In [80]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import sklearn
import sklearn as sk
import math
from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import confusion_matrix, plot_confusion_matrix, classification_report
from pandas.plotting import scatter_matrix
```

In [81]:

```
data = pd.read_csv('diabetes.arff.csv')
```

In [82]:

```
data.head(10)
```

Out[82]:

	preg	plas	pres	skin	insu	mass	pedi	age	class
0	6	148	72	35	0	33.6	0.627	50	tested_positive
1	1	85	66	29	0	26.6	0.351	31	tested_negative
2	8	183	64	0	0	23.3	0.672	32	tested_positive
3	1	89	66	23	94	28.1	0.167	21	tested_negative
4	0	137	40	35	168	43.1	2.288	33	tested_positive
5	5	116	74	0	0	25.6	0.201	30	tested_negative
6	3	78	50	32	88	31.0	0.248	26	tested_positive
7	10	115	0	0	0	35.3	0.134	29	tested_negative
8	2	197	70	45	543	30.5	0.158	53	tested_positive
9	8	125	96	0	0	0.0	0.232	54	tested_positive

In [83]:

```
data.shape
```

Out[83]:

```
(768, 9)
```

In [68]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   preg    768 non-null    int64  
 1   plas    768 non-null    int64  
 2   pres    768 non-null    int64  
 3   skin    768 non-null    int64  
 4   insu    768 non-null    int64  
 5   mass    768 non-null    float64
 6   pedi    768 non-null    float64
 7   age     768 non-null    int64  
 8   class   768 non-null    object 
```

```
 8   class   768 non-null    object
dtypes: float64(2), int64(6), object(1)
memory usage: 54.1+ KB
```

In [84]:

```
data
```

Out[84]:

	preg	plas	pres	skin	insu	mass	pedi	age	class
0	6	148	72	35	0	33.6	0.627	50	tested_positive
1	1	85	66	29	0	26.6	0.351	31	tested_negative
2	8	183	64	0	0	23.3	0.672	32	tested_positive
3	1	89	66	23	94	28.1	0.167	21	tested_negative
4	0	137	40	35	168	43.1	2.288	33	tested_positive
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	tested_negative
764	2	122	70	27	0	36.8	0.340	27	tested_negative
765	5	121	72	23	112	26.2	0.245	30	tested_negative
766	1	126	60	0	0	30.1	0.349	47	tested_positive
767	1	93	70	31	0	30.4	0.315	23	tested_negative

768 rows × 9 columns

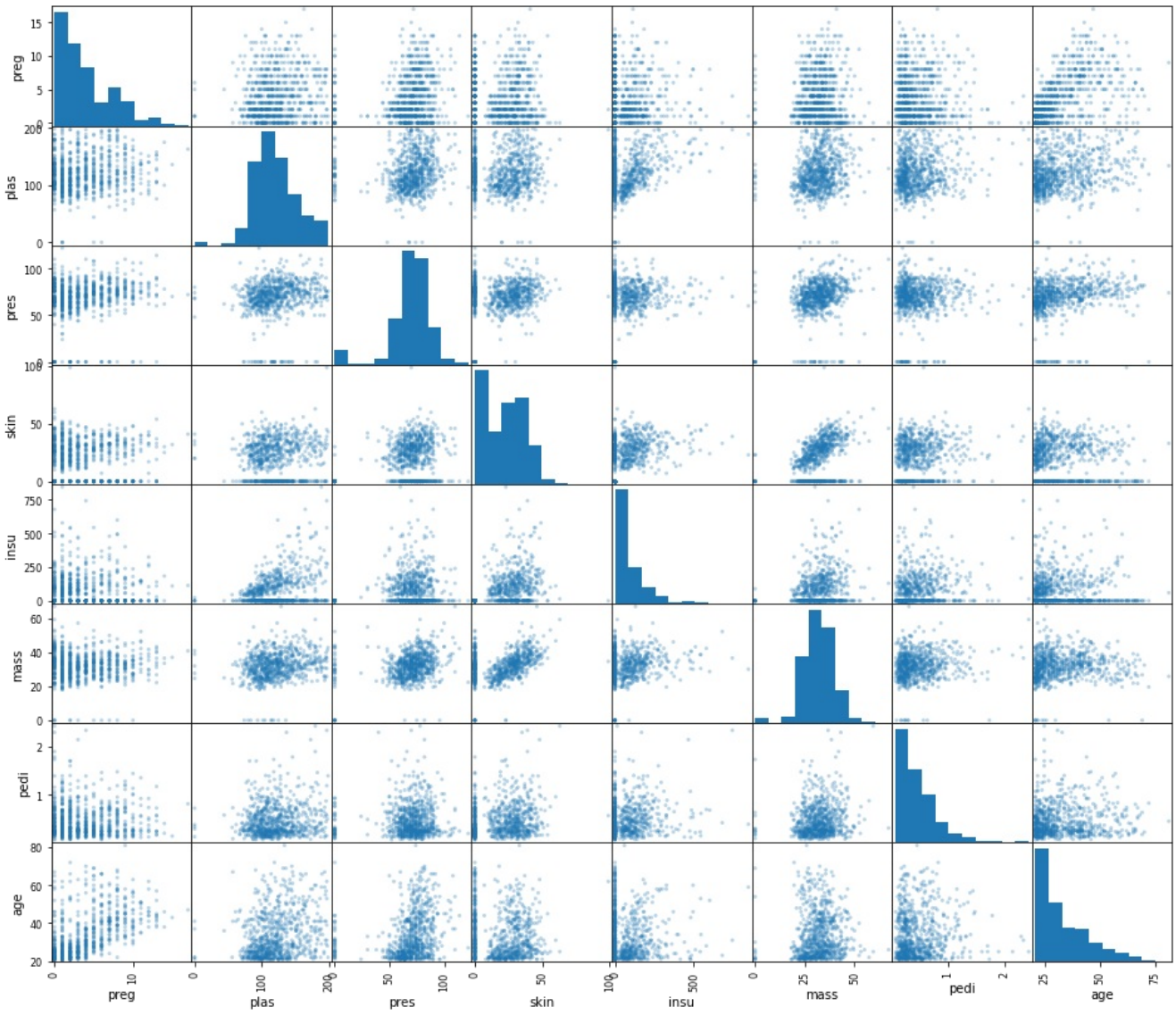
In [69]:

```
pd.plotting.scatter_matrix(data.loc[:,'preg':'class'],figsize = (16,14),alpha = 0.3)
```

Out[69]:

```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF881F3D0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8841790>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF886FBE0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF889A0D0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF88D44C0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8900850>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8900940>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF892BDF0>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8991640>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF89BEA90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF89EAEE0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8A1A0D0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8A4F7F0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8A79FA0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8AAE760>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8AD7EE0>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF83B9700>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF86AA3D0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF87A3430>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF88B0B50>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF89F6EB0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF633CC10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF636A100>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF63A24F0>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF63CF940>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF63FDD90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF6437220>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF6462670>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF648FAC0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF64BEF10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF64F73A0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF65247F0>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF6550C40>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF657E130>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF65B4520>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF65E1970>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF660EDC0>,
```

```
 <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF6648250>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF66756A0>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF66A1AF0>],
 [<matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF66CFF40>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8B183D0>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8B45820>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8B72C70>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8BA0160>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8BD8550>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8C09460>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8C32BE0>],
 [<matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8C683A0>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8C90B20>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8CC92E0>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8CF2A60>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8D28220>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8D509A0>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8D86160>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8DAF8E0>],
 [<matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8DD9100>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8E0F820>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8E39FA0>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8E6E760>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8E98EE0>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8ECD6A0>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8EF7E20>,
  <matplotlib.axes._subplots.AxesSubplot object at 0x0000029AF8F2E5E0>]],
dtype=object)
```



In [85]:

```
data.isnull().values.any()
```

Out[85]:

```
False
```

In [71]:

```
feature_columns = ['preg','plas','pres','skin','insu','mass','pedi','age']
predicted_class = ['class']
```

In [86]:

```
def tran_class(x):
    if x =='tested_positive':
        return 1
    if x =='tested_negative':
        return 0

data.head()
```

Out[86]:

	preg	plas	pres	skin	insu	mass	pedi	age	class
0	6	148	72	35	0	33.6	0.627	50	tested_positive
1	1	85	66	29	0	26.6	0.351	31	tested_negative
2	8	183	64	0	0	23.3	0.672	32	tested_positive
3	1	89	66	23	94	28.1	0.167	21	tested_negative
4	0	137	40	35	168	43.1	2.288	33	tested_positive

In [87]:

```
# data['class'] = data['class'].apply({ 'tested_negativ': 0, 'tested_positiv': 1 }.get)
data['class'] = data['class'].apply(lambda x: tran_class(x))
data.head()
```

Out[87]:

	preg	plas	pres	skin	insu	mass	pedi	age	class
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

In [88]:

```
data
```

Out[88]:

	preg	plas	pres	skin	insu	mass	pedi	age	class
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0

	preg	plas	pres	skin	insu	mass	pedi	age	class
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

768 rows × 9 columns

In [89]:

```
from sklearn.model_selection import train_test_split
feature_columns = ['preg','plas','pres','skin','insu','mass','pedi','age']
predicted_class = ['class']
```

In [90]:

```
x = data[feature_columns].values
y = data[predicted_class].values
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.3, random_state=10)
```

In [75]:

```
from sklearn.ensemble import RandomForestClassifier
random_forest_model = RandomForestClassifier(random_state=10)
```

In [91]:

```
random_forest_model.fit(x_train,y_train.ravel())
```

Out[91]:

```
RandomForestClassifier(random_state=10)
```

In [92]:

```
predict_train_data = random_forest_model.predict(x_test)
from sklearn import metrics
print("Accuracy = {0:.3f}".format(metrics.accuracy_score(y_test,predict_train_data)))
```

```
Accuracy = 0.749
```

In [77]:

```
from sklearn import preprocessing
```

In [93]:

```
le = preprocessing.LabelEncoder()
```

In [94]:

```
from sklearn.neighbors import KNeighborsClassifier
model = KNeighborsClassifier(n_neighbors = 15)
```

In [96]:

```
model.fit(x_train,y_train)
```

```
<ipython-input-96-4719cf73997a>:1: DataConversionWarning: A column-vector y was passed when a 1d a
rray was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  model.fit(x_train,y_train)
```

Out[96]:

```
KNeighborsClassifier(n_neighbors=15)
```

In [ ]:

```
```

Exercise 2

In [24]:

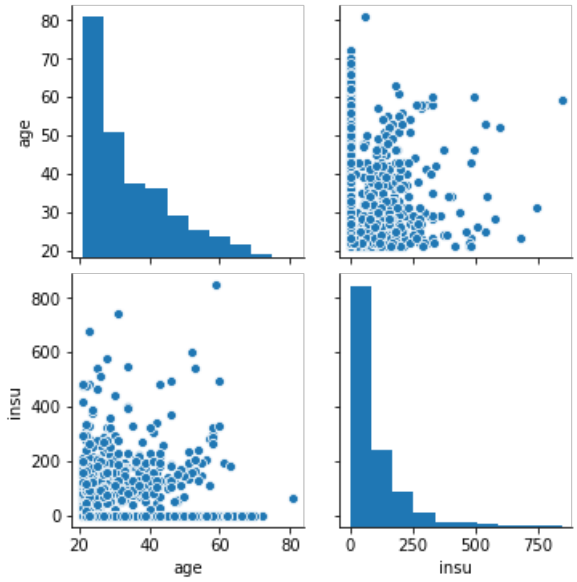
```
import seaborn
```

In [39]:

```
seaborn.pairplot(data[['age','insu']])
```

Out[39]:

```
<seaborn.axisgrid.PairGrid at 0x29af7e89cd0>
```



In [40]:

```
import sklearn.cluster as cluster
```

In [41]:

```
kmeans = cluster.KMeans(n_clusters = 2)
```

In [43]:

```
kmeans = kmeans.fit(data[['age','insu']])
```

In [46]:

```
kmeans.cluster_centers_
```

Out[46]:

```
array([[ 33.20621931,  33.6710311 ],
       [ 33.37579618, 259.31847134]])
```

In [48]:

```
data['age_clusters'] = kmeans.labels_
```

In [49]:

```
data
```

Out[49]:

	preg	plas	pres	skin	insu	mass	pedi	age	class	age_clusters
0	6	148	72	35	0	33.6	0.627	50	1	0
1	1	85	66	29	0	26.6	0.351	31	0	0
2	8	183	64	0	0	23.3	0.672	32	1	0
3	1	89	66	23	94	28.1	0.167	21	0	0
4	0	137	40	35	168	43.1	2.288	33	1	1
...	...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0	1
764	2	122	70	27	0	36.8	0.340	27	0	0
765	5	121	72	23	112	26.2	0.245	30	0	0
766	1	126	60	0	0	30.1	0.349	47	1	0
767	1	93	70	31	0	30.4	0.315	23	0	0

768 rows × 10 columns

In [50]:

```
data['age_clusters'].value_counts()
```

Out[50]:

```
0    611
1    157
Name: age_clusters, dtype: int64
```

In [58]:

```
data['class'] = kmeans.labels_
```

In [59]:

```
data['class'].value_counts()
```

Out[59]:

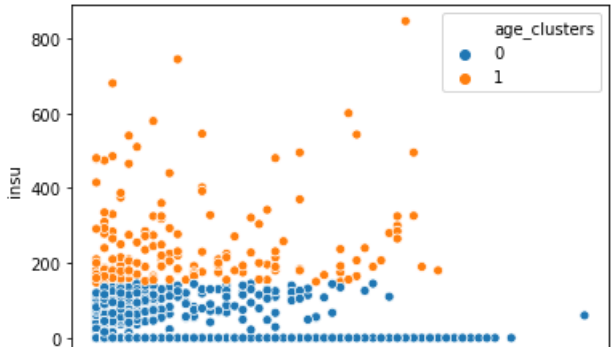
```
0    611
1    157
Name: class, dtype: int64
```

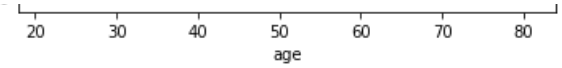
In [60]:

```
seaborn.scatterplot(x = 'age', y = 'insu',hue = 'age_clusters',data = data)
```

Out[60]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x29af8742c10>
```



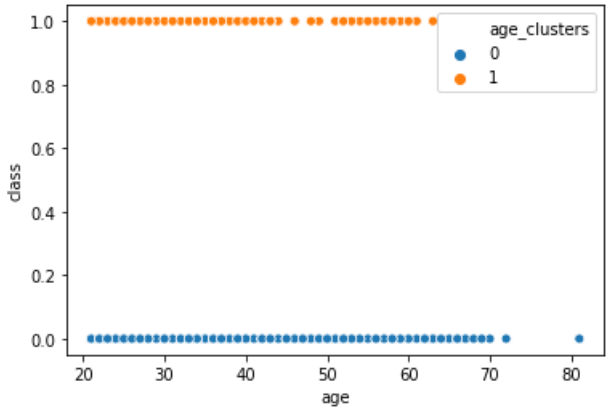


In [61]:

```
seaborn.scatterplot(x = 'age', y = 'class',hue = 'age_clusters',data = data)
```

Out[61]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x29af87ab670>
```



In [ ]:

```

```