```
# coding: utf-8

# In[66]:

#Assignment 4: Creating graphs for your data
# import libraries
get_ipython().magic('matplotlib inline')
import pandas
import numpy as np
import seaborn as sn
import matplotlib.pyplot as plt

print("avoid run time error message")
pandas.set_option('display.float_format', lambda x:'%f'%x)


# In[67]:

#Import dataset
data = pandas.read_csv("gapminder.csv", low_memory = False)

#Convert all variable names to lowercaes
data.columns = map(str.lower, data.columns)


# In[103]:

# Set missing values to "nan"
data["incomeperperson"] = data["incomeperperson"].replace(0, np.nan)
data["suicideper100th"] = data["suicideper100th"].replace(0, np.nan)
data["employrate"] = data["employrate"].replace(0, np.nan)

#set avoid run time error message
data['incomeperperson'] = data['incomeperperson'].convert_objects(convert_numeric=True)
data['suicideper100th'] = data['suicideper100th'].convert_objects(convert_numeric=True)


# In[104]:

#Create varible Income Categories (based on the worldbank information)
def INCOMECAT(row):
    if row['incomeperperson'] <= 1035:
        return 1
    elif 1035 < row['incomeperperson']  <= 4085:
        return 2
    elif 4085 < row['incomeperperson'] <= 12615:
        return 3
    else:
        return 4

data["INCOMECAT"] = data.apply(lambda row: INCOMECAT(row), axis=1)
data['INCOMECAT'] = data['INCOMECAT'].astype('category')
data['INCOMECAT'] = data['INCOMECAT'].cat.rename_categories(['low','lower middle', 'upper middle','high'])


# In[ ]:

#Create varible Asia

def Asia(row):
    if row['country'] == "":
        return 1
    else:
        return 0
















# In[105]:

#Create a subset of the dataset to include only variables of interest
sub1 = data[["country", "incomeperperson","suicideper100th","employrate", "INCOMECAT"]]
print('preview dataset')
print(sub1.head(n=10))
```

```
# In[106]:

# Make a categorical count plot for the different income groups.

sn.countplot(x='INCOMECAT', data = sub1, palette = 'Greens_d')
plt.xlabel("Income Category")
plt.ylabel("count")
plt.show(block=True)


# In[107]:

#create DEVELOPED (boolean) row.

def DEVELOPED (row):
    if row["incomeperperson"] >= 12615.0:
        return 1
    else:
        return 0

data["DEVELOPED"] = data.apply(lambda row: DEVELOPED(row), axis =1)


# In[112]:

#create sub2 dataset to include only developed countries ("incomeperperson" >= 12615.0)
sub2 = sub1[(data["DEVELOPED"] != 0)]
print('preview dataset')
print(sub2.head(n=100))


# In[109]:

#Quantitative variables graphing study
#Describe each of the quantitative variables

desc1 = sub2['incomeperperson'].describe()
print(desc1)


# In[110]:

desc2 = sub2["suicideper100th"].describe()
print(desc2)
```

```
# In[111]:

#Quantitative plot study
# incomeperperson v.s suicideper100th rate (All Developed Countries)
#The plot indicates that the two variables have a low positive correlated relationship.


scat1 = sn.regplot(x='incomeperperson',y='suicideper100th', fit_reg=True, data=sub2)
plt.xlabel('incomeperperson')
plt.ylabel('suicideper100th')
plt.title("Scatterplot for the Association Between incomeperperson and suicideper100th Rate")
plt.show()


# In[127]:

#create ASIA countries (boolean) row.

def ASIA (row):
    if row["country"] == "Brunei":
        return 1
    elif row["country"] == "Cyprus":
        return 1
    elif row["country"] == "Hong Kong, China":
        return 1
    elif row["country"] == "Israel":
        return 1
    elif row["country"] == "Japan":
        return 1
    elif row["country"] == "Korea, Rep.":
        return 1
    elif row["country"] == "Macao, China":
        return 1
    elif row["country"] == "Qatar":
        return 1
    elif row["country"] == "Singapore":
        return 1
    elif row["country"] == "United Arab Emirates":
        return 1
    else:
        return 0
```

```
data["ASIA"] = data.apply(lambda row: ASIA(row), axis =1)


# In[128]:

#create sub3 dataset to include only asian developed countries ("incomeperperson" >= 12615.0)
sub3 = sub1[(data["ASIA"] == 1)]
print('preview dataset')
print(sub3.head(n=100))


# In[130]:

#Quantitative plot study
# incomeperperson v.s suicideper100th rate (All Asian Developed Countries)
#The plot indicates that the two variables have a positive correlated relationship. but also a lot of varibility.

scat2 = sn.regplot(x='incomeperperson',y='suicideper100th', fit_reg=True, data=sub3)
plt.xlabel('incomeperperson')
plt.ylabel('suicideper100th')
plt.title("Scatterplot for the Association Between incomeperperson and suicideper100th Rate")
plt.show()


# In[ ]:
```