (1.24.3)
Requirement already satisfied: pandas>=0.25 in c:\programdata\anaconda3\lib\site-pack
ages (from seaborn) (1.5.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in c:\programdata\anaconda3\li
b\site-packages (from seaborn) (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in c:\programdata\anaconda3\lib\site-
packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\programdata\anaconda3\lib\site-pack
ages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\programdata\anaconda3\lib\site
-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\programdata\anaconda3\lib\site
-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\programdata\anaconda3\lib\site-p
ackages (from matplotlib!=3.6.1,>=3.1->seaborn) (23.0)
Requirement already satisfied: pillow>=6.2.0 in c:\programdata\anaconda3\lib\site-pac
kages (from matplotlib!=3.6.1,>=3.1->seaborn) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\programdata\anaconda3\lib\site-
packages (from matplotlib!=3.6.1,>=3.1->seaborn) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\programdata\anaconda3\lib\s
ite-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\programdata\anaconda3\lib\site-pack
ages (from pandas>=0.25->seaborn) (2022.7)
Requirement already satisfied: six>=1.5 in c:\programdata\anaconda3\lib\site-packages
(from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.1->seaborn) (1.16.0)
```

```
DEPRECATION: Loading egg at c:\programdata\anaconda3\lib\site-packages\vboxapi-1.0-py
3.11.egg is deprecated. pip 23.3 will enforce this behaviour change. A possible repla
cement is to use pip for package installation..
```

In [80]:
```
import seaborn as sns
wcorr = womenDf.corr()
sns.heatmap(wcorr)
#plt.savefig('attribute_correlations.png', tight_layout=True)
```

```
C:\Users\TIPQC\AppData\Local\Temp\ipykernel_8516\3809214714.py:2: FutureWarning: The
default value of numeric_only in DataFrame.corr is deprecated. In a future version, i
t will default to False. Select only valid columns or specify the value of numeric_on
ly to silence this warning.
  wcorr = womenDf.corr()
```

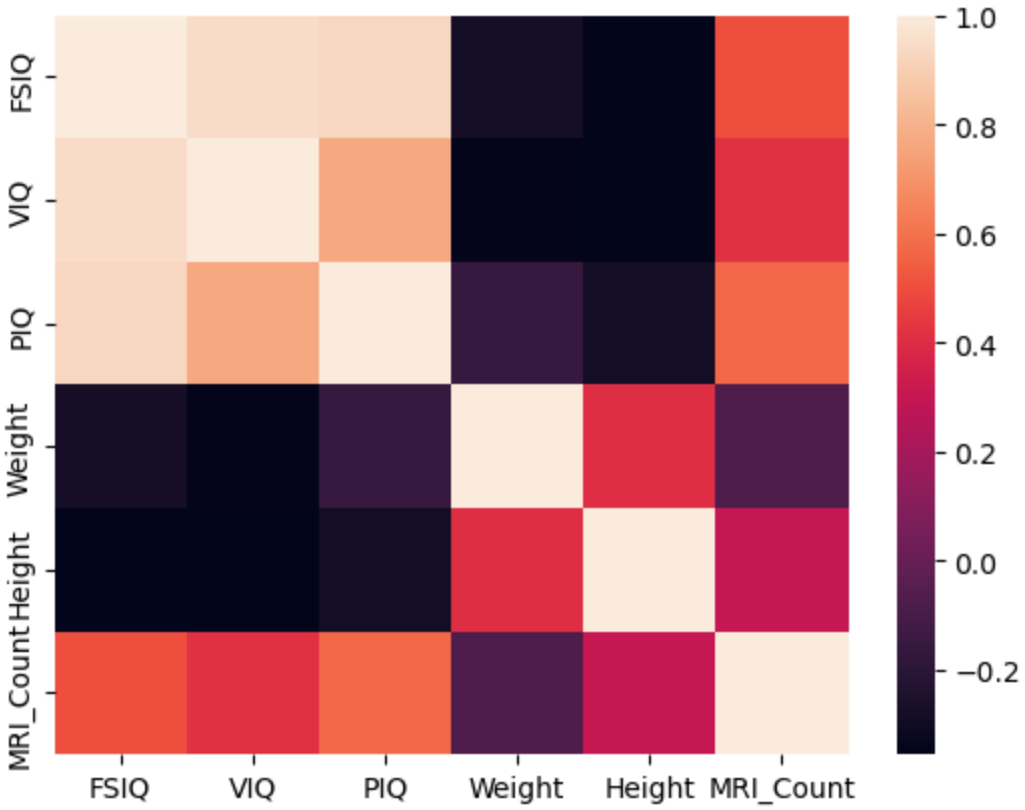
Out[80]:
```
<Axes: >
```



In [81]:
```
mcorr = menDf.corr()
sns.heatmap(mcorr)
#plt.savefig('attribute_correlations.png', tight_layout=True)
```

```
C:\Users\TIPQC\AppData\Local\Temp\ipykernel_8516\2815386289.py:1: FutureWarning: The
default value of numeric_only in DataFrame.corr is deprecated. In a future version, i
t will default to False. Select only valid columns or specify the value of numeric_on
ly to silence this warning.
  mcorr = menDf.corr()
```

Out[81]:
```
<Axes: >
```



Many variable pairs present correlation close to zero. What does that mean?

It means that there's not much correlation between the variables.

Why seperate the genders?

It would be useful to seperate genders due to the notable differences between a man and a woman.

What variables have stronger correlation with brain size (MRI_Count)? Is that expected? Explain.

In the heat map i've generated for the male that seems that PIQ,VIQ,FSIQ have a correlation coefficient greater than 0.4, although since this seems to be in scientific context this would not mean much where 0.9 is expected to establish a meaningful correlation between variables. The heatmap for females presents weight and PIQ to be the strong variables but essentially not that significant or meaningful as i have said earlier.

For expectations, I don't have much since I'm not familiar with brain imaging to have correlation with IQ since I don't even know what MRI count actually is except for its usage to find what the brain looks like in that sense i expected that there's not much correlation between those variables as theres a lot of factors affecting intelligence.

Supplementary Activity

In [82]:
```
dFile = './diabetes.csv'
diaFrame= pd.read_csv(dFile)
```

In [83]:
```
diaFrame.head()
```

Out[83]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age
0	6	148	72	35	0	33.6	0.627	50
1	1	85	66	29	0	26.6	0.351	31
2	8	183	64	0	0	23.3	0.672	32
3	1	89	66	23	94	28.1	0.167	21
4	0	137	40	35	168	43.1	2.288	33

In [85]:
```
diaFrame.columns
```

Out[85]:
```
Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',
       'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'],
      dtype='object')
```

In [86]:
```
diaFrame.describe()
```

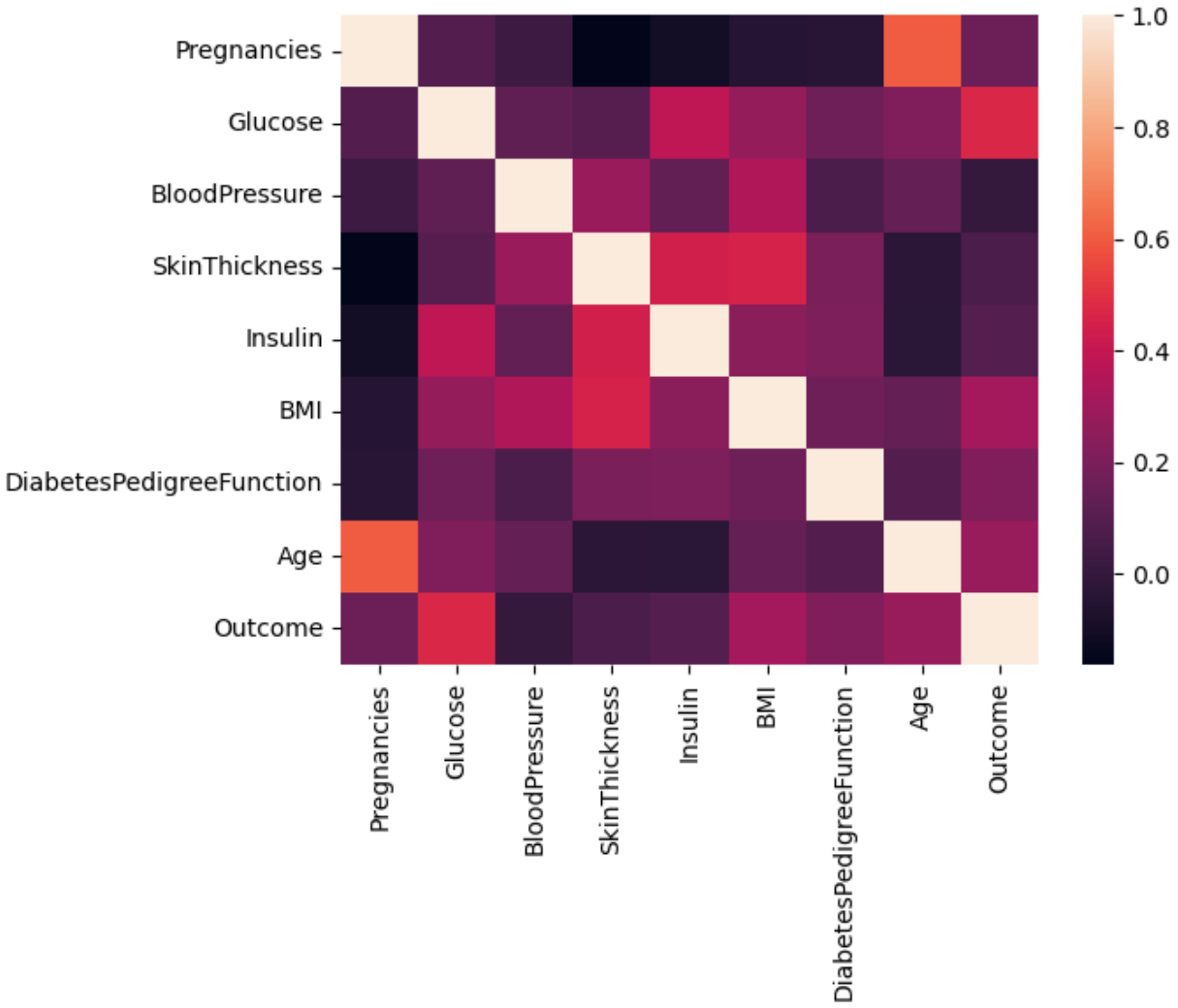
Out[86]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigr
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	

In [88]:
```
OldDf = diaFrame[(diaFrame.Age >= 40)]
YoungDf = diaFrame[(diaFrame.Age <40)]
```

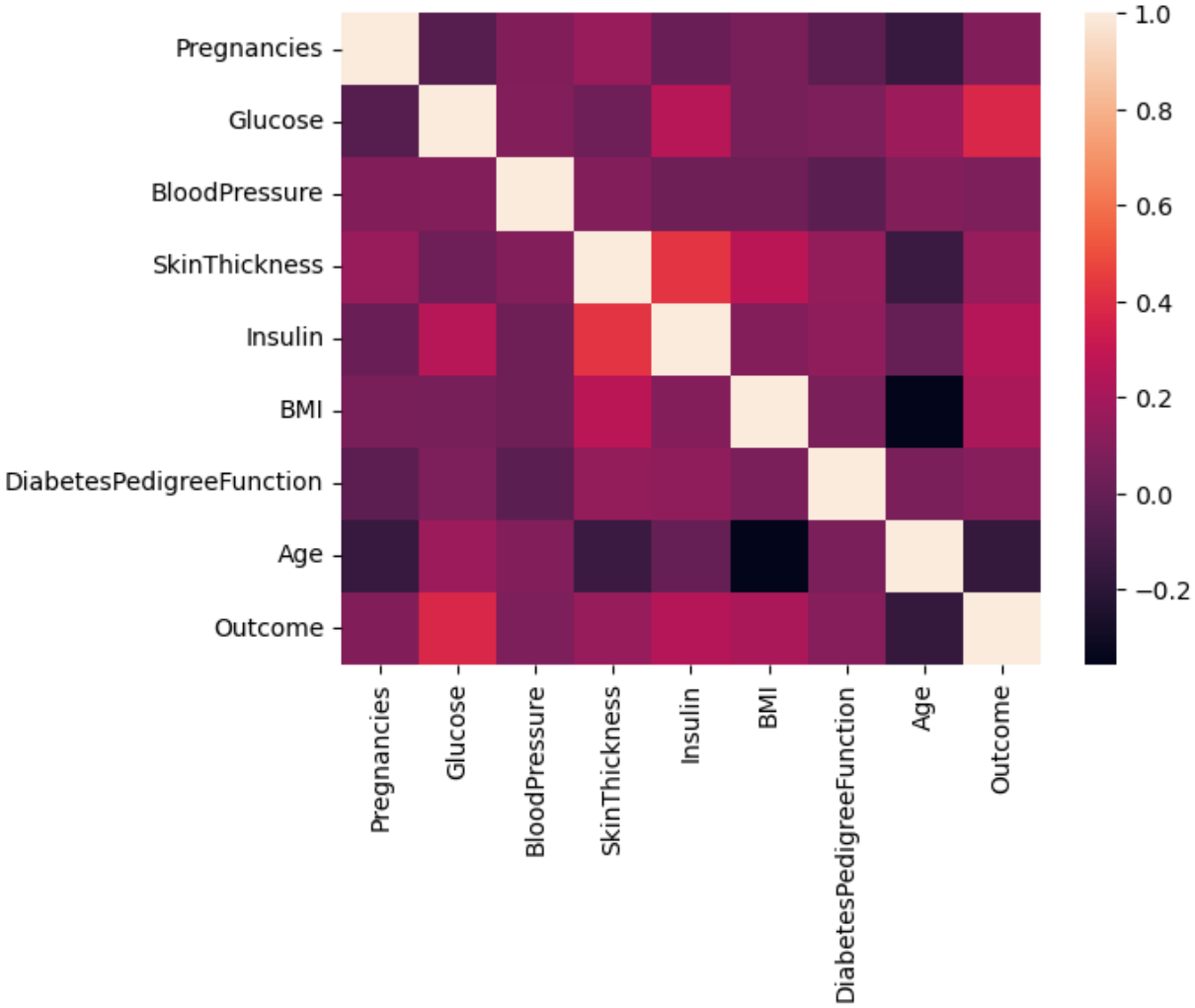
In [90]:
```
YOUNG=YoungDf.corr(method='pearson')
sns.heatmap(YOUNG)
```

Out[90]:
```
<Axes: >
```



In [91]:
```
OLD=OldDf.corr(method='pearson')
sns.heatmap(OLD)
```

Out[91]:
```
<Axes: >
```



Analysis

As expected glucose stands out when the outcome wether a person have diabetes or not although i kind of expected it to be around 0.9 like a linear relationship but i guess its not that of a universal standard as there seems to be a lot of people that have an above average glucose. What i'm suprised though is the age not having much correlation and old age seems to have much less correlation than being young as i've compared the two correlation heatmaps of old and young people that are tested for diabetes.

Conclusions/observations/analysis

In this activity, i've learned how to interact with text data set and set it up for pandas to be read in a csv format. This one though I had to set up a white space delimiter for it to properly divide the columns. This was also the time that i had the easiest way of finding the correlation of variables in the data set and while there's not much by the standards i refer it did help with those visualization i've acquired with seaborn. The context in which the data comes from also allows me to infer the standard of correlation coefficient with relevance to intelligence and brain imaging data.

In [ ]: