Overall Analysis:

- Happiness Score is strongly related to GDP & Health.
- North America is consistently the happiest region over last 3 years
- All top 5 countries over last 3 years are from Europe region
- Finland has consistently topped being the happiest country in last 3 years
- Bottom 5 countries over last 3 years are mostly from Africa Region
- Commonwealth & Europe region had no outliers over last 3 years
- The country that experienced a significant increase in happiness is Benin, Africa. There is a moderate increase after calcualting the delta score is 1.075.
- Afghanistan has experienced a significant decline over the years furthersome, after calculating the delta score, the country is, scoring -1.0651.
- The happiness score in US from year 2018 to 2019 has shown increasing trend.
- Rate of hapiness from 2019 to 2020 is far higher than the rate from 2018 to 2019
- As per the r-value from the below graph, Happiness score and HDI have postive correlation.

Importing Required Modules

In [1]:
```
# Importing modules
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import linregress
import numpy as np
import requests
import os
import json
from pprint import pprint
```

Read CSV Data and Store in DataFrame

In [2]:
```
# Read file and store it in DataFrame
orig_data_2018 = pd.read_csv("Resource/2018.csv")
orig_data_2019 = pd.read_csv("Resource/2019.csv")
orig_data_2020 = pd.read_csv("Resource/2020.csv")



# Get only required columns from dataframe
data_2018 = orig_data_2018[["Country or region", "Score", "GDP per capita", "Heal
                            "Perceptions of corruption"]]

# Rename Columns so they are uniform across all years
data_2018 = data_2018.rename(columns={"Country or region" : "Country",
                                      "Healthy life expectancy" : "Health",
                                      "GDP per capita" : "GDP_per_Capita",
                                      "Perceptions of corruption" : "Corruption"}

# Get only required columns from dataframe
data_2019 = orig_data_2019[["Country or region", "Score", "GDP per capita", "Heal
                            "Perceptions of corruption"]]

# Rename Columns so they are uniform across all years
data_2019 = data_2019.rename(columns={"Country or region" : "Country",
                                      "Healthy life expectancy" : "Health",
                                      "GDP per capita" : "GDP_per_Capita",
                                      "Perceptions of corruption" : "Corruption"
                                      })

# Get only required columns from dataframe
data_2020 = orig_data_2020[["Country name", "Regional indicator", "Ladder score",
                            "Healthy life expectancy", "Perceptions of corruptior

# Rename Columns so they are uniform across all years
data_2020 = data_2020.rename(columns = {"Country name" : "Country",
                                        "Regional indicator" : "Region",
                                        "Ladder score" : "Score",
                                        "Logged GDP per capita" : "GDP_per_Capita
                                        "Healthy life expectancy" : "Health",
                                        "Perceptions of corruption" : "Corruptior
```

Data Cleanup

- Merge all three DF into one DF
- Get average of Score, GDP, Health, Corruption for all years
- Recategorize Region

In [3]:
```
# Merge all three DF to one by adding suffixes
merged_data = pd.merge(pd.merge(data_2020,data_2019,on = "Country", how="inner",
                        data_2018,on = "Country", how="inner" ,suffixes=['_2020','

# Display merged Data Frame
merged_data
```

Out[3]:

	Country	Region	Score_2020	GDP_per_Capita_2020	Health_2020	Corruption_2020	Score_
0	Finland	Western Europe	7.8087	10.639267	71.900825	0.195445	
1	Denmark	Western Europe	7.6456	10.774001	72.402504	0.168489	
2	Switzerland	Western Europe	7.5599	10.979933	74.102448	0.303728	
3	Iceland	Western Europe	7.5045	10.772559	73.000000	0.711710	
4	Norway	Western Europe	7.4880	11.087804	73.200783	0.263218	
...	...	...	...	...	...	...	
139	Central African Republic	Sub-Saharan Africa	3.4759	6.625160	45.200001	0.891807	
140	Rwanda	Sub-Saharan Africa	3.3123	7.600104	61.098846	0.183541	
141	Zimbabwe	Sub-Saharan Africa	3.2992	7.865712	55.617260	0.810237	
142	South Sudan	Sub-Saharan Africa	2.8166	7.425360	51.000000	0.763417	
143	Afghanistan	South Asia	2.5669	7.462861	52.590000	0.933687	

144 rows × 14 columns

In [4]:
```
# Check if there are any missing values in dataframe
merged_data.count()
```

Out[4]:
```
Country                 144
Region                  144
Score_2020              144
GDP_per_Capita_2020     144
Health_2020             144
Corruption_2020         144
Score_2019              144
GDP_per_Capita_2019     144
Health_2019             144
Corruption_2019         144
Score_2018              144
GDP_per_Capita_2018     144
Health_2018             144
Corruption_2018         143
dtype: int64
```

In [5]:
```
# Replace empty cell value with NaN
merged_data = merged_data.replace('', np.nan, regex=True)

# Drop any rows which contains NaN values
merged_data = merged_data.dropna(how='any')

# Re-verify if values are consistent
merged_data.count()
```

Out[5]:
```
Country                 143
Region                  143
Score_2020              143
GDP_per_Capita_2020     143
Health_2020             143
Corruption_2020         143
Score_2019              143
GDP_per_Capita_2019     143
Health_2019             143
Corruption_2019         143
Score_2018              143
GDP_per_Capita_2018     143
Health_2018             143
Corruption_2018         143
dtype: int64
```

In [6]:
```
# Get the unique region names
merged_data["Region"].unique()

# Combine (Rename) all regions which has divided into multiple regions for exampl
# "South Asia" & "Southeast Asia" as "Asia"
merged_data = merged_data.replace({"Region" : { "Central and Eastern Europe" : "E
                                                "East Asia" : "Asia",
                                                "Middle East and North Africa" :
                                                "South Asia" : "Asia",
                                                "Southeast Asia" : "Asia",
                                                "Sub-Saharan Africa" : "Africa",
                                                "Western Europe" : "Europe",
                                                "Commonwealth of Independent Sta
                                                "North America and ANZ" : "N. Am
                                                "Latin America and Caribbean" :
                                               }
                                   })

# Display unique region names
merged_data["Region"].unique()
```

Out[6]:
```
array(['Europe', 'N. America', 'Africa', 'S. America', 'Asia',
       'Commonwealth'], dtype=object)
```

In [7]:
```
# Display final Data Frame
merged_data
```

Out[7]:

	Country	Region	Score_2020	GDP_per_Capita_2020	Health_2020	Corruption_2020	Score_
0	Finland	Europe	7.8087	10.639267	71.900825	0.195445	7
1	Denmark	Europe	7.6456	10.774001	72.402504	0.168489	7
2	Switzerland	Europe	7.5599	10.979933	74.102448	0.303728	7
3	Iceland	Europe	7.5045	10.772559	73.000000	0.711710	7
4	Norway	Europe	7.4880	11.087804	73.200783	0.263218	7
...	...	...	...	...	...	...	
139	Central African Republic	Africa	3.4759	6.625160	45.200001	0.891807	3
140	Rwanda	Africa	3.3123	7.600104	61.098846	0.183541	3
141	Zimbabwe	Africa	3.2992	7.865712	55.617260	0.810237	3
142	South Sudan	Africa	2.8166	7.425360	51.000000	0.763417	2
143	Afghanistan	Asia	2.5669	7.462861	52.590000	0.933687	3

143 rows × 14 columns

Question 1 : How does happiness relates on different factors : GDP, Health & Corruption

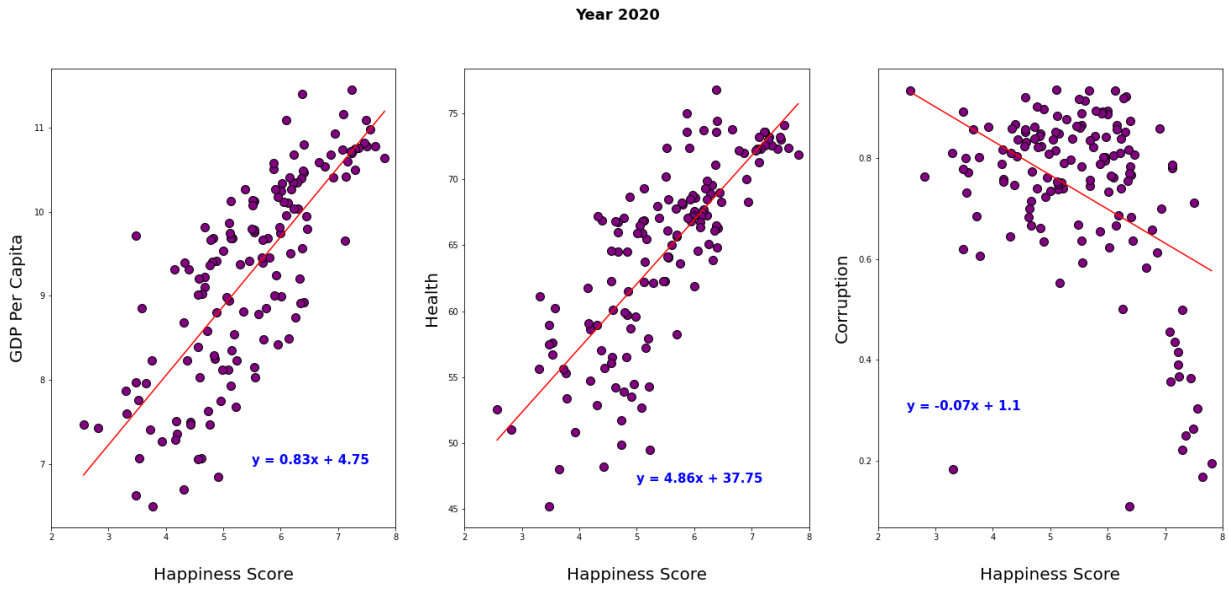
Analysis:

- Happiness score is strongly related to GDP based off of r-value.
- Happiness score is strongly related to Health based off of r-value.
- Happiness score increases as GDP & Health increases.
- Happiness score is not related to corruption based off of r-value. Looks like people are not that much concern about corruption.
- Corelation of GDP, Health to happiness score is almost same across last 3 years.

In [8]:
```
# Defined method to calculate linear regression, parameters as below:
# ax = subplot name
# x_values = series/column values to plot on x-axis
# y_values = series/column values to plot on y-axis
# ylabel = string to display as y-label
# eqplace = location to place equation on plot

def score_linear(ax, x_values, y_values, ylabel, eqplace):

    # Get the linear equation
    (slope, intercept, rvalue, pvalue, stderr) = linregress(x_values, y_values)
    regress_values = x_values * slope + intercept
    line_eq = "y = " + str(round(slope,2)) + "x + " + str(round(intercept,2))

    # Plot the scatter plot
    ax.scatter(x_values,y_values, edgecolors="black", facecolors="purple", s=100)
    ax.plot(x_values,regress_values,"r-")

    # Annotate the line_equation on graph
    ax.annotate(line_eq,eqplace,fontsize=15,color="blue",fontweight="bold")

    # Set label for x & y axis and xlim
    ax.set_xlabel("Happiness Score", labelpad=30, fontsize=20)
    ax.set_ylabel(ylabel, labelpad=10, fontsize=20)
    ax.set_xlim(2,8)


    # Print r-value
    print(f"The r-value for {ylabel} is : {rvalue}")
```

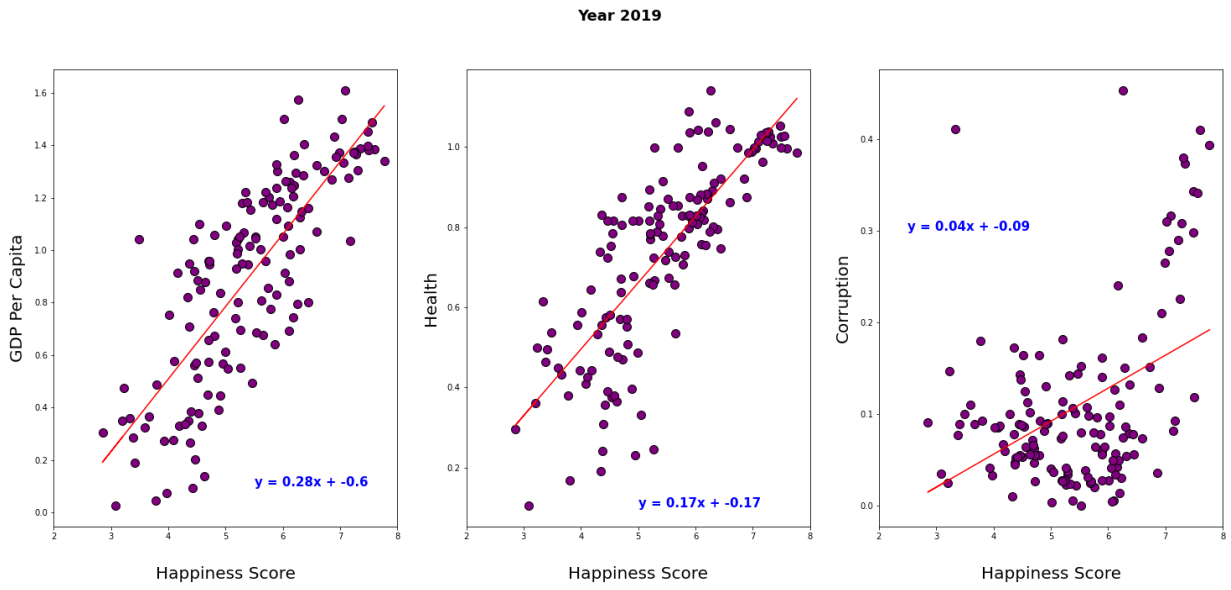
In [9]:
```
# Create a figure with 1 row and 3 columns, set figure size
fig1,(ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(25,10))

# Set the figure title
fig1.suptitle('Year 2020', fontsize=18, fontweight='bold')

# Call method score_linear by providing required parameters to plot linear regres

# Happiness score vs GDP for year 2020:
score_linear(ax1, merged_data["Score_2020"], merged_data["GDP_per_Capita_2020"],
# Happiness score vs Health for year 2020:
score_linear(ax2, merged_data["Score_2020"], merged_data["Health_2020"], "Health"
# Happiness score vs Corruption for year 2020:
score_linear(ax3, merged_data["Score_2020"], merged_data["Corruption_2020"], "Cor

# Save fig
fig1.savefig("Plot_Images/2020_regression")

# Display Plot
plt.show()
```

```
The r-value for GDP Per Capita is : 0.7803058524869457
The r-value for Health is : 0.7866048436107645
The r-value for Corruption is : -0.43026737412953575
```



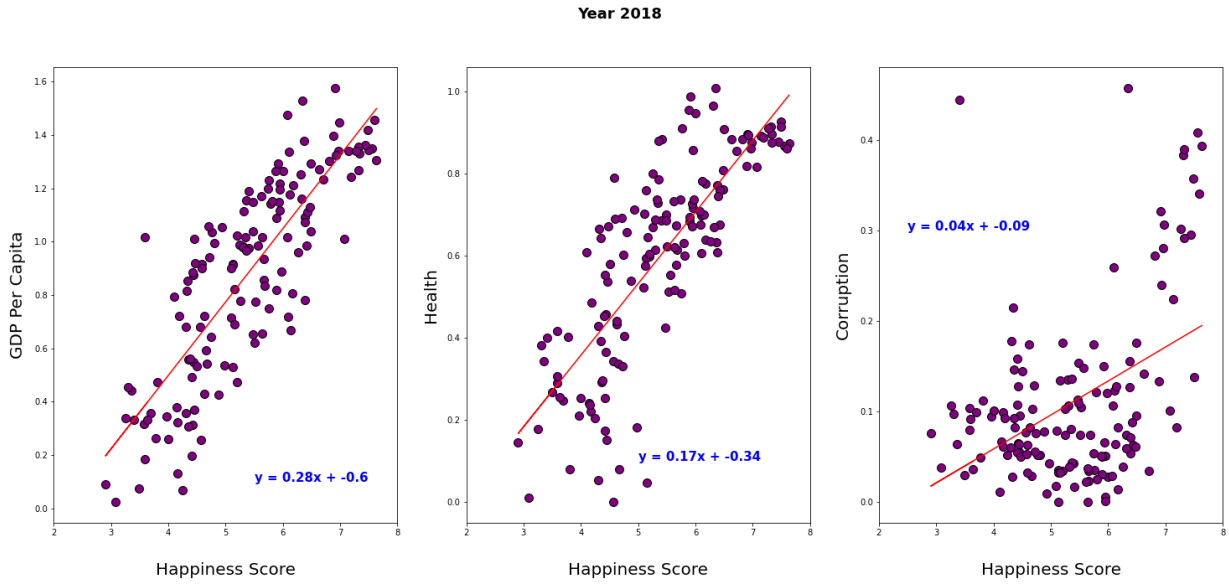
In [10]:
```
# Create a figure with 1 row and 3 columns, set figure size
fig1,(ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(25,10))

# Set the figure title
fig1.suptitle('Year 2019', fontsize=18, fontweight='bold')

# Call method score_linear by providing required parameters to plot linear regres

# Happiness score vs GDP for year 2019:
score_linear(ax1, merged_data["Score_2019"], merged_data["GDP_per_Capita_2019"],
# Happiness score vs Health for year 2019:
score_linear(ax2, merged_data["Score_2019"], merged_data["Health_2019"], "Health"
# Happiness score vs Corruption for year 2019:
score_linear(ax3, merged_data["Score_2019"], merged_data["Corruption_2019"], "Cor

# Save fig
fig1.savefig("Plot_Images/2019_regression")

# Display Plot
plt.show()
```

```
The r-value for GDP Per Capita is : 0.8012042082191385
The r-value for Health is : 0.7950662928800899
The r-value for Corruption is : 0.42308551500872704
```



In [11]:
```
# Create a figure with 1 row and 3 columns, set figure size
fig1,(ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(25,10))

# Set the figure title
fig1.suptitle('Year 2018', fontsize=18, fontweight='bold')

# Call method score_linear by providing required parameters to plot linear regres

# Happiness score vs GDP for year 2018:
score_linear(ax1, merged_data["Score_2018"], merged_data["GDP_per_Capita_2018"],
# Happiness score vs Health for year 2018:
score_linear(ax2, merged_data["Score_2018"], merged_data["Health_2018"], "Health'
# Happiness score vs Corruption for year 2018:
score_linear(ax3, merged_data["Score_2018"], merged_data["Corruption_2018"], "Cor

# Save fig
fig1.savefig("Plot_Images/2018_regression")

# Display Plot
plt.show()
```

```
The r-value for GDP Per Capita is : 0.8267674298914323
The r-value for Health is : 0.7968283105259638
The r-value for Corruption is : 0.43477457143982645
```



Question 2: What are the top 5 & bottom 5 countries in regards with happiness Score? Are there any similarities in demographics in top 5 and Bottom 5 countries?

Analysis:

- All top 5 countries are from Europe region for all 3 years
- Finland maintained its position of the most happiest country all over years
- Most of Bottom 5 countries are from Africa region over all 3 years
- Could not comapre similarities in terms of demographics as health measures are different in year 2020 compare to 2018 & 2019
- For top 5 countries in year 2018 & 2019 there is not much differnce between Health & GDP demographics
- Looking at bottom 5 countries in year 2018 & 2019, looks like country Central African Republic is an exception as its demographics are lower than that of the bottom most country still it has greater happiness score.

Top 5 countries

In [12]:
```
# Sort the data frame by Happiness score in particular year in descending order t
top_5_2020 = merged_data.sort_values(by=["Score_2020"], ascending=False).head(5)
top_5_2019 = merged_data.sort_values(by=["Score_2019"], ascending=False).head(5)
top_5_2018 = merged_data.sort_values(by=["Score_2018"], ascending=False).head(5)

# Get only required columns in dataframe
top_5_2020 = top_5_2020[["Country", "Region", "Score_2020", "GDP_per_Capita_2020"
top_5_2019 = top_5_2019[["Country", "Region", "Score_2019", "GDP_per_Capita_2019"
top_5_2018 = top_5_2018[["Country", "Region", "Score_2018", "GDP_per_Capita_2018"

# Display top 5 countries for year 2020
top_5_2020
```

Out[12]:

	Country	Region	Score_2020	GDP_per_Capita_2020	Health_2020	Corruption_2020
0	Finland	Europe	7.8087	10.639267	71.900825	0.195445
1	Denmark	Europe	7.6456	10.774001	72.402504	0.168489
2	Switzerland	Europe	7.5599	10.979933	74.102448	0.303728
3	Iceland	Europe	7.5045	10.772559	73.000000	0.711710
4	Norway	Europe	7.4880	11.087804	73.200783	0.263218

In [13]:
```
# Display top 5 countries for year 2019
top_5_2019
```

Out[13]:

	Country	Region	Score_2019	GDP_per_Capita_2019	Health_2019	Corruption_2019
0	Finland	Europe	7.769	1.340	0.986	0.393
1	Denmark	Europe	7.600	1.383	0.996	0.410
4	Norway	Europe	7.554	1.488	1.028	0.341
3	Iceland	Europe	7.494	1.380	1.026	0.118
5	Netherlands	Europe	7.488	1.396	0.999	0.298

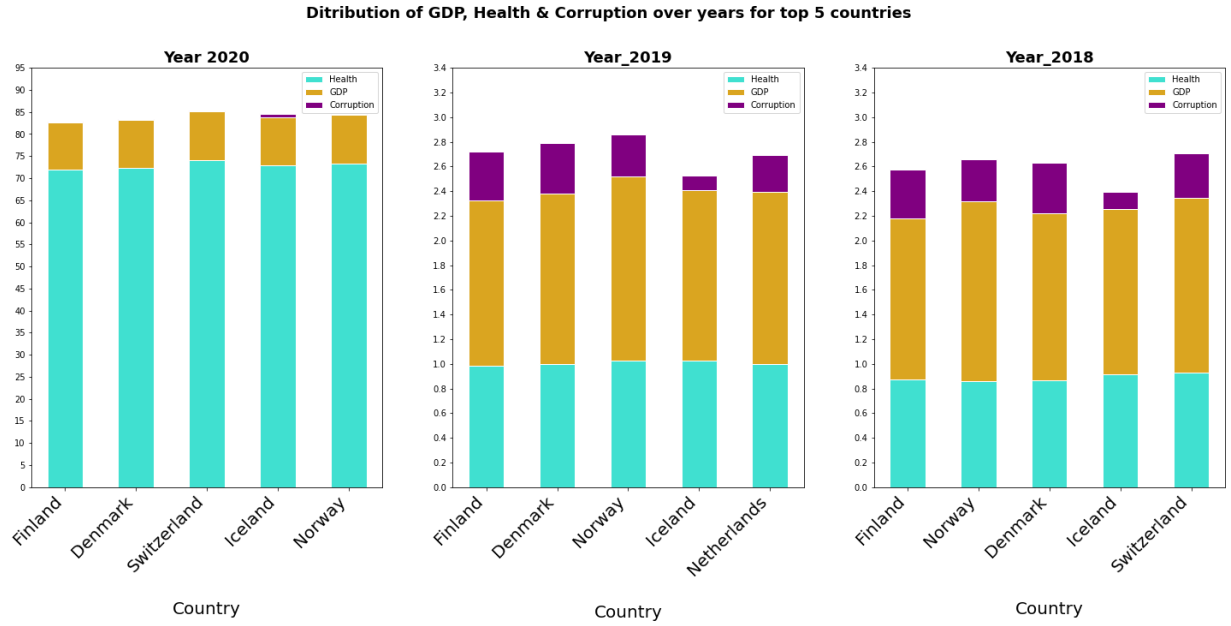
In [14]:
```
# Display top 5 countries for year 2018
top_5_2018
```

Out[14]:

	Country	Region	Score_2018	GDP_per_Capita_2018	Health_2018	Corruption_2018
0	Finland	Europe	7.632	1.305	0.874	0.393
4	Norway	Europe	7.594	1.456	0.861	0.340
1	Denmark	Europe	7.555	1.351	0.868	0.408
3	Iceland	Europe	7.495	1.343	0.914	0.138
2	Switzerland	Europe	7.487	1.420	0.927	0.357

In [15]:
```
# Method to plot stacked bar graph to display distribution of demographics for ea
# ax = subplot
# ax_title = subplot title
# x_value = x-axis values(ticks), which is top 5 country names for particular yea
# y_value1 = y-axis values, which is Health score
# y_value2 = y-axis values, which is GDP score
# y_value3 = y-axis values, which is Corruption score
# max_yticks = float/int, max y limit
# spacing = float/integer, interval value to place y values on axis

def year_stackBar(ax, ax_title, x_value, y_value1, y_value2, y_value3, max_yticks

    # Plot the stacked bar graph
    p1 = ax.bar(x_value, y_value1, color='turquoise', edgecolor='white', width=0.
    p2 = ax.bar(x_value, y_value2, color='goldenrod', edgecolor='white', width=0.
    p3 = ax.bar(x_value, y_value3, color='purple', edgecolor='white', width=0.5,
             bottom = y_value1 + y_value2)

    # Set the yticks, x-label & title
    ax.set_yticks(np.arange(0, max_yticks, spacing))
    ax.set_xlabel("Country", labelpad=30, fontsize=20)
    ax.set_title(ax_title, fontsize=18, fontweight='bold')
    ax.tick_params(axis='x', labelsize=20)

    # Add legend for the plot
    ax.legend((p1[0], p2[0], p3[0]), ('Health', 'GDP', 'Corruption'))
```

In [16]:
```
# Draw a figure with 1 row and 2 columns and set figure size
fig1,(ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(25,10))

# Set the figure title
fig1.suptitle('Ditribution of GDP, Health & Corruption over years for top 5 count

# Call the above function year_stackBar to plot stacked bar graph of demographics

# Stacked bar graph for Year 2020
year_stackBar(ax1, "Year 2020", top_5_2020["Country"], top_5_2020["Health_2020"],
              top_5_2020["Corruption_2020"], 100, 5)

# Stacked bar graph for Year 2019
year_stackBar(ax2, "Year_2019", top_5_2019["Country"], top_5_2019["Health_2019"],
              top_5_2019["Corruption_2019"], 3.5, 0.2)

# Stacked bar graph for Year 2018
year_stackBar(ax3, "Year_2018", top_5_2018["Country"], top_5_2018["Health_2018"],
              top_5_2018["Corruption_2018"], 3.5, 0.2)

# Rotate xticks at 45 degrees
fig1.autofmt_xdate(rotation=45)

# Save fig
fig1.savefig("Plot_Images/Top5_stackedBar.png")

# Display Plot
plt.show()
```



Bottom 5 countries

In [17]:
```
# Sort the data frame by Happiness score in particular year in descending order c
bottom_5_2020 = merged_data.sort_values(by=["Score_2020"], ascending=False).tail(
bottom_5_2019 = merged_data.sort_values(by=["Score_2019"], ascending=False).tail(
bottom_5_2018 = merged_data.sort_values(by=["Score_2018"], ascending=False).tail(

# Get only required columns for that year in dataframe
bottom_5_2020 = bottom_5_2020[["Country", "Region", "Score_2020", "GDP_per_Capita
bottom_5_2019 = bottom_5_2019[["Country", "Region", "Score_2019", "GDP_per_Capita
bottom_5_2018 = bottom_5_2018[["Country", "Region", "Score_2018", "GDP_per_Capita

# Display bottom 5 countries for year 2020
bottom_5_2020
```

Out[17]:

	Country	Region	Score_2020	GDP_per_Capita_2020	Health_2020	Corruption_2020
139	Central African Republic	Africa	3.4759	6.625160	45.200001	0.891807
140	Rwanda	Africa	3.3123	7.600104	61.098846	0.183541
141	Zimbabwe	Africa	3.2992	7.865712	55.617260	0.810237
142	South Sudan	Africa	2.8166	7.425360	51.000000	0.763417
143	Afghanistan	Asia	2.5669	7.462861	52.590000	0.933687

In [18]:
```
# Display bottom 5 countries for year 2019
bottom_5_2019
```

Out[18]:

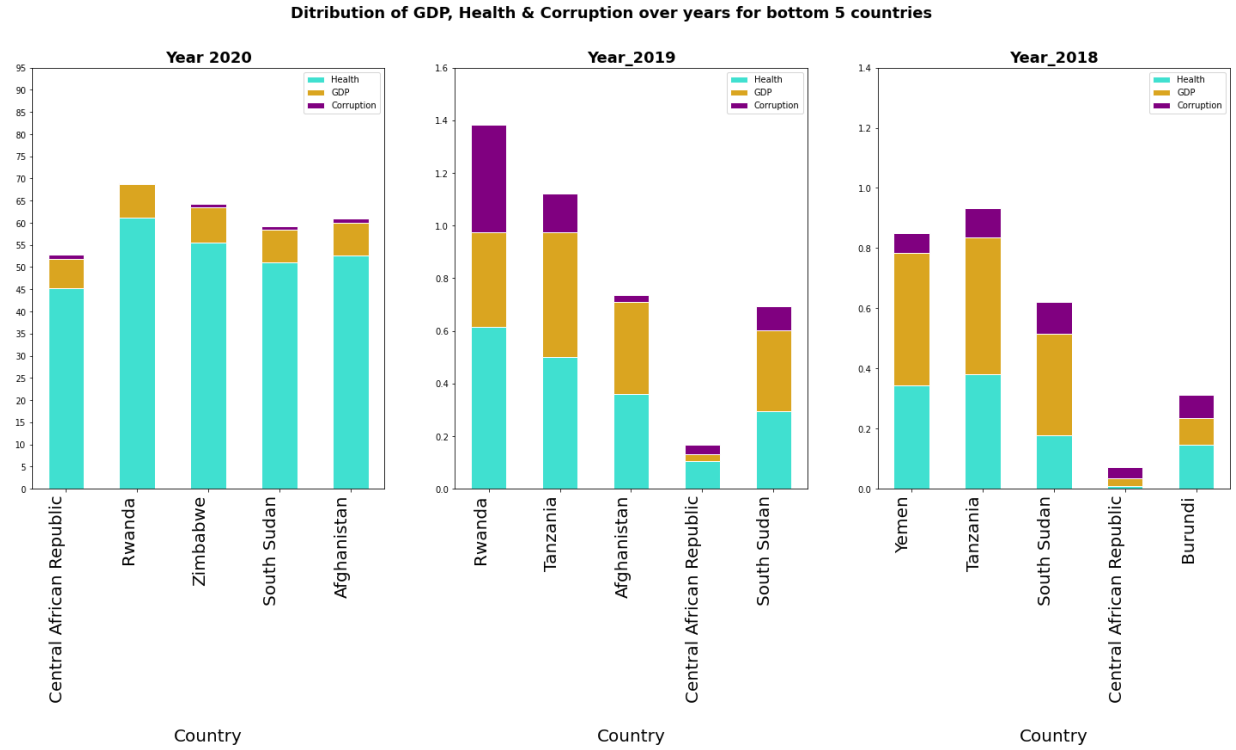
	Country	Region	Score_2019	GDP_per_Capita_2019	Health_2019	Corruption_2019
140	Rwanda	Africa	3.334	0.359	0.614	0.411
138	Tanzania	Africa	3.231	0.476	0.499	0.147
143	Afghanistan	Asia	3.203	0.350	0.361	0.025
139	Central African Republic	Africa	3.083	0.026	0.105	0.035
142	South Sudan	Africa	2.853	0.306	0.295	0.091

In [19]:
```
# Display bottom 5 countries for year 2018
bottom_5_2018
```

Out[19]:

	Country	Region	Score_2018	GDP_per_Capita_2018	Health_2018	Corruption_2018
136	Yemen	Africa	3.355	0.442	0.343	0.064
138	Tanzania	Africa	3.303	0.455	0.381	0.097
142	South Sudan	Africa	3.254	0.337	0.177	0.106
139	Central African Republic	Africa	3.083	0.024	0.010	0.038
130	Burundi	Africa	2.905	0.091	0.145	0.076

In [20]:
```
# Draw a figure with 1 row and 3 columns and set figure size
fig1,(ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(25,10))

# Set figure title
fig1.suptitle('Ditribution of GDP, Health & Corruption over years for bottom 5 co

# Call the above function year_stackBar to plot stacked bar graph of demographics

# Stacked bar graph for Year 2020
year_stackBar(ax1, "Year 2020", bottom_5_2020["Country"], bottom_5_2020["Health_2
              bottom_5_2020["Corruption_2020"], 100, 5)

# Stacked bar graph for Year 2019
year_stackBar(ax2, "Year_2019", bottom_5_2019["Country"], bottom_5_2019["Health_2
              bottom_5_2019["Corruption_2019"], 1.8, 0.2)

# Stacked bar graph for Year 2018
year_stackBar(ax3, "Year_2018", bottom_5_2018["Country"], bottom_5_2018["Health_2
              bottom_5_2018["Corruption_2018"], 1.5, 0.2)

# Rotate xticks at 90 degrees
fig1.autofmt_xdate(rotation=90)

# Save fig
fig1.savefig("Plot_Images/bottom5_stackedBar.png")

# Display Plot
plt.show()
```



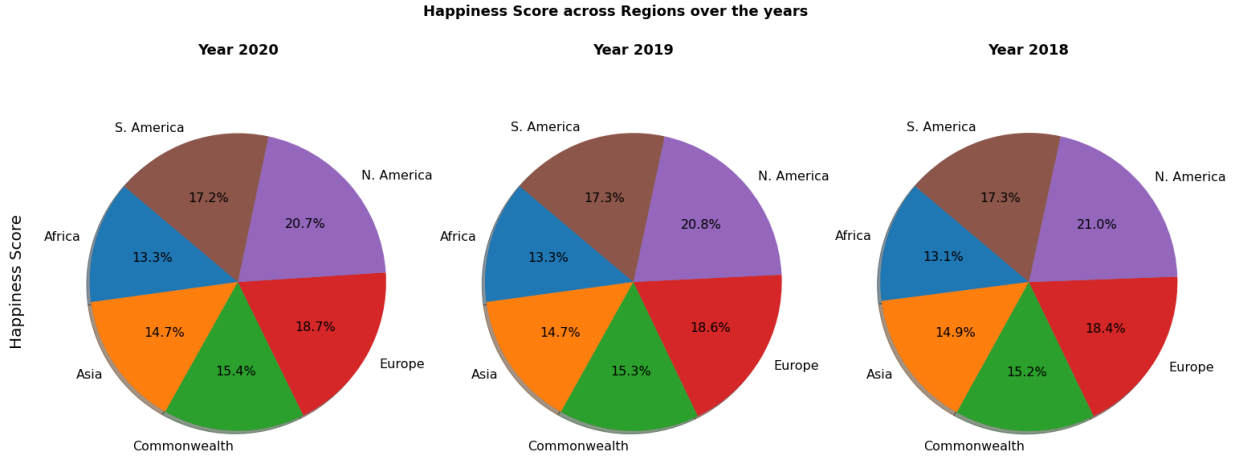
Question 3 : Which is the most and the least Happiest Region? Are there any outliers in terms of happiness score across region over the years?

Analysis:

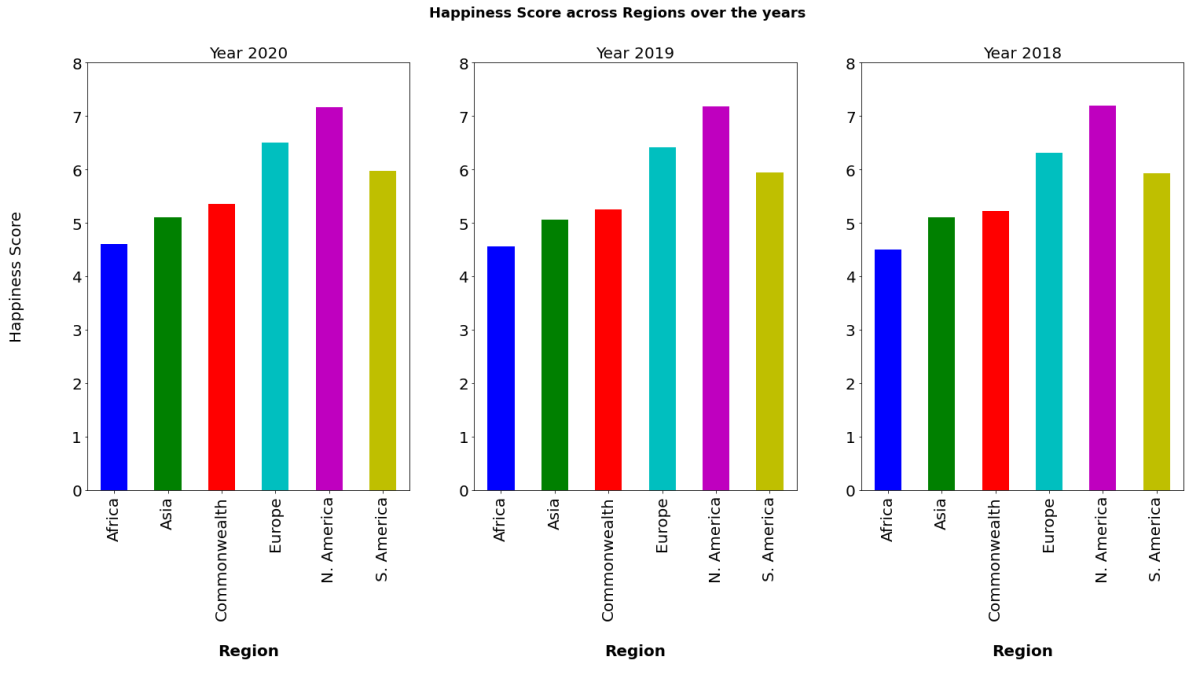
- Noth America is the consistently most happiest region over the years.
- Africa is consistently least happiest region over the years.
- Region wise there is not much difference in happiness score over last 3 years.
- Africa region has most outliers over the last 3 years
- Commonwealth (Coomonweath of Independent States) & Europe region has been consistent over the last 3 years with no outliers at all.

In [21]:
```
# Using Matplotlib

# Create a groupby object by region
region_stat = merged_data.groupby("Region")

# Get the happiness score by region for a particular year
region_score_2020 = region_stat["Score_2020"].mean()
region_score_2019 = region_stat["Score_2019"].mean()
region_score_2018 = region_stat["Score_2018"].mean()

# Get the regions name from index values
regions_2020 = list(region_score_2020.index.values)
regions_2019 = list(region_score_2019.index.values)
regions_2018 = list(region_score_2018.index.values)

# Method to plot pie chart for happiness score by region for particular year.
# Parameters are as below:
# ax = subplot
# values = Happiness_score
# label = pie_chart label, region names
# title = sub plot title

def region_pie(ax, values, label, title):

    #Plot the graph
    ax.pie(values, labels = label,  autopct="%1.1f%%", shadow=True, startangle=14

    # Set the font size, title & ylabel
    ax.set_title(title, pad=10, fontsize=18, fontweight='bold')
    ax.axis("equal")
```

In [22]:
```
# Create a figure with one row and three columns and set
fig1,(ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(25,10))

# Set the figure title
fig1.suptitle('Happiness Score across Regions over the years', fontsize=18, fontw

# Set the y-label
ax1.set_ylabel("Happiness Score", labelpad=60, fontsize=20)

# Call method region_pie to plot pie chart for Happiness Score across Regions ove

# Year 2020
region_pie(ax1, region_score_2020, regions_2020, "Year 2020")
# Year 2019
region_pie(ax2, region_score_2019, regions_2019, "Year 2019")
# Year 2018
region_pie(ax3, region_score_2018, regions_2018, "Year 2018")

# Save fig
fig1.savefig("Plot_Images/Score_ocross_region_pie.png")

# Show plot
plt.show()
```



In [23]:
```
# Using pandas.plot i.e df.plot

# Draw a figure with one row and 3 column, set figure size
fig1,(ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(25,10))

# Set figure titile
fig1.suptitle('Happiness Score across Regions over the years', fontsize=18, fontw

# groupby region and get the mean of happiness score for year 2020 to plot bar gr
merged_data.groupby("Region")["Score_2020"].mean().plot(ax=ax1, kind='bar', color
                                                        ylim=(0,8), title = "Year

# groupby region and get the mean of happiness score for year 2019 to plot bar gr
merged_data.groupby("Region")["Score_2019"].mean().plot(ax=ax2, kind='bar', color
                                                         ylim=(0,8), title = "Year

# groupby region and get the mean of happiness score for year 2018 to plot bar gr
merged_data.groupby("Region")["Score_2018"].mean().plot(ax=ax3, kind='bar', color
                                                        ylim=(0,8), title = "Year

# Set the ylabel
ax1.set_ylabel("Happiness Score", labelpad=60, fontsize=20)

# Set the size of the title for all sub-plots
ax1.title.set_size(20)
ax2.title.set_size(20)
ax3.title.set_size(20)

# Set x-label for all sub-plots
ax1.set_xlabel("Region", labelpad=30, fontsize=20, fontweight='bold')
ax2.set_xlabel("Region", labelpad=30, fontsize=20, fontweight='bold')
ax3.set_xlabel("Region", labelpad=30, fontsize=20, fontweight='bold')

# Save fig
fig1.savefig("Plot_Images/Score_ocross_region_Bar.png")

# Show plot
plt.show()
```



Happiness BoxPlots by Regions

In [24]:
```
regions = regions_2020

# Method to calculate quartiles, lower & upper bound for region specific happines
# Parameters as below:
# colname : column name for year specific happiness score
# Year : Year for which quartiles are being calculated

def cal_quartiles(colname, year):
    counter = 0 # counter is declared to track the first iteration

    #Iterate through list of regions
    for region in regions:

        # From mergerd_data df, if region name matches the one from list, add hap
        region_df = merged_data.loc[merged_data["Region"] == region, colname]

        # Get the quartiles, Q1, Q3 & IQR of happiness score for specific region
        # Calculate lowe & upper bound to get potential outliers
        quartiles = region_df.quantile([.25,.5,.75])
        lowerq = quartiles[0.25]
        upperq = quartiles[0.75]
        iqr = upperq - lowerq
        lower_bound = lowerq - (1.5 * iqr)
        upper_bound = upperq + (1.5 * iqr)

        # Print the data received. Counter is used to print the dotted line only
        if counter == 0:
            space = '                    '
            print(f"=========================================================\n{spa
            print("=========================================================")
            counter +=1
        print(f"The lower quartile of {region} Happiness Score is: {lowerq}")
        print(f"The upper quartile of {region} Happiness Score is: {upperq}")
        print(f"The interquartuile range of {region} Happiness Score is: {iqr}")
        print(f"The median of {region} Happiness Score is: {quartiles[0.5]}")
        print(f"Values below {lower_bound} could be outliers for {region} Happine
        print(f"Values above {upper_bound} could be outliers for {region} Happine
        print("---------------------------------------------------------")
```

In [25]:
```
# Call method cal_quartiles to calculate quartiles, lower & upper bound for regio
cal_quartiles("Score_2020", 2020)
```

```
=========================================================
                    Year 2020
=========================================================
The lower quartile of Africa Happiness Score is: 4.1620498894999995
The upper quartile of Africa Happiness Score is: 5.02504980525
The interquartuile range of Africa Happiness Score is: 0.8629999157500006
The median of Africa Happiness Score is: 4.6284999845
Values below 2.8675500158749987 could be outliers for Africa Happiness Score
Values above 6.319549678875001 could be outliers for Africa Happiness Score
---------------------------------------------------------
The lower quartile of Asia Happiness Score is: 4.8406000135000005
The upper quartile of Asia Happiness Score is: 5.782049894
The interquartuile range of Asia Happiness Score is: 0.9414498804999996
The median of Asia Happiness Score is: 5.285600185
Values below 3.428425192750001 could be outliers for Asia Happiness Score
Values above 7.19422471475 could be outliers for Asia Happiness Score
---------------------------------------------------------
The lower quartile of Commonwealth Happiness Score is: 5.008525013999999
The upper quartile of Commonwealth Happiness Score is: 5.56864988775
The interquartuile range of Commonwealth Happiness Score is: 0.5601248737500013
The median of Commonwealth Happiness Score is: 5.540699959
Values below 4.168337703374997 could be outliers for Commonwealth Happiness Sco
re
Values above 6.408837198375002 could be outliers for Commonwealth Happiness Sco
re
---------------------------------------------------------
The lower quartile of Europe Happiness Score is: 5.987800001749999
The upper quartile of Europe Happiness Score is: 7.182750225500001
The interquartuile range of Europe Happiness Score is: 1.194950223750002
The median of Europe Happiness Score is: 6.375400066000001
Values below 4.195374666124996 could be outliers for Europe Happiness Score
Values above 8.975175561125004 could be outliers for Europe Happiness Score
---------------------------------------------------------
The lower quartile of N. America Happiness Score is: 7.15199983125
The upper quartile of N. America Happiness Score is: 7.248975038499999
The interquartuile range of N. America Happiness Score is: 0.09697520724999897
The median of N. America Happiness Score is: 7.227449893999999
Values below 7.006537020375002 could be outliers for N. America Happiness Score
Values above 7.394437849374998 could be outliers for N. America Happiness Score
---------------------------------------------------------
The lower quartile of S. America Happiness Score is: 5.784475088500001
The upper quartile of S. America Happiness Score is: 6.3551249502500005
The interquartuile range of S. America Happiness Score is: 0.5706498617499998
The median of S. America Happiness Score is: 6.055900097
Values below 4.928500295875001 could be outliers for S. America Happiness Score
Values above 7.211099742875 could be outliers for S. America Happiness Score
---------------------------------------------------------
```

In [26]:
```
# Call method cal_quartiles to calculate quartiles, lower & upper bound for regio
cal_quartiles("Score_2019", 2019)
```

```
=========================================================
                    Year 2019
=========================================================
The lower quartile of Africa Happiness Score is: 4.101500000000001
The upper quartile of Africa Happiness Score is: 4.957000000000001
The interquartuile range of Africa Happiness Score is: 0.8555000000000001
The median of Africa Happiness Score is: 4.5215
Values below 2.8182500000000004 could be outliers for Africa Happiness Score
Values above 6.240250000000001 could be outliers for Africa Happiness Score
---------------------------------------------------------
The lower quartile of Asia Happiness Score is: 4.578
The upper quartile of Asia Happiness Score is: 5.6419999999999995
The interquartuile range of Asia Happiness Score is: 1.0639999999999992
The median of Asia Happiness Score is: 5.191
Values below 2.9820000000000015 could be outliers for Asia Happiness Score
Values above 7.237999999999998 could be outliers for Asia Happiness Score
---------------------------------------------------------
The lower quartile of Commonwealth Happiness Score is: 5.04575
The upper quartile of Commonwealth Happiness Score is: 5.55875
The interquartuile range of Commonwealth Happiness Score is: 0.5129999999999999
The median of Commonwealth Happiness Score is: 5.292
Values below 4.27625 could be outliers for Commonwealth Happiness Score
Values above 6.32825 could be outliers for Commonwealth Happiness Score
---------------------------------------------------------
The lower quartile of Europe Happiness Score is: 5.859249999999999
The upper quartile of Europe Happiness Score is: 7.063
The interquartuile range of Europe Happiness Score is: 1.2037500000000003
The median of Europe Happiness Score is: 6.2105
Values below 4.0536249999999985 could be outliers for Europe Happiness Score
Values above 8.868625 could be outliers for Europe Happiness Score
---------------------------------------------------------
The lower quartile of N. America Happiness Score is: 7.143999999999999
The upper quartile of N. America Happiness Score is: 7.28525
The interquartuile range of N. America Happiness Score is: 0.14125000000000032
The median of N. America Happiness Score is: 7.253
Values below 6.932124999999999 could be outliers for N. America Happiness Score
Values above 7.4971250000000005 could be outliers for N. America Happiness Scor
e
---------------------------------------------------------
The lower quartile of S. America Happiness Score is: 5.77
The upper quartile of S. America Happiness Score is: 6.30525
The interquartuile range of S. America Happiness Score is: 0.5352500000000004
The median of S. America Happiness Score is: 6.0955
Values below 4.967124999999999 could be outliers for S. America Happiness Score
Values above 7.108125000000001 could be outliers for S. America Happiness Score
---------------------------------------------------------
```

In [27]:
```
# Call method cal_quartiles to calculate quartiles, lower & upper bound for regio
cal_quartiles("Score_2018", 2018)
```

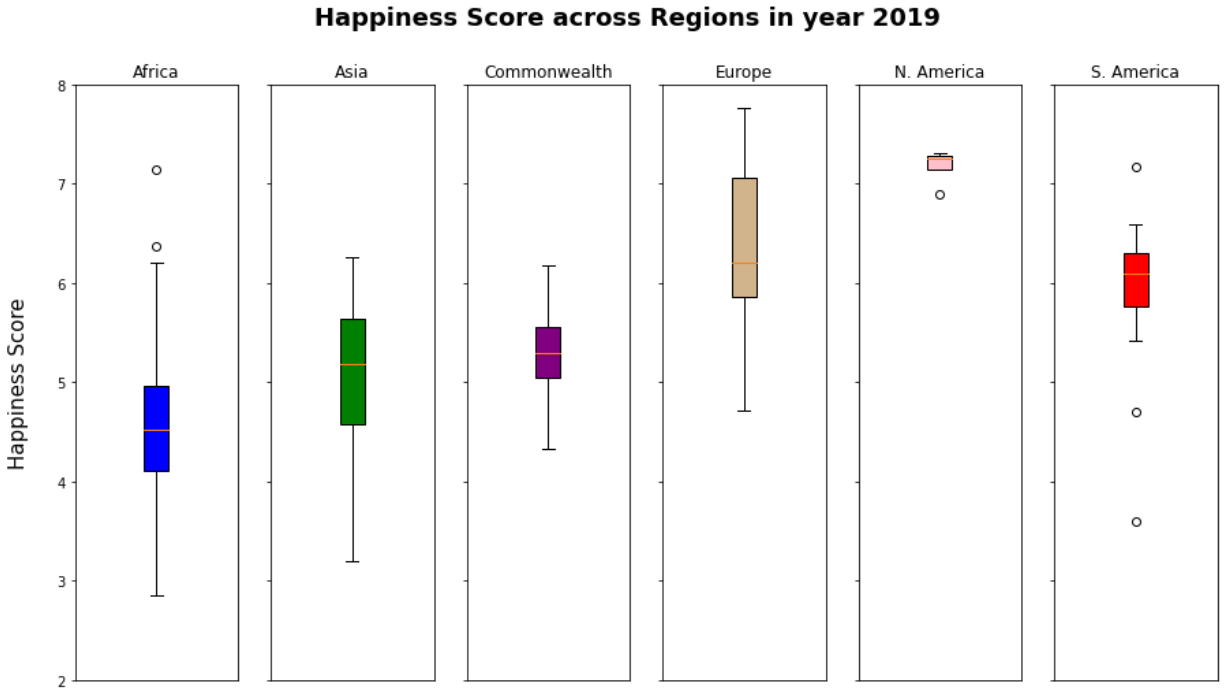
```
=========================================================
                    Year 2018
=========================================================
The lower quartile of Africa Happiness Score is: 3.99025
The upper quartile of Africa Happiness Score is: 4.812250000000001
The interquartuile range of Africa Happiness Score is: 0.8220000000000005
The median of Africa Happiness Score is: 4.432499999999999
Values below 2.757249999999999 could be outliers for Africa Happiness Score
Values above 6.045250000000001 could be outliers for Africa Happiness Score
---------------------------------------------------------
The lower quartile of Asia Happiness Score is: 4.4855
The upper quartile of Asia Happiness Score is: 5.6995000000000005
The interquartuile range of Asia Happiness Score is: 1.2140000000000004
The median of Asia Happiness Score is: 5.103
Values below 2.6644999999999994 could be outliers for Asia Happiness Score
Values above 7.520500000000001 could be outliers for Asia Happiness Score
---------------------------------------------------------
The lower quartile of Commonwealth Happiness Score is: 4.93325
The upper quartile of Commonwealth Happiness Score is: 5.677499999999999
The interquartuile range of Commonwealth Happiness Score is: 0.7442499999999992
The median of Commonwealth Happiness Score is: 5.3420000000000005
Values below 3.8168750000000014 could be outliers for Commonwealth Happiness Sc
ore
Values above 6.793874999999998 could be outliers for Commonwealth Happiness Sco
re
---------------------------------------------------------
The lower quartile of Europe Happiness Score is: 5.651500000000001
The upper quartile of Europe Happiness Score is: 7.0175
The interquartuile range of Europe Happiness Score is: 1.3659999999999988
The median of Europe Happiness Score is: 6.148
Values below 3.602500000000003 could be outliers for Europe Happiness Score
Values above 9.066499999999998 could be outliers for Europe Happiness Score
---------------------------------------------------------
The lower quartile of N. America Happiness Score is: 7.1755
The upper quartile of N. America Happiness Score is: 7.325
The interquartuile range of N. America Happiness Score is: 0.14949999999999974
The median of N. America Happiness Score is: 7.298
Values below 6.951250000000001 could be outliers for N. America Happiness Score
Values above 7.54925 could be outliers for N. America Happiness Score
---------------------------------------------------------
The lower quartile of S. America Happiness Score is: 5.6765
The upper quartile of S. America Happiness Score is: 6.3957500000000005
The interquartuile range of S. America Happiness Score is: 0.7192500000000006
The median of S. America Happiness Score is: 6.154
Values below 4.597624999999999 could be outliers for S. America Happiness Score
Values above 7.474625000000001 could be outliers for S. America Happiness Score
---------------------------------------------------------
```

In [28]:
```
# Methd to plot box plots for region wise happiness score over years
# Parameters are as below:
# colname : column name for year specific happiness score
# Year : Year for which quartiles are being calculated

def boxPlot_region(colname, year):

    # Create list of an empty lists which would contain the happiness score by re
    score_by_region = [[] for i in range(len(regions))]

    # region list created above is as below with their indices:
    # 0 - Africa, so add Happiness Scoree for this region at index 0 of list scor
    # 1 - Asia, so add Happiness Scoree for this region at index 1 of list score_
    # 2 - Commonwealth, so add Happiness Scoree for this region at index 2 of lis
    # 3 - Europe, so add Happiness Scoree for this region at index 3 of list scor
    # 4 - N. America, so add Happiness Scoree for this region at index 4 of list
    # 5 - S. America, so add Happiness Scoree for this region at index 5 of list

    # Iterate through a dataframe
    for i in range(len(merged_data)):

        # With iloc methd get the region & Happiness score from each row
        region = merged_data.iloc[i,:]["Region"]
        score = round(merged_data.iloc[i,:][colname],3)

        # Iterate through a list of regions with index
        for j in range(len(regions)):

            # If region from list matches with the region from data frame,
            # add score to list with same index inside a list
            if region == regions[j]:
                score_by_region[j].append(score)


    # Plot Boxplots:
    # To plot boxplots for all regions in same figure, declard axs as array with
    fig1,axs = plt.subplots(nrows=1, ncols=6, sharey=True, figsize=(15,8))

    #Set title for the figure
    fig1.suptitle(f'Happiness Score across Regions in year {year}', fontsize=18,

    # Set Y lable
    axs[0].set_ylabel("Happiness Score", labelpad=20, fontsize=15)
    axs[0].set_ylim(2,8)

    # Declare a list of colors
    colors = ['blue', 'green', 'purple', 'tan', 'pink', 'red']

    # Set the title, draw boxplots with region & Happiness Score list and set xti
    for i in range(len(regions)):
        axs[i].set_title(regions[i])
        box = axs[i].boxplot(score_by_region[i], patch_artist=True)
        axs[i].set_xticks([])
        plt.setp(box["boxes"], facecolor=colors[i])

    # Save fig
```

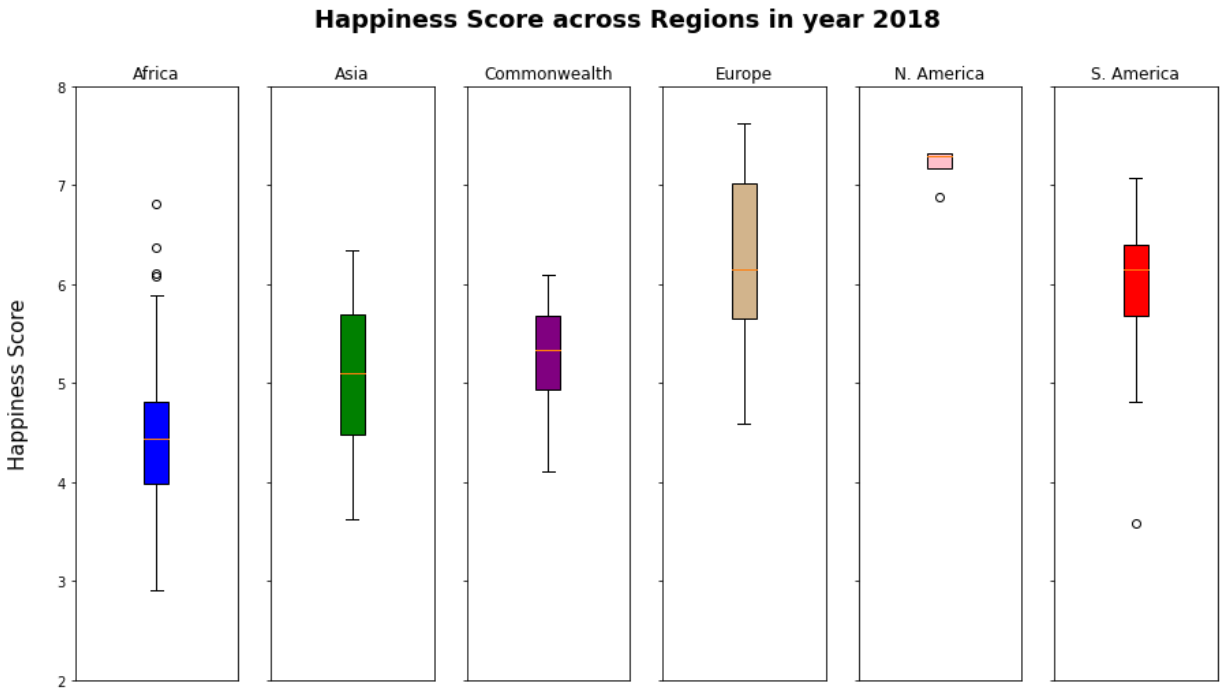
```
    fig1.savefig(f"Plot_Images/Region_Score_boxPlot_{year}.png")

    #Display plot
    plt.show()
```

In [29]:
```
# Call method boxPlot_region to plot for year 2020
boxPlot_region("Score_2020", 2020)
```

In [30]:
```
# Call method boxPlot_region to plot for year 2019
boxPlot_region("Score_2019", 2019)
```



In [31]:
```
# Call method boxPlot_region to plot for year 2018
boxPlot_region("Score_2018", 2018)
```



Question 4: Which country experiences a significant increase or decrease in happiness?

Analysis

- The country that experienced a significant increase in happiness is Benin, Africa. There is a moderate increase after calcualting the delta score is 1.075.
- Afghanistan has experienced a significant decline over the years furthersome, after calculating the delta score, the country is, scoring -1.0651.

In [32]:
```
# Creating a df for calculating the delta score over the years.
delta_data = pd.DataFrame(merged_data,columns = ['Country','Region','Score_2018',

delta_data['Delta Cal'] = (delta_data['Score_2019']-delta_data['Score_2018']) + (

# Sorting dataframe based on delta value

delta_data= delta_data.sort_values(by=['Delta Cal'], ascending=False)

# Top and bottom country according to Delta Score

Top_HappinessScore = delta_data.head(1)
Low_HappinessScore = delta_data.tail(1)

# Displaying updated dataframe
delta_data
```

Out[32]:

	Country	Region	Score_2018	Score_2019	Score_2020	Delta Cal
81	Benin	Africa	4.141	4.883	5.2160	1.0750
116	Liberia	Africa	3.495	3.975	4.5579	1.0629
95	Guinea	Africa	3.964	4.534	4.9493	0.9853
130	Burundi	Africa	2.905	3.775	3.7753	0.8703
96	Niger	Africa	4.166	4.628	4.9096	0.7436
...	...	...	...	...	...	...
134	India	Asia	4.190	4.015	3.5733	-0.6167
131	Zambia	Africa	4.377	4.107	3.7594	-0.6176
129	Sierra Leone	Africa	4.571	4.374	3.9264	-0.6446
77	Malaysia	Asia	6.322	5.339	5.3843	-0.9377
143	Afghanistan	Asia	3.632	3.203	2.5669	-1.0651

143 rows × 6 columns

In [33]:
```
# Top country with High Happiness Score
Top_HappinessScore
```

Out[33]:

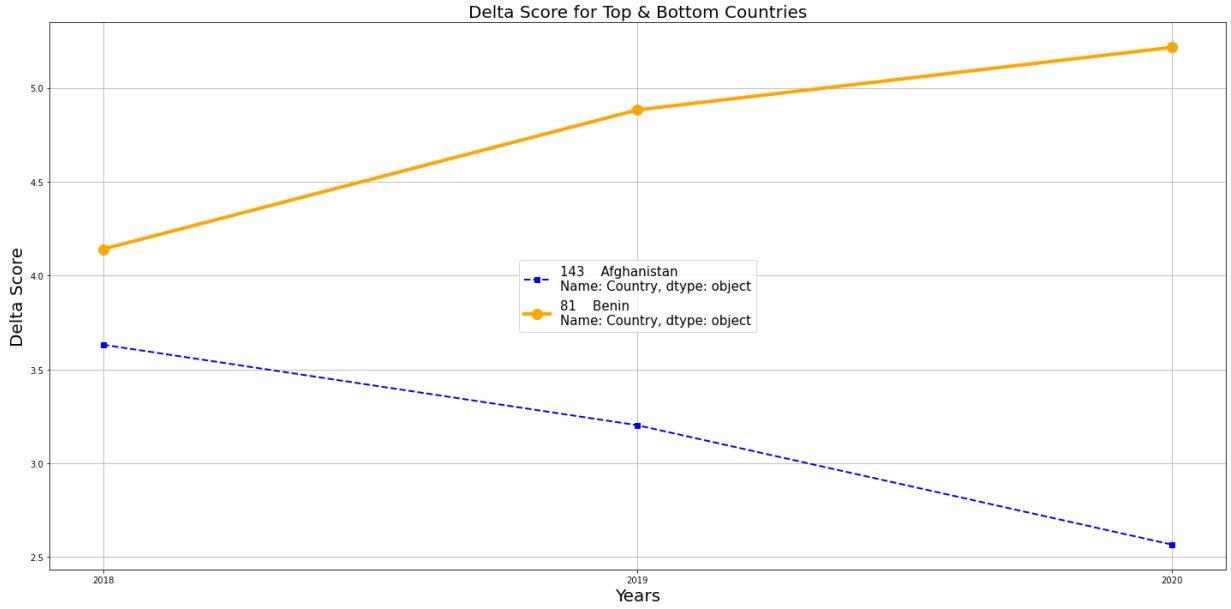
	Country	Region	Score_2018	Score_2019	Score_2020	Delta Cal
81	Benin	Africa	4.141	4.883	5.216	1.075

In [34]:
```
# Bottom country with lowest Happiness Score
Low_HappinessScore
```

Out[34]:

	Country	Region	Score_2018	Score_2019	Score_2020	Delta Cal
143	Afghanistan	Asia	3.632	3.203	2.5669	-1.0651

In [35]:
```
# Declaring variables for plotting the graph

year = ['2018','2019','2020']

# Getting top and bottom score for 2018, 2019, 2020
Top_Score = Top_HappinessScore.iloc[0,2:5]
Low_Score = Low_HappinessScore.iloc[0,2:5]

# Plot labels
top_label=Top_HappinessScore['Country']
bottom_label=Low_HappinessScore['Country']

#Adjusting the figure size
plt.figure(figsize=(20,10))

# Line plot
Low_Plot, = plt.plot(year, Low_Score, color="blue", marker ='s', linewidth=2, lir
Top_Plot, = plt.plot(year, Top_Score, color="orange", marker ='o', markersize=12,


#Setting tittle, label and legend
plt.title('Delta Score for Top & Bottom Countries', fontsize=20)
plt.xlabel('Years',fontsize=20)
plt.ylabel('Delta Score',fontsize=20)
plt.legend(loc="center",prop={"size":15})
plt.tight_layout()

# Saving plot image
plt.savefig("Plot_Images/Delta Score.png")
plt.grid()

# Displaying the plot
plt.show()
```



Question 5: What is the trend of happiness score in USA, over three years?

Analysis:

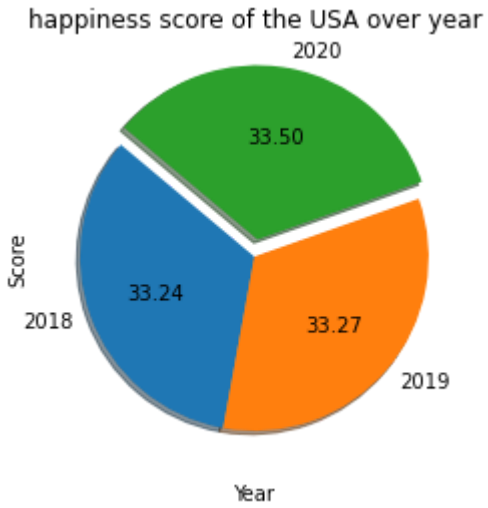
- The happiness score in US from year 2018 to 2019 has shown increasing trend.
- Rate of hapiness from 2019 to 2020 is far higher than the rate from 2018 to 2019d.

In [36]:
```
#select USA data and cleanup
score_year_df = merged_data[['Country','Score_2018','Score_2019','Score_2020']]
only_USA = score_year_df.loc[merged_data["Country"] == "United States", :]

# Deleteing country column
del only_USA['Country']

only_USA
```

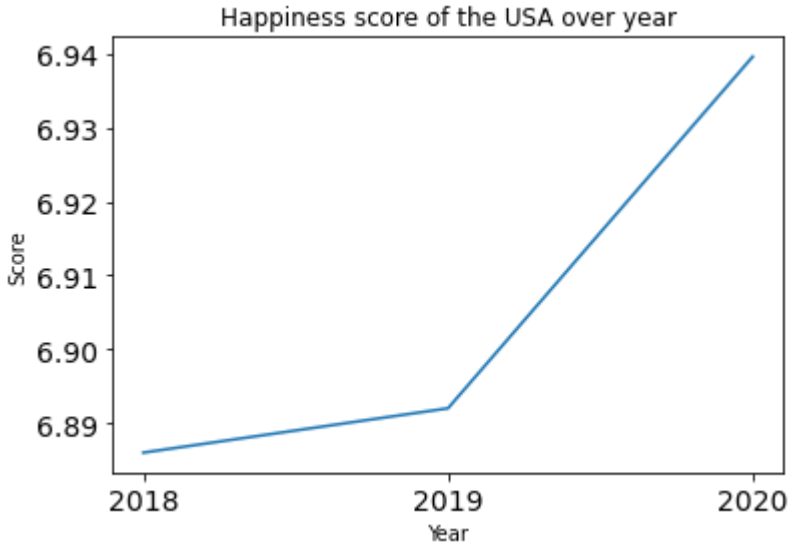
Out[36]:

	Score_2018	Score_2019	Score_2020
17	6.886	6.892	6.9396

In [37]:
```
# Declaring variables for plotting the graph
year = ["2018", "2019", "2020"]
score = only_USA.iloc[0,-3:]
explode=[0,0,0.1]

#Plotting pie plot
plt.pie(score, labels = year,autopct="%1.2f", shadow=True, startangle=140, explod

#Setting tittle and label
plt.xlabel("Year")
plt.ylabel("Score")
plt.title('happiness score of the USA over year')

# Saving plot image
plt.savefig("Plot_Images/happiness_only_US_pie.png")

# Displaying the plot
plt.show()
```



In [38]:
```
#Setting tittle and label
plt.xlabel("Year")
plt.ylabel("Score")
plt.title('Happiness score of the USA over year')
plt.tick_params(axis = 'both', which = 'major', labelsize=14)

#Line plot
plt.plot(year, score)

# Saving plot image
plt.savefig("Plot_Images/happiness_only_US_line.png")

# Displaying the plot
plt.show()
```



Question 6: Compare Happiness score to HDI.

Analysis:

- As per the r-value from the below graph, Happiness score and HDI have postive correlation.
- Note:HDI Score is avalible only for year 2018. Website is not updated for year 2019 and 2020.

In [39]:
```
# Base HDI URL
url = "http://ec2-54-174-131-205.compute-1.amazonaws.com/API/HDRO_API.php/"
indicator = "137506"
year = "2018"

# Build partial query URL
query_url = f"{url}indicator_id={indicator}/year={year}"

# Processing Get request
response = requests.get(query_url).json()

# Print Response from URL
pprint(response)
```

```
{'country_name': {'AFG': 'Afghanistan',
                  'AGO': 'Angola',
                  'ALB': 'Albania',
                  'AND': 'Andorra',
                  'ARE': 'United Arab Emirates',
                  'ARG': 'Argentina',
                  'ARM': 'Armenia',
                  'ATG': 'Antigua and Barbuda',
                  'AUS': 'Australia',
                  'AUT': 'Austria',
                  'AZE': 'Azerbaijan',
                  'BDI': 'Burundi',
                  'BEL': 'Belgium',
                  'BEN': 'Benin',
                  'BFA': 'Burkina Faso',
                  'BGD': 'Bangladesh',
                  'BGR': 'Bulgaria',
                  'BHR': 'Bahrain',
                  'BHS': 'Bahamas',
```

In [40]:
```
# Read thru the API response and get the HDI score for all the countries present

Sort_data_2018 = data_2018.copy()

coutry_dict = response['country_name']
HDI_dict = response['indicator_value']

# Adding new column- HDI Score
Sort_data_2018['HDI_Score'] = ''

# Iterate thru Sroted data, if contry names matches get the HDI score from teh AP
for index,row in Sort_data_2018.iterrows():
    for code, name in coutry_dict.items():
        if name  == row['Country'] :
            country_code = code
            HDI_score = response['indicator_value'][country_code]['137506']['2018
            Sort_data_2018.iloc[index,5] = HDI_score
            break

Sort_data_2018
```

Out[40]:

	Country	Score	GDP_per_Capita	Health	Corruption	HDI_Score
0	Finland	7.632	1.305	0.874	0.393	0.925
1	Norway	7.594	1.456	0.861	0.340	0.954
2	Denmark	7.555	1.351	0.868	0.408	0.93
3	Iceland	7.495	1.343	0.914	0.138	0.938
4	Switzerland	7.487	1.420	0.927	0.357	0.946
...	...	...	...	...	...	...
151	Yemen	3.355	0.442	0.343	0.064	0.463
152	Tanzania	3.303	0.455	0.381	0.097	
153	South Sudan	3.254	0.337	0.177	0.106	0.413
154	Central African Republic	3.083	0.024	0.010	0.038	0.381
155	Burundi	2.905	0.091	0.145	0.076	0.423

156 rows × 6 columns

In [41]:
```
# Drop the rows that have the null values for the HDI score

HDI_list = []
HDI_list = Sort_data_2018["HDI_Score"]

index_names = Sort_data_2018[(Sort_data_2018['HDI_Score'] == "")].index

Update_data_2018 = Sort_data_2018.drop(index_names)
Update_data_2018
```

Out[41]:

	Country	Score	GDP_per_Capita	Health	Corruption	HDI_Score
0	Finland	7.632	1.305	0.874	0.393	0.925
1	Norway	7.594	1.456	0.861	0.340	0.954
2	Denmark	7.555	1.351	0.868	0.408	0.93
3	Iceland	7.495	1.343	0.914	0.138	0.938
4	Switzerland	7.487	1.420	0.927	0.357	0.946
...	...	...	...	...	...	...
150	Rwanda	3.408	0.332	0.400	0.444	0.536
151	Yemen	3.355	0.442	0.343	0.064	0.463
153	South Sudan	3.254	0.337	0.177	0.106	0.413
154	Central African Republic	3.083	0.024	0.010	0.038	0.381
155	Burundi	2.905	0.091	0.145	0.076	0.423

134 rows × 6 columns

In [42]:
```
# Plot Happiness score to HDI for 2018 and get the correaltion between them.

x_values = Update_data_2018['Score']
y_values = Update_data_2018['HDI_Score'].astype(float)

# Get the linear equation
(slope, intercept, rvalue, pvalue, stderr) = linregress(x_values, y_values)

regress_values = x_values * slope + intercept

line_eq = "y = " + str(round(slope,2)) + "x + " + str(round(intercept,2))

# Plot the scatter plot
plt.scatter(x_values,y_values)
plt.plot(x_values,regress_values,"r-")


# Annotate the line_equation on graph
plt.annotate(line_eq,(5,0.4),fontsize=15,color="red")

# Set label for x & y axis and xlim
plt.title(f"Happiness Score vs. HDI Score")
plt.xlabel('Hapiness Score')
plt.ylabel('HDI Score')
plt.xlim(2,8)
plt.ylim(0.3,1)
plt.grid()

# Print r2-value
print(f"The r-squared is: {rvalue**2}")

# Saving plot image
plt.savefig(os.path.join('Plot_Images',f'Happiness Score vs. HDI Score.png'))

# Displaying the plot
plt.show()
```

```
The r-squared is: 0.7025275089527554
```