Students exam performance

In [0]:

```
from matplotlib import pyplot as plt
```

In [1]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [0]:

```

```

In [0]:

```
plt.hist(xyz_avg[:,1])
plt.title('Average $y(t)$');
```

In [3]:

```
data = pd.read_csv('../demo/StudentsPerformance.csv')
```

In [4]:

```
data.head()
```

Out[4]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

In [5]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 8 columns):
gender                         1000 non-null object
race/ethnicity                 1000 non-null object
parental level of education    1000 non-null object
lunch                          1000 non-null object
test preparation course        1000 non-null object
math score                     1000 non-null int64
reading score                  1000 non-null int64
writing score                  1000 non-null int64
dtypes: int64(3), object(5)
memory usage: 62.6+ KB
```

In [6]:

```
data.isnull().sum()
```

Out[6]:

```
gender                         0
race/ethnicity                 0
parental level of education    0
lunch                          0
test preparation course        0
math score                     0
reading score                  0
writing score                  0
dtype: int64
```

In [7]:

```
data.corr()
```

Out[7]:

	math score	reading score	writing score
math score	1.000000	0.817580	0.802642
reading score	0.817580	1.000000	0.954598
writing score	0.802642	0.954598	1.000000

In [8]:

```
data.describe(include='all')
```

Out[8]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
count	1000	1000	1000	1000	1000	1000.00000	1000.000000	1000.000000
unique	2	5	6	2	2	NaN	NaN	NaN
top	female	group C	some college	standard	none	NaN	NaN	NaN
freq	518	319	226	645	642	NaN	NaN	NaN
mean	NaN	NaN	NaN	NaN	NaN	66.08900	69.169000	68.054000
std	NaN	NaN	NaN	NaN	NaN	15.16308	14.600192	15.195657
min	NaN	NaN	NaN	NaN	NaN	0.00000	17.000000	10.000000
25%	NaN	NaN	NaN	NaN	NaN	57.00000	59.000000	57.750000
50%	NaN	NaN	NaN	NaN	NaN	66.00000	70.000000	69.000000
75%	NaN	NaN	NaN	NaN	NaN	77.00000	79.000000	79.000000
max	NaN	NaN	NaN	NaN	NaN	100.00000	100.000000	100.000000

In [0]:

```

```

In [9]:

```
data["performance_score"]=(data["math score"]+data["reading score"]+data["writing score"]
)/3
```

In [12]:

```
passmarks=40
data["outcome"]=np.where(data["performance_score"]<passmarks,"Fail","Pass")
```

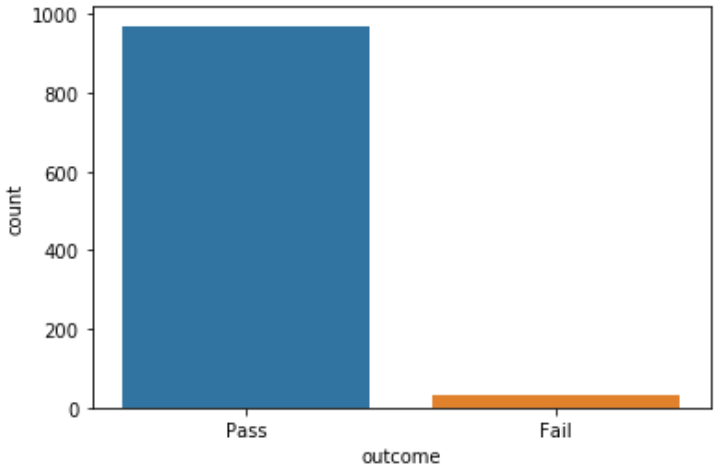
In [15]:

```
sns.countplot(x='outcome',data=data)
```

Out[15]:

<matplotlib.axes._subplots.AxesSubplot at 0x7f99842f0810>

Out[15]:



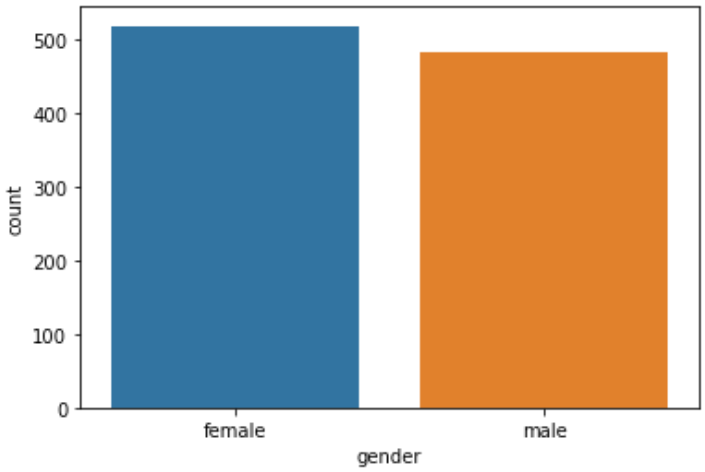
In [16]:

```
sns.countplot(x='gender',data=data)
```

Out[16]:

<matplotlib.axes._subplots.AxesSubplot at 0x7f9984220e50>

Out[16]:



In [19]:

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

Out[19]:

```
female    518
male      482
Name: gender, dtype: int64
```

In [20]:

```
data['race/ethnicity'].value_counts()
```

Out[20]:

```
group C    319
group D    262
group B    190
group E    140
group A     89
Name: race/ethnicity, dtype: int64
```

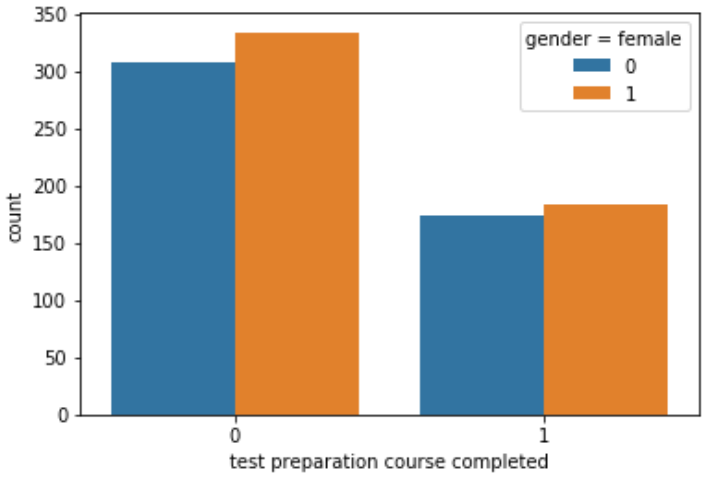
In [62]:

```
sns.countplot(data=data,x='test preparation course completed',hue='gender = female')
```

Out[62]:

<matplotlib.axes._subplots.AxesSubplot at 0x7f99841a33d0>

Out[62]:



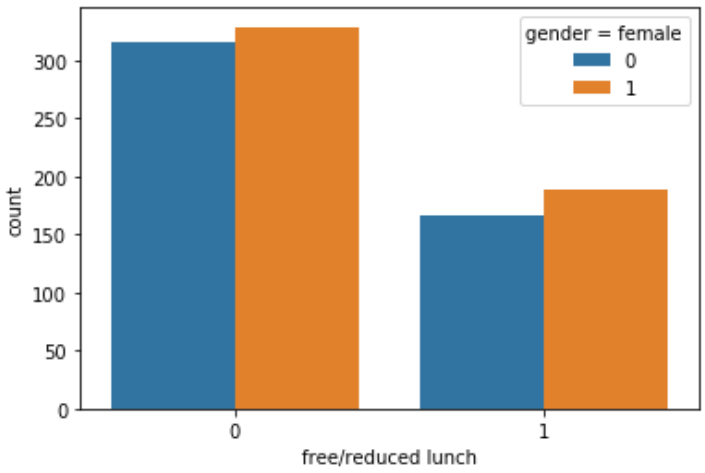
In [63]:

```
sns.countplot(data=data,x='free/reduced lunch',hue='gender = female')
```

Out[63]:

<matplotlib.axes._subplots.AxesSubplot at 0x7f9966188890>

Out[63]:



In [64]:

```
passmarks=40
data["outcome"]=np.where(data["performance_score"]<passmarks,"Fail","Pass")
```

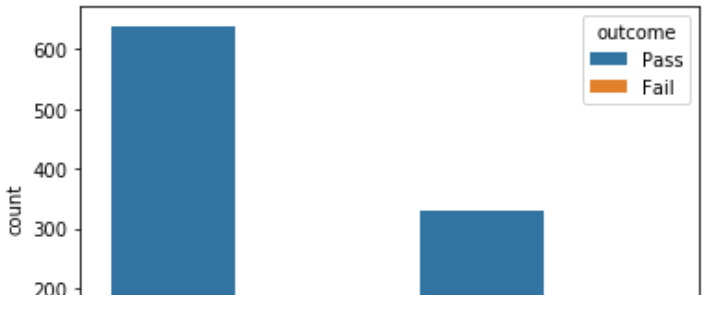
In [65]:

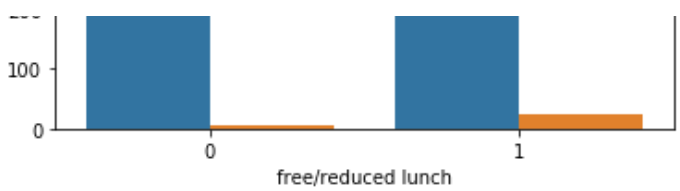
```
sns.countplot(data=data,x='free/reduced lunch',hue='outcome')
```

Out[65]:

<matplotlib.axes._subplots.AxesSubplot at 0x7f9966111f10>

Out[65]:





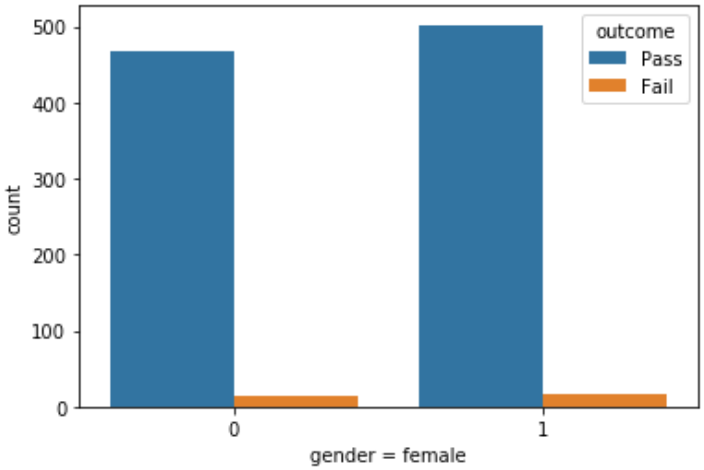
In [66]:

```
sns.countplot(data=data,x='gender = female',hue='outcome')
```

Out[66]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f9966064d90>
```

Out[66]:



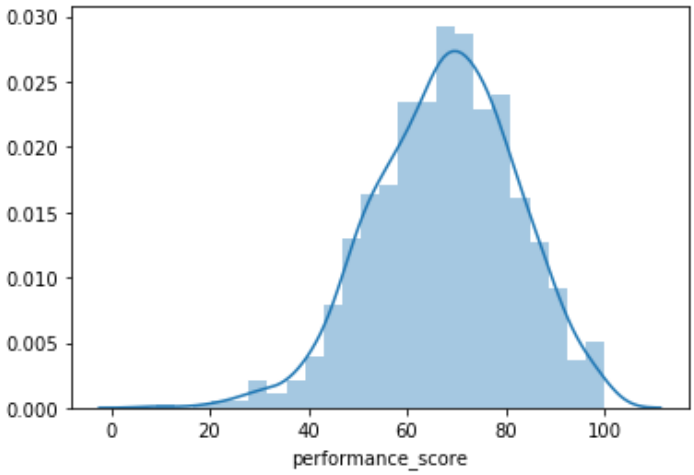
In [67]:

```
sns.distplot(data['performance_score'])
```

Out[67]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f996603d550>
```

Out[67]:



In [0]:

```

```

In [0]:

```

```