ook604b3ca9cb Draft saved

t View Run Add-ons Help

Share | Save Version 0

Run All | Code | Draft Session (7m)

Data + Add data

input (1.43 KB)
countries-with-gdp
output (44.1MB / 19.6GB)
/kaggle/working

Settings

Language Python

Environment Preferences

Want more power? Access free GPU or turn on an internet connection. Get phone verified

Code Help

Find Code Help

Search for examples of how to do things

```python
#import Libraries
import numpy as np
```

```python
#getting Data From File
fileCountryWithGDP = open("../input/countries-with-gdp/Countries with GDP.txt","r")
listCountryWithGDP = fileCountryWithGDP.readlines()
print(listCountryWithGDP)
```

```
['List of countries:\n', "'Algeria','Angola','Argentina','Australia','Austria','Bahamas','Bangladesh','Belarus','Belgium','Bhutan','Brazil','Bulgari
a','Cambodia','Cameroon','Chile','China','Colombia','Cyprus','Denmark','El Salvador','Estonia','Ethiopia','Fiji','Finland','France','Georgia','Ghan
a','Grenada','Guinea','Haiti','Honduras','Hungary','India','Indonesia','Ireland','Italy','Japan','Kenya', 'South Korea','Liberia','Malaysia','Mexic
o', 'Morocco','Nepal','New Zealand','Norway','Pakistan', 'Peru','Qatar','Russia','Singapore','South Africa','Spain','Sweden','Switzerland','Thailan
d', 'United Arab Emirates','United Kingdom','United States','Uruguay','Venezuela','Vietnam','Zimbabwe'\n", 'GDP values for each country:\n', '2255.22
5482,629.9553062,11601.63022,25306.82494,27266.40335,19466.99052,588.3691778,2890.345675,24733.62696,1445.760002,4803.398244,2618.876037,590.4521124,
665.7982328,7122.938458,2639.54156,3362.4656,15378.16704,30860.12808,2579.115607,6525.541272,229.6769525,2242.689259,27570.4852,23016.84778,1334.6467
73,402.6953275,6047.200797,394.1156638,385.5793827,1414.072488,5745.981529,837.7464011,1206.991065,27715.52837,18937.24998,39578.07441,478.2194906,16
684.21278,279.2204061,5345.213415,6288.25324,1908.304416,274.8728621,14646.42094,40034.85063,672.1547506,3359.517402,36152.66676,3054.727742,33529.83
052,3825.093781,15428.32098,33630.24604,39170.41371,2699.123242,21058.43643,28272.40661,37691.02733,9581.05659,5671.912202,757.4009286,347.7456605']
```

```python
# there are four lines in files and second and fourth lines are of our concern
strCountries = listCountryWithGDP[1]
strCountriesGDP =listCountryWithGDP[3]
```

```python
print(strCountries)
```

```
'Algeria','Angola','Argentina','Australia','Austria','Bahamas','Bangladesh','Belarus','Belgium','Bhutan','Brazil','Bulgaria','Cambodia','Cameroon','C
hile','China','Colombia','Cyprus','Denmark','El Salvador','Estonia','Ethiopia','Fiji','Finland','France','Georgia','Ghana','Grenada','Guinea','Hait
i','Honduras','Hungary','India','Indonesia','Ireland','Italy','Japan','Kenya', 'South Korea','Liberia','Malaysia','Mexico', 'Morocco','Nepal','New Ze
aland','Norway','Pakistan', 'Peru','Qatar','Russia','Singapore','South Africa','Spain','Sweden','Switzerland','Thailand', 'United Arab Emirates','Uni
ted Kingdom','United States','Uruguay','Venezuela','Vietnam','Zimbabwe'
```

```python
print(strCountriesGDP)
```

```
2255.225482,629.9553062,11601.63022,25306.82494,27266.40335,19466.99052,588.3691778,2890.345675,24733.62696,1445.760002,4803.398244,2618.876037,590.4
521124,665.7982328,7122.938458,2639.54156,3362.4656,15378.16704,30860.12808,2579.115607,6525.541272,229.6769525,2242.689259,27570.4852,23016.84778,13
34.646773,402.6953275,6047.200797,394.1156638,385.5793827,1414.072488,5745.981529,837.7464011,1206.991065,27715.52837,18937.24998,39578.07441,478.219
4906,16684.21278,279.2204061,5345.213415,6288.25324,1908.304416,274.8728621,14646.42094,40034.85063,672.1547506,3359.517402,36152.66676,3054.727742,3
3529.83052,3825.093781,15428.32098,33630.24604,39170.41371,2699.123242,21058.43643,28272.40661,37691.02733,9581.05659,5671.912202,757.4009286,347.745
6605
```

```python
# Convert string into numpy array
listCountries = strCountries.split(",") # Converting String to list
arrCountries = np.array(listCountries) # Converting List to numpy
print(arrCountries)
```

```
["'Algeria'" "'Angola'" "'Argentina'" "'Australia'" "'Austria'"
 "'Bahamas'" "'Bangladesh'" "'Belarus'" "'Belgium'" "'Bhutan'" "'Brazil'"
 "'Bulgaria'" "'Cambodia'" "'Cameroon'" "'Chile'" "'China'" "'Colombia'"
 "'Cyprus'" "'Denmark'" "'El Salvador'" "'Estonia'" "'Ethiopia'" "'Fiji'"
 "'Finland'" "'France'" "'Georgia'" "'Ghana'" "'Grenada'" "'Guinea'"
 "'Haiti'" "'Honduras'" "'Hungary'" "'India'" "'Indonesia'" "'Ireland'"
 "'Italy'" "'Japan'" "'Kenya'" " 'South Korea'" "'Liberia'" "'Malaysia'"
 "'Mexico'" " 'Morocco'" "'Nepal'" "'New Zealand'" "'Norway'" "'Pakistan'"
 " 'Peru'" "'Qatar'" "'Russia'" "'Singapore'" "'South Africa'" "'Spain'"
 "'Sweden'" "'Switzerland'" "'Thailand'" " 'United Arab Emirates'"
 "'United Kingdom'" "'United States'" "'Uruguay'" "'Venezuela'"
 "'Vietnam'" "'Zimbabwe'\n"]
```

```python
#same process for country
listCountriesGDP = strCountriesGDP.split(",")
arrCountriesGDP = np.array(listCountriesGDP)
print(arrCountriesGDP)
```

```
['2255.225482' '629.9553062' '11601.63022' '25306.82494' '27266.40335'
 '19466.99052' '588.3691778' '2890.345675' '24733.62696' '1445.760002'
 '4803.398244' '2618.876037' '590.4521124' '665.7982328' '7122.938458'
 '2639.54156' '3362.4656' '15378.16704' '30860.12808' '2579.115607'
 '6525.541272' '229.6769525' '2242.689259' '27570.4852' '23016.84778'
 '1334.646773' '402.6953275' '6047.200797' '394.1156638' '385.5793827'
 '1414.072488' '5745.981529' '837.7464011' '1206.991065' '27715.52837'
 '18937.24998' '39578.07441' '478.2194906' '16684.21278' '279.2204061'
 '5345.213415' '6288.25324' '1908.304416' '274.8728621' '14646.42094'
 '40034.85063' '672.1547506' '3359.517402' '36152.66676' '3054.727742'
 '33529.83052' '3825.093781' '15428.32098' '33630.24604' '39170.41371'
 '2699.123242' '21058.43643' '28272.40661' '37691.02733' '9581.05659'
 '5671.912202' '757.4009286' '347.7456605']
```

[15]:
```python
# Now we have two numpy array one for countries and another for GDP
# lets check whether both have same elements or not to check gdp corresponding to the country
arrCountriesGDP.size == arrCountries.size
```

[15]:
```
True
```

[16]:
```python
# Both datasets have equal number of elements now we have accurate data so we can proceed further
# now lets clean our data values inside array lets filter countries array first
arrCountries
```

[16]:
```
array(["'Algeria'", "'Angola'", "'Argentina'", "'Australia'", "'Austria'",
       "'Bahamas'", "'Bangladesh'", "'Belarus'", "'Belgium'", "'Bhutan'",
       "'Brazil'", "'Bulgaria'", "'Cambodia'", "'Cameroon'", "'Chile'",
       "'China'", "'Colombia'", "'Cyprus'", "'Denmark'", "'El Salvador'",
       "'Estonia'", "'Ethiopia'", "'Fiji'", "'Finland'", "'France'",
       "'Georgia'", "'Ghana'", "'Grenada'", "'Guinea'", "'Haiti'",
       "'Honduras'", "'Hungary'", "'India'", "'Indonesia'", "'Ireland'",
       "'Italy'", "'Japan'", "'Kenya'", " 'South Korea'", "'Liberia'",
       "'Malaysia'", "'Mexico'", " 'Morocco'", "'Nepal'", "'New Zealand'",
       "'Norway'", "'Pakistan'", " 'Peru'", "'Qatar'", "'Russia'",
       "'Singapore'", "'South Africa'", "'Spain'", "'Sweden'",
       "'Switzerland'", "'Thailand'", " 'United Arab Emirates'",
       "'United Kingdom'", "'United States'", "'Uruguay'", "'Venezuela'",
       "'Vietnam'", "'Zimbabwe'\n"], dtype='<U23')
```

[17]:
```python
# Countries name have extra single quotes and a \n is also there with the last element
arrCountries = np.chararray.replace(arrCountries,"\n","")
arrCountries = np.chararray.replace(arrCountries,"'","")
print(arrCountries)
```

```
['Algeria' 'Angola' 'Argentina' 'Australia' 'Austria' 'Bahamas'
 'Bangladesh' 'Belarus' 'Belgium' 'Bhutan' 'Brazil' 'Bulgaria' 'Cambodia'
 'Cameroon' 'Chile' 'China' 'Colombia' 'Cyprus' 'Denmark' 'El Salvador'
 'Estonia' 'Ethiopia' 'Fiji' 'Finland' 'France' 'Georgia' 'Ghana'
 'Grenada' 'Guinea' 'Haiti' 'Honduras' 'Hungary' 'India' 'Indonesia'
 'Ireland' 'Italy' 'Japan' 'Kenya' ' South Korea' 'Liberia' 'Malaysia'
 'Mexico' ' Morocco' 'Nepal' 'New Zealand' 'Norway' 'Pakistan' ' Peru'
 'Qatar' 'Russia' 'Singapore' 'South Africa' 'Spain' 'Sweden'
 'Switzerland' 'Thailand' ' United Arab Emirates' 'United Kingdom'
 'United States' 'Uruguay' 'Venezuela' 'Vietnam' 'Zimbabwe']
```

[18]:
```python
arrCountriesGDP
```

[18]:
```
array(['2255.225482', '629.9553062', '11601.63022', '25306.82494',
       '27266.40335', '19466.99052', '588.3691778', '2890.345675',
       '24733.62696', '1445.760002', '4803.398244', '2618.876037',
       '590.4521124', '665.7982328', '7122.938458', '2639.54156',
       '3362.4656', '15378.16704', '30860.12808', '2579.115607',
       '6525.541272', '229.6769525', '2242.689259', '27570.4852',
       '23016.84778', '1334.646773', '402.6953275', '6047.200797',
       '394.1156638', '385.5793827', '1414.072488', '5745.981529',
       '837.7464011', '1206.991065', '27715.52837', '18937.24998',
       '39578.07441', '478.2194906', '16684.21278', '279.2204061',
       '5345.213415', '6288.25324', '1908.304416', '274.8728621',
       '14646.42094', '40034.85063', '672.1547506', '3359.517402',
       '36152.66676', '3054.727742', '33529.83052', '3825.093781',
```

[19]:
```python
#now Comes the main part

# find and print the name of country with highest GDP
#typecasting of GDP to float as we need to
arrCountriesGDP = arrCountriesGDP.astype('float')
arrCountries[arrCountriesGDP.argmax()]
```

[19]: 'Norway'

[20]:
```python
#find and print the name of country with lowest GDP
arrCountries[arrCountriesGDP.argmin()]
```

[20]: 'Ethiopia'

[21]:
```python
#print out text and input values iteratively
for country in arrCountries:
    print(country)
```

```
Algeria
Angola
Argentina
Australia
```

[22]:
```python
#print out the entire list of countries with their GDPs
for index,country in enumerate(arrCountries):
    print("Country : ",country.strip(),"| GDP: ",arrCountriesGDP[index])
```

```
Country :  Algeria | GDP:  2255.225482
Country :  Angola | GDP:  629.9553062
Country :  Argentina | GDP:  11601.63022
Country :  Australia | GDP:  25306.82494
Country :  Austria | GDP:  27266.40335
Country :  Bahamas | GDP:  19466.99052
Country :  Bangladesh | GDP:  588.3691778
Country :  Belarus | GDP:  2890.345675
Country :  Belgium | GDP:  24733.62696
Country :  Bhutan | GDP:  1445.760002
Country :  Brazil | GDP:  4803.398244
Country :  Bulgaria | GDP:  2618.876037
Country :  Cambodia | GDP:  590.4521124
Country :  Cameroon | GDP:  665.7982328
Country :  Chile | GDP:  7122.938458
Country :  China | GDP:  2639.54156
Country :  Colombia | GDP:  3362.4656
Country :  Cyprus | GDP:  15378.16704
Country :  Denmark | GDP:  30860.12808
Country :  El Salvador | GDP:  2579.115607
Country :  Estonia | GDP:  6525.541272
Country :  Ethiopia | GDP:  229.6769525
Country :  Fiji | GDP:  2242.689259
Country :  Finland | GDP:  27570.4852
Country :  France | GDP:  23016.84778
Country :  Georgia | GDP:  1334.646773
Country :  Ghana | GDP:  402.6953275
Country :  Grenada | GDP:  6047.200797
Country :  Guinea | GDP:  394.1156638
Country :  Haiti | GDP:  385.5793827
Country :  Honduras | GDP:  1414.072488
```

Run All Code Draft Session (13m)

[23]:
```
# Print the highest GDP value, lowest GDP value, mean GDP value, standardized GDP value and sum of all GDP values
max(arrCountriesGDP)
```

[23]: 40034.85063

[24]:
```
min(arrCountriesGDP)
```

[24]: 229.6769525

[25]:
```
arrCountriesGDP.mean()
```

[25]: 11289.409271639683

[26]:
```
arrCountriesGDP.std()
```

[26]: 12743.828910617945

Data + Add data
input (1.43 KB)
countries-with-gdp
output (44.1MB / 19.6GB)
/kaggle/working

Settings
Language Python
Environment Preferences

Want more power? Access free GPU or turn on an internet connection. Get phone verified

Code Help
Find Code Help
Search for examples of how to do things

[26]:
```
arrCountriesGDP.std()
```

[26]: 12743.828910617945

[27]:
```
sum(arrCountriesGDP)
```

[27]: 711232.7841133001

+ Code + Markdown

Code Help
Find Code Help
Search for examples of how to do things