PISA 2012 Data Analysis

Prepared by Donald Ghazi

Dataset Description

PISA (Programme for International Student Assessment)

"PISA is a survey of students' skills and knowledge as they approach the end of compulsory education. It is not a conventional school test. Rather than examining how well students have learned the school curriculum, it looks at how well prepared they are for life beyond school."

"Around 510,000 students in 65 economies took part in the PISA 2012 assessment of reading, mathematics and science representing about 28 million 15-year-olds globally. Of those economies, 44 took part in an assessment of creative problem solving and 18 in an assessment of financial literacy."

Source: https://docs.google.com/document/d/e/2PACX-1vQmkX4iOT6Rcrin42vslquX2_wQCjIa_hbwD0xmxrERPSOJYDtpNc_3wwK_p9_KpOsfA6QVyEHdxxq7/pub?embedded=True

For this project, I was really interested to see what kind of variables I will be working with because I was informed that the file size is rather large and there's an extensive list of dictionary list that wasn't explained well prior to embarking on this project. Furthermore, as this is an official survey study that was conducted by a research institute, I knew that it will be challenging, but if I my dataframe was clean and tidy, I can be as creative as I want in my insights and visualization section.

As always, we want to gather out datasource and this time. In the following step, I'm running Jupyter Notebook from my our server and uploaded the files that I donwloaded from the Udacity's server. It did take a bit since the files are rather large.

Gathering Data

In [1]:
```
# import packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:
```
# import PISA 2012 data
pisa_2012 = pd.read_csv("/Users/donaldghazi/Desktop/pisa2012.csv",encoding='latin1',low_me
```

In [3]:
```
# import PISA 2012 Dictionary data
pisa_dict_2012 = pd.read_csv("/Users/donaldghazi/Desktop/pisadict2012.csv",encoding='latir
```

Assessing Data

Looking at the first few rows of the dataframe is always a good start, but I wanted to get a bigger picture of the dataset and try to understand what these variables may mean and how I would like to organize them prior to cleaning.

In [4]:
```
pisa_2012.sample(20)
```

Out[4]:

	Unnamed: 0	CNT	SUBNATIO	STRATUM	OECD	NC	SCHOOLID	STIDSTD	ST01Q01	ST
199688	199689	United Kingdom	8260000	GBR1105	OECD	United Kingdom (excl.Scotland)	424	10572	11	
177231	177232	Finland	2460000	FIN0003	OECD	Finland	56	1557	9	
169839	169840	Spain	7240900	ESP0918	OECD	Spain	866	24257	10	
464250	464251	Tunisia	7880000	TUN0013	Non-OECD	Tunisia	115	3268	10	
424540	424541	Singapore	7020000	SGP0201	Non-OECD	Singapore	54	1765	10	
250436	250437	Italy	3800000	ITA1902	OECD	Italy	380	9847	10	
401561	401562	China-Shanghai	1560000	QCN0002	Non-OECD	China (Shanghai)	40	1345	10	
26394	26395	Australia	360000	AUS0205	OECD	Australia	229	4244	9	
254233	254234	Italy	3800000	ITA1501	OECD	Italy	518	13644	10	
246809	246810	Italy	3800000	ITA1902	OECD	Italy	240	6220	9	
382385	382386	Poland	6160000	POL0001	OECD	Poland	140	3464	9	
318838	318839	Mexico	4840000	MEX0410	OECD	Mexico	134	3136	10	
13865	13866	United Arab Emirates	7840000	ARE0769	Non-OECD	United Arab Emirates	362	9123	10	
14884	14885	United Arab Emirates	7840100	ARE0101	Non-OECD	United Arab Emirates	402	10142	8	
14395	14396	United Arab Emirates	7840100	ARE0108	Non-OECD	United Arab Emirates	383	9653	10	
393135	393136	Qatar	6340000	QAT0004	Non-OECD	Qatar	47	3885	10	
47940	47941	Belgium	560100	BEL0111	OECD	Belgium	220	6554	9	
351214	351215	Montenegro	4990000	MNE0006	Non-OECD	Montenegro	19	1706	10	
484805	484806	Vietnam	7040000	VNM0208	Non-OECD	Viet Nam	141	4275	10	
162993	162994	Spain	7240700	ESP0713	OECD	Spain	616	17411	10	

20 rows × 636 columns

The pisa_2012 data does look pretty clean but there's a lot of columns that we can't really see. Further, I don't know what they really meant. This warned me that I should look at the csv file separately by running an IDLE and also read from the PISA 2012 booklet that can be found online.

In [5]:
```
# inspect df
pisa_2012.shape[0]
```

Out[5]:
```
485490
```

In [6]:
```
pisa_2012.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 485490 entries, 0 to 485489
Columns: 636 entries, Unnamed: 0 to VER_STU
dtypes: float64(250), int64(18), object(368)
memory usage: 2.3+ GB
```

- The total number of students is 485,490.
- There's 636 columns so we only want to choose columns that we really need.

To reiterate, we want to only keep the variables we find interesting and will help us gain best insights that are creative and fulfilling for us at the end. Knowing that PISA tests on Math, Reading, and Science, I was more interested in Reading Scores and Language-related variables.

In [7]:
```
# now inspect the column descriptions
pisa_dict_2012
```

Out[7]:

	Unnamed: 0	x
0	CNT	Country code 3-character
1	SUBNATIO	Adjudicated sub-region code 7-digit code (3-di...
2	STRATUM	Stratum ID 7-character (cnt + region ID + orig...
3	OECD	OECD country
4	NC	National Centre 6-digit Code
...	...	...
630	W_FSTR80	FINAL STUDENT REPLICATE BRR-FAY WEIGHT80
631	WVARSTRR	RANDOMIZED FINAL VARIANCE STRATUM (1-80)
632	VAR_UNIT	RANDOMLY ASSIGNED VARIANCE UNIT
633	SENWGT_STU	Senate weight - sum of weight within the count...
634	VER_STU	Date of the database creation

635 rows × 2 columns

In [8]:
```
pisa_dict_2012.head(10)
```

Out[8]:

	Unnamed: 0	x
0	CNT	Country code 3-character
1	SUBNATIO	Adjudicated sub-region code 7-digit code (3-di...
2	STRATUM	Stratum ID 7-character (cnt + region ID + orig...
3	OECD	OECD country
4	NC	National Centre 6-digit Code
5	SCHOOLID	School ID 7-digit (region ID + stratum ID + 3-...
6	STIDSTD	Student ID
7	ST01Q01	International Grade
8	ST02Q01	National Study Programme
9	ST03Q01	Birth - Month

- There's a lot of variables that went into the survey.
- I used Atom to open up the CSV file to read the dictionary in detail

I read the descriptions of the variables carefully and tried to understand how some of them were measured. Although it isn't super clear as to how some of the variables were derived, I was preparing myself for the cleaning portion as it is the longest, hardest, but the most important portion.

In [9]:
```
# CNT = Country Code
# NC  = National Centre Code
# use groupby based on 'NC' then within each 'NC', we group based on 'CNT'
# then count and sort values in decreaing amount
pisa_2012.groupby('NC')['CNT'].count().sort_values(ascending=False)
```

Out[9]:
```
NC
Mexico                        33806
Italy                         31073
Spain                         25313
Canada                        21544
Brazil                        19204
                              ...
New Zealand                    4291
Iceland                        3508
United Kingdom (Scotland)      2945
Perm (Russian Federation)      1761
Liechtenstein                   293
Name: CNT, Length: 66, dtype: int64
```

- We see that the country with the highest amount of participants was Mexico (33806) while Liechtenstein had the least amount (293).

Further, these are the columns from the dictionary list I find interesting and want to focus my project on.

- "AGE", "Age
- "CNT","Country code 3-character"
- "ST04Q01","Gender"
- "ST26Q12","Possessions - dictionary"
- "ST25Q01","International Language at Home"
- "TCHBEHFA","Teacher Behaviour: Formative Assessment"
- "TCHBEHSO","Teacher Behaviour: Student Orientation"
- "TCHBEHTD","Teacher Behaviour: Teacher-directed Instruction"
- "PV1MATH","Plausible value 1 in mathematics"
- "PV2MATH","Plausible value 2 in mathematics"
- "PV3MATH","Plausible value 3 in mathematics"
- "PV4MATH","Plausible value 4 in mathematics"
- "PV5MATH","Plausible value 5 in mathematics"
- "PV1READ","Plausible value 1 in reading"
- "PV2READ","Plausible value 2 in reading"
- "PV3READ","Plausible value 3 in reading"
- "PV4READ","Plausible value 4 in reading"
- "PV5READ","Plausible value 5 in reading"
- "PV1SCIE","Plausible value 1 in science"
- "PV2SCIE","Plausible value 2 in science"
- "PV3SCIE","Plausible value 3 in science"
- "PV4SCIE","Plausible value 4 in science"

- "PV5SCIE","Plausible value 5 in science"

Brainstorming Questions....

- *How did students perform in average?*
- *Which gender performed better in Reading?*
- *Did students with dictionaries perform better in Reading than those that didn't?*
- *Do students who possess dictionaries, alsmo more likely to possess literature books?*
- *What's the correlation between the three Teacher Behaviour scores?*
- *What's the correlation between the three subject tests? Is there a greater correlation between Math and Science scores?*

Cleaning Data

We know that there's 636 columns and we want to only keep the ones that will help us answer our questions.

In [35]:
```
# make a copy of the original df
pisa_2012_clean = pisa_2012.copy()
```

In [36]:
```
# keep the columns that we will need for our analysis
pisa_2012_clean = pisa_2012_clean[['CNT','ST04Q01','ST26Q12','AGE','ST26Q07','ST25Q01','TC
                                   'PV2MATH', 'PV3MATH', 'PV4MATH', 'PV5MATH', 'PV1READ',
                                   'PV4READ', 'PV5READ','PV1SCIE', 'PV2SCIE', 'PV3SCIE', '
```

In [37]:
```
# inspect
pisa_2012_clean
```

Out[37]:

	CNT	ST04Q01	ST26Q12	AGE	ST26Q07	ST25Q01	TCHBEHFA	TCHBEHSO	TCHBEHTD	PV1MATH
0	Albania	Female	Yes	16.17	No	Language of the test	1.3625	0.9374	0.4297	406.8469
1	Albania	Female	Yes	16.17	Yes	Language of the test	NaN	NaN	NaN	486.1427
2	Albania	Female	Yes	15.58	Yes	Language of the test	NaN	NaN	NaN	533.2684
3	Albania	Female	Yes	15.67	Yes	Language of the test	0.7644	3.3108	2.3916	412.2215
4	Albania	Female	Yes	15.50	Yes	Language of the test	0.7644	0.9374	0.4297	381.9209
...	...	...	...	...	...	...	...	...	...	...
485485	Vietnam	Female	Yes	15.83	No	Language of the test	NaN	NaN	NaN	477.1849
485486	Vietnam	Male	Yes	16.17	Yes	Language of the test	-0.2859	-0.1057	0.4297	518.9360
485487	Vietnam	Male	Yes	15.83	No	Language of the test	-0.9632	-0.1057	-0.5612	475.2376
485488	Vietnam	Male	Yes	15.83	No	Language of the test	-0.2859	0.2217	0.4297	550.9503

	CNT	ST04Q01	ST26Q12	AGE	ST26Q07	ST25Q01	TCHBEHFA	TCHBEHSO	TCHBEHTD	PV1MATH
485489	Vietnam	Female	Yes	15.33	Yes	Language of the test	NaN	NaN	NaN	470.0187

485490 rows × 24 columns

In [38]:
```
# doublecheck
pisa_2012_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 485490 entries, 0 to 485489
Data columns (total 24 columns):
 #   Column    Non-Null Count   Dtype  
---  ------    --------------   -----  
 0   CNT       485490 non-null  object 
 1   ST04Q01   485490 non-null  object 
 2   ST26Q12   474039 non-null  object 
 3   AGE       485374 non-null  float64
 4   ST26Q07   465860 non-null  object 
 5   ST25Q01   465496 non-null  object 
 6   TCHBEHFA  314678 non-null  float64
 7   TCHBEHSO  315114 non-null  float64
 8   TCHBEHTD  315519 non-null  float64
 9   PV1MATH   485490 non-null  float64
 10  PV2MATH   485490 non-null  float64
 11  PV3MATH   485490 non-null  float64
 12  PV4MATH   485490 non-null  float64
 13  PV5MATH   485490 non-null  float64
 14  PV1READ   485490 non-null  float64
 15  PV2READ   485490 non-null  float64
 16  PV3READ   485490 non-null  float64
 17  PV4READ   485490 non-null  float64
 18  PV5READ   485490 non-null  float64
 19  PV1SCIE   485490 non-null  float64
 20  PV2SCIE   485490 non-null  float64
 21  PV3SCIE   485490 non-null  float64
 22  PV4SCIE   485490 non-null  float64
 23  PV5SCIE   485490 non-null  float64
dtypes: float64(19), object(5)
memory usage: 88.9+ MB
```

I'm looking at the column above and my goal is to decrease the number of columns.

Let's first replace the missing values of in AGE column with the average.

In [39]:
```
#https://docs.scipy.org/doc/numpy-1.13.0/reference/generated/numpy.isfinite.html
pisa_2012_clean.loc[np.isfinite(pisa_2012_clean['AGE']) == False, 'AGE'] = pisa_2012_clear
pisa_2012_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 485490 entries, 0 to 485489
Data columns (total 24 columns):
 #   Column    Non-Null Count   Dtype  
---  ------    --------------   -----  
 0   CNT       485490 non-null  object 
 1   ST04Q01   485490 non-null  object 
 2   ST26Q12   474039 non-null  object 
 3   AGE       485490 non-null  float64
 4   ST26Q07   465860 non-null  object 
 5   ST25Q01   465496 non-null  object 
 6   TCHBEHFA  314678 non-null  float64
 7   TCHBEHSO  315114 non-null  float64
```

```
 8   TCHBEHTD  315519 non-null  float64
 9   PV1MATH   485490 non-null  float64
 10  PV2MATH   485490 non-null  float64
 11  PV3MATH   485490 non-null  float64
 12  PV4MATH   485490 non-null  float64
 13  PV5MATH   485490 non-null  float64
 14  PV1READ   485490 non-null  float64
 15  PV2READ   485490 non-null  float64
 16  PV3READ   485490 non-null  float64
 17  PV4READ   485490 non-null  float64
 18  PV5READ   485490 non-null  float64
 19  PV1SCIE   485490 non-null  float64
 20  PV2SCIE   485490 non-null  float64
 21  PV3SCIE   485490 non-null  float64
 22  PV4SCIE   485490 non-null  float64
 23  PV5SCIE   485490 non-null  float64
dtypes: float64(19), object(5)
memory usage: 88.9+ MB
```

Now, let's do the same for the three Teacher Behaviors.

In [40]:

```
# https://docs.scipy.org/doc/numpy-1.13.0/reference/generated/numpy.isfinite.html
# repeat the same process above for each teacher behavior
pisa_2012_clean.loc[np.isfinite(pisa_2012_clean['TCHBEHFA']) == False, 'TCHBEHFA'] = pisa_
pisa_2012_clean.loc[np.isfinite(pisa_2012_clean['TCHBEHSO']) == False, 'TCHBEHSO'] = pisa_
pisa_2012_clean.loc[np.isfinite(pisa_2012_clean['TCHBEHTD']) == False, 'TCHBEHTD'] = pisa_
pisa_2012_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 485490 entries, 0 to 485489
Data columns (total 24 columns):
 #   Column    Non-Null Count   Dtype
---  ------    --------------   -----
 0   CNT       485490 non-null  object
 1   ST04Q01   485490 non-null  object
 2   ST26Q12   474039 non-null  object
 3   AGE       485490 non-null  float64
 4   ST26Q07   465860 non-null  object
 5   ST25Q01   465496 non-null  object
 6   TCHBEHFA  485490 non-null  float64
 7   TCHBEHSO  485490 non-null  float64
 8   TCHBEHTD  485490 non-null  float64
 9   PV1MATH   485490 non-null  float64
 10  PV2MATH   485490 non-null  float64
 11  PV3MATH   485490 non-null  float64
 12  PV4MATH   485490 non-null  float64
 13  PV5MATH   485490 non-null  float64
 14  PV1READ   485490 non-null  float64
 15  PV2READ   485490 non-null  float64
 16  PV3READ   485490 non-null  float64
 17  PV4READ   485490 non-null  float64
 18  PV5READ   485490 non-null  float64
 19  PV1SCIE   485490 non-null  float64
 20  PV2SCIE   485490 non-null  float64
 21  PV3SCIE   485490 non-null  float64
 22  PV4SCIE   485490 non-null  float64
 23  PV5SCIE   485490 non-null  float64
dtypes: float64(19), object(5)
memory usage: 88.9+ MB
```

Now we have all the missing values filled in, we can organize them. Let's first look at our plausible values of each subject. We can create a separate column for each subject (Math, Reading, and Science) and each column will contain the mean value.

In [41]:
```
# we have 5 plausible values per subject, so add them in their own respective subject and
# store each results in its own respective subject column
pisa_2012_clean['Math Score'] = (pisa_2012_clean['PV1MATH'] + pisa_2012_clean['PV2MATH'] +
pisa_2012_clean['Reading Score'] = (pisa_2012_clean['PV1READ'] + pisa_2012_clean['PV2READ'
pisa_2012_clean['Science Score'] = (pisa_2012_clean['PV1SCIE'] + pisa_2012_clean['PV2SCIE'
```

In [42]:
```
pisa_2012_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 485490 entries, 0 to 485489
Data columns (total 27 columns):
 #   Column         Non-Null Count   Dtype
---  ------         --------------   -----
 0   CNT            485490 non-null  object
 1   ST04Q01        485490 non-null  object
 2   ST26Q12        474039 non-null  object
 3   AGE            485490 non-null  float64
 4   ST26Q07        465860 non-null  object
 5   ST25Q01        465496 non-null  object
 6   TCHBEHFA       485490 non-null  float64
 7   TCHBEHSO       485490 non-null  float64
 8   TCHBEHTD       485490 non-null  float64
 9   PV1MATH        485490 non-null  float64
 10  PV2MATH        485490 non-null  float64
 11  PV3MATH        485490 non-null  float64
 12  PV4MATH        485490 non-null  float64
 13  PV5MATH        485490 non-null  float64
 14  PV1READ        485490 non-null  float64
 15  PV2READ        485490 non-null  float64
 16  PV3READ        485490 non-null  float64
 17  PV4READ        485490 non-null  float64
 18  PV5READ        485490 non-null  float64
 19  PV1SCIE        485490 non-null  float64
 20  PV2SCIE        485490 non-null  float64
 21  PV3SCIE        485490 non-null  float64
 22  PV4SCIE        485490 non-null  float64
 23  PV5SCIE        485490 non-null  float64
 24  Math Score     485490 non-null  float64
 25  Reading Score  485490 non-null  float64
 26  Science Score  485490 non-null  float64
dtypes: float64(22), object(5)
memory usage: 100.0+ MB
```

In [43]:
```
# now we can drop the columns
pisa_2012_clean.drop(pisa_2012_clean.iloc[:, 9:24], inplace = True, axis = 1)
pisa_2012_clean
```

Out[43]:

	CNT	ST04Q01	ST26Q12	AGE	ST26Q07	ST25Q01	TCHBEHFA	TCHBEHSO	TCHBEHTD	Math Score
0	Albania	Female	Yes	16.17	No	Language of the test	1.36250	0.937400	0.429700	366.18634
1	Albania	Female	Yes	16.17	Yes	Language of the test	0.13793	0.209052	0.147423	470.56396
2	Albania	Female	Yes	15.58	Yes	Language of the test	0.13793	0.209052	0.147423	505.53824
3	Albania	Female	Yes	15.67	Yes	Language of the test	0.76440	3.310800	2.391600	449.45476
4	Albania	Female	Yes	15.50	Yes	Language of the test	0.76440	0.937400	0.429700	385.50398
...	...	...	...	...	...	...	...	...	...	...

	CNT	ST04Q01	ST26Q12	AGE	ST26Q07	ST25Q01	TCHBEHFA	TCHBEHSO	TCHBEHTD	Math Score
485485	Vietnam	Female	Yes	15.83	No	Language of the test	0.13793	0.209052	0.147423	486.22058
485486	Vietnam	Male	Yes	16.17	Yes	Language of the test	-0.28590	-0.105700	0.429700	529.21794
485487	Vietnam	Male	Yes	15.83	No	Language of the test	-0.96320	-0.105700	-0.561200	486.29850
485488	Vietnam	Male	Yes	15.83	No	Language of the test	-0.28590	0.221700	0.429700	522.90856
485489	Vietnam	Female	Yes	15.33	Yes	Language of the test	0.13793	0.209052	0.147423	454.43994

485490 rows × 12 columns

In [44]:
```
# double check
pisa_2012_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 485490 entries, 0 to 485489
Data columns (total 12 columns):
 #   Column         Non-Null Count   Dtype  
---  ------         --------------   -----  
 0   CNT            485490 non-null  object 
 1   ST04Q01        485490 non-null  object 
 2   ST26Q12        474039 non-null  object 
 3   AGE            485490 non-null  float64
 4   ST26Q07        465860 non-null  object 
 5   ST25Q01        465496 non-null  object 
 6   TCHBEHFA       485490 non-null  float64
 7   TCHBEHSO       485490 non-null  float64
 8   TCHBEHTD       485490 non-null  float64
 9   Math Score     485490 non-null  float64
 10  Reading Score  485490 non-null  float64
 11  Science Score  485490 non-null  float64
dtypes: float64(7), object(5)
memory usage: 44.4+ MB
```

Getting a long. Now, although the three Teacher Behavior evaluations/scores may seem like they can be grouped but that's not the case. This is because they aren't plausible values like the test subjects values we just cleaned up. These three Teacher Behavior scores measured a specific teaching style which will be explained further.

Let's fill in the missing values for the three columns below as unkown

In [45]:
```
# replace all NaN values for Dictionary as NA
pisa_2012_clean.loc[pisa_2012_clean['ST26Q12'].isna() == True,'ST26Q12'] = 'NA'
```

In [46]:
```
# replace all NaN values for Literature as NA
pisa_2012_clean.loc[pisa_2012_clean['ST26Q07'].isna() == True,'ST26Q07'] = 'NA'
```

In [47]:
```
# replace all Nan values for International Language at Home as NA
pisa_2012_clean.loc[pisa_2012_clean['ST25Q01'].isna() == True,'ST25Q01'] = 'NA'
```

We can change the default variable names for the sake of the project.

In [48]:
```
# https://www.oecd.org/pisa/pisaproducts/PISA%202012%20Technical%20Report_Chapter%2016.pdf
```

```
# rename the column names
pisa_2012_clean.rename({'CNT':'Country',
                        'AGE':'Age',
                        'ST04Q01':'Gender',
                        'ST26Q12': 'Dictionary',
                        'ST26Q07': 'Literature',
                        'ST25Q01': 'Test Language', # IT SHOWS IF THE STUDENT TOOK THE TES
                        'TCHBEHFA':'Formative Assessment',
                        'TCHBEHSO' :'Student Orientation',
                        'TCHBEHTD' : 'Teacher-Directed Instruction'}, axis = 'columns', in
```

In [49]:

```
# check
pisa_2012_clean.sample(10)
```

Out[49]:

	Country	Gender	Dictionary	Age	Literature	Test Language	Formative Assessment	Student Orientation	Teacher-Directed Instruction	M Sc
204007	Greece	Female	Yes	15.33	Yes	Language of the test	0.13793	0.209052	0.147423	438.54
163094	Spain	Female	Yes	15.58	Yes	Other language	-0.28590	0.485500	-0.808300	509.35
191421	United Kingdom	Male	Yes	15.42	Yes	Language of the test	0.13793	0.209052	0.147423	423.59
225826	Indonesia	Male	Yes	15.92	No	Other language	1.36250	1.154700	-0.079800	262.27
102436	Switzerland	Female	Yes	16.08	No	Language of the test	-0.96320	0.221700	-1.673100	598.38
467828	Turkey	Female	Yes	16.08	Yes	Language of the test	0.25090	-0.580900	0.167200	535.60
428715	Serbia	Male	Yes	15.33	Yes	Language of the test	1.04160	0.485500	1.076800	592.85
326119	Mexico	Male	Yes	15.75	Yes	Language of the test	0.13793	0.209052	0.147423	380.59
339143	Mexico	Female	NA	16.08	No	NA	0.50540	1.382300	2.563000	314.38
308910	Latvia	Male	Yes	16.08	Yes	Language of the test	0.13793	0.209052	0.147423	486.29

In [50]:

```
pisa_2012_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 485490 entries, 0 to 485489
Data columns (total 12 columns):
 #   Column                       Non-Null Count   Dtype
---  ------                       --------------   -----
 0   Country                      485490 non-null  object
 1   Gender                       485490 non-null  object
 2   Dictionary                   485490 non-null  object
 3   Age                          485490 non-null  float64
 4   Literature                   485490 non-null  object
 5   Test Language                485490 non-null  object
 6   Formative Assessment         485490 non-null  float64
 7   Student Orientation          485490 non-null  float64
 8   Teacher-Directed Instruction 485490 non-null  float64
 9   Math Score                   485490 non-null  float64
 10  Reading Score                485490 non-null  float64
 11  Science Score                485490 non-null  float64
```

```
dtypes: float64(7), object(5)
memory usage: 44.4+ MB
```

Our dataframe is now clean and tidy. We're ready for Exploratory Data Analysis (EDA)

Exploratory Data Analysis

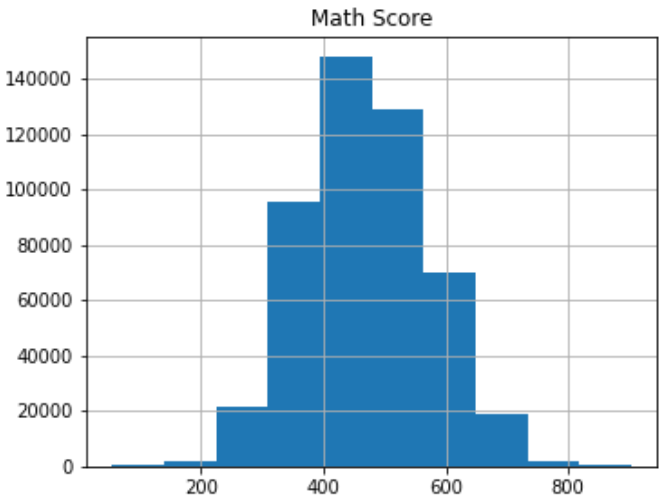
Method : Univariate Analysis

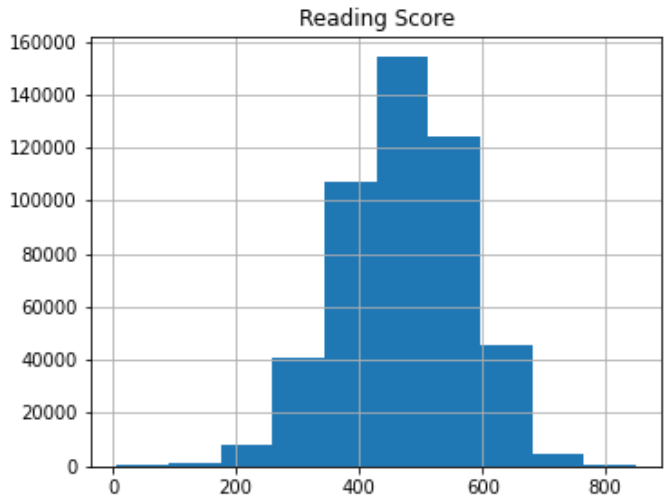
- Univariate visualization provides us summary statistics for one variable.

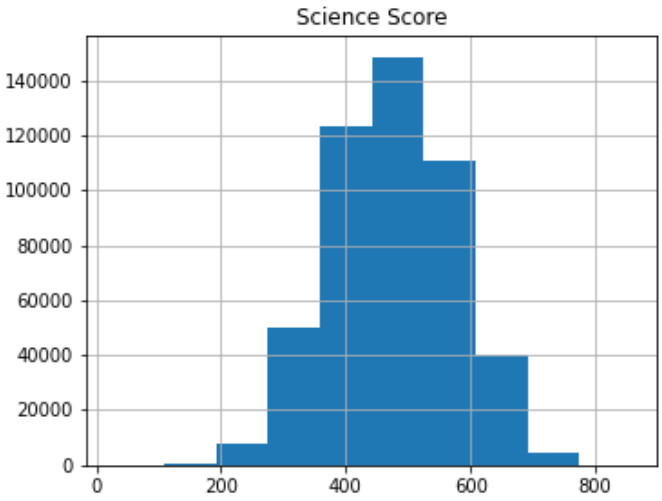
1) How did students perform in each subject?

In [51]:

```
# historgram gives the density of distributions from point to point in general terms.
# we want to see the distribution of scores for each of the subject
# we need 3 subplots as there's three subjects (Math, Reading, and Science)

features = ['Math Score','Reading Score','Science Score']
pisa_2012_clean[features].hist(figsize=(13, 10));
```





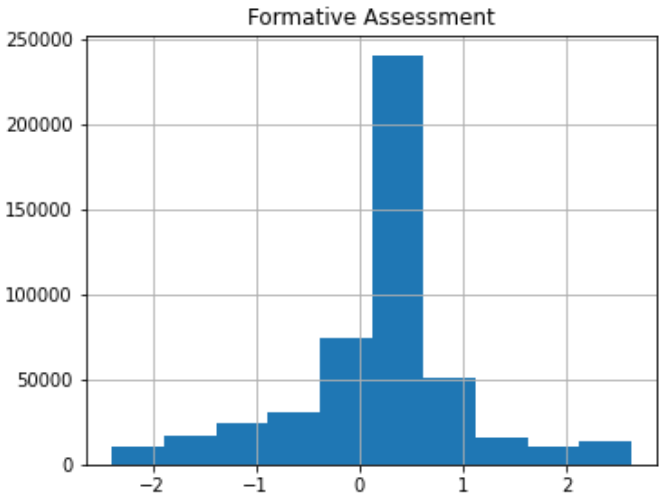


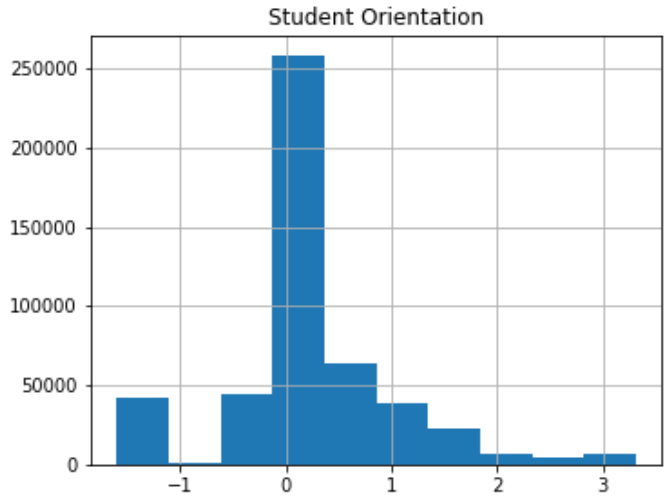
Histogram Visualization Analysis

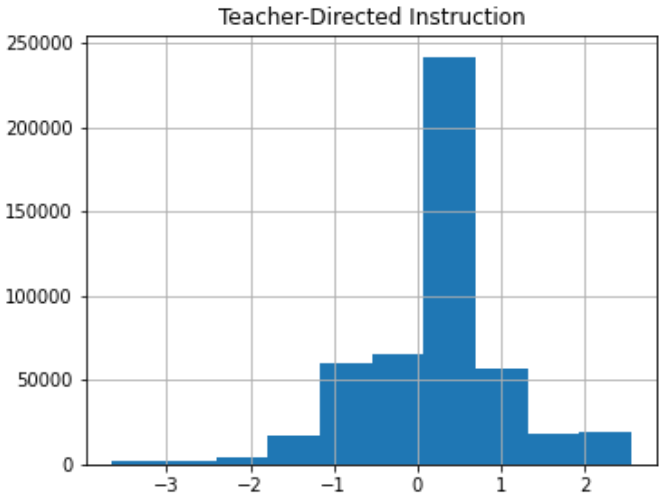
- In each subject, scores are normally distributed (bell curve)
- Distribution of each subject is unimodal
- Scores between 300 and 600 in each subject saw the highest student count

2) What was the distribution for each teacher behavior score?

In [52]:
```
features2 = ['Formative Assessment','Student Orientation','Teacher-Directed Instruction']
pisa_2012_clean[features2].hist(figsize=(13, 10));
```



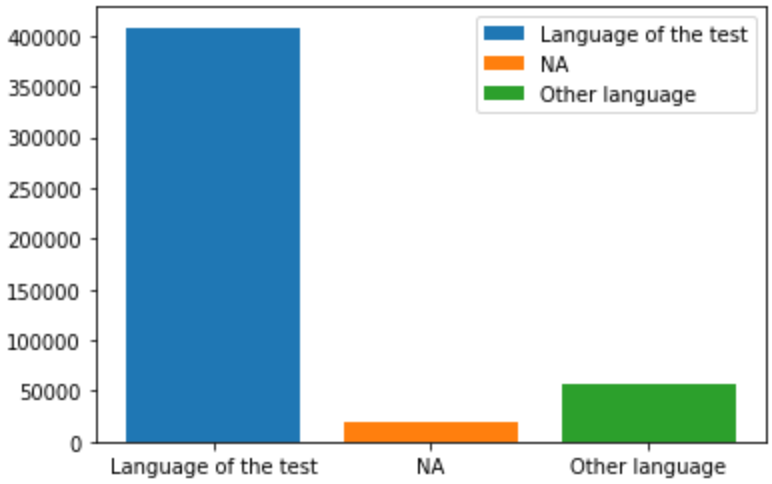




- This is some what unimodeal but we see that on average most students scored between 0 and 1.

3) About how many students were non-native speakers?

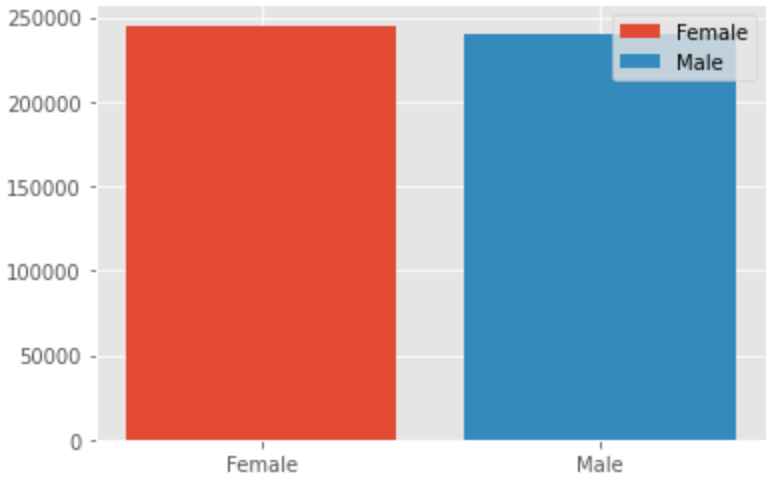
In [53]:
```
#https://stackoverflow.com/questions/43549901/visualize-data-from-one-column
labels = []
for i, dfi in enumerate(pisa_2012_clean.groupby(["Test Language"])):
    labels.append(dfi[0])
    plt.bar(i, dfi[1].count(), label=dfi[0])
plt.xticks(range(len(labels)), labels)
plt.legend()
plt.show()
```



- #### The vast majority of the test takers were native speakers as expected. And about 1.5% were non-native speakers when taking the exams.

4) Which gender was more represented?

In [60]:
```
#https://stackoverflow.com/questions/43549901/visualize-data-from-one-column
labels = []
for i, dfi in enumerate(pisa_2012_clean.groupby(["Gender"])):
    labels.append(dfi[0])
    plt.bar(i, dfi[1].count(), label=dfi[0])
plt.xticks(range(len(labels)), labels)
plt.legend()
plt.show()
```



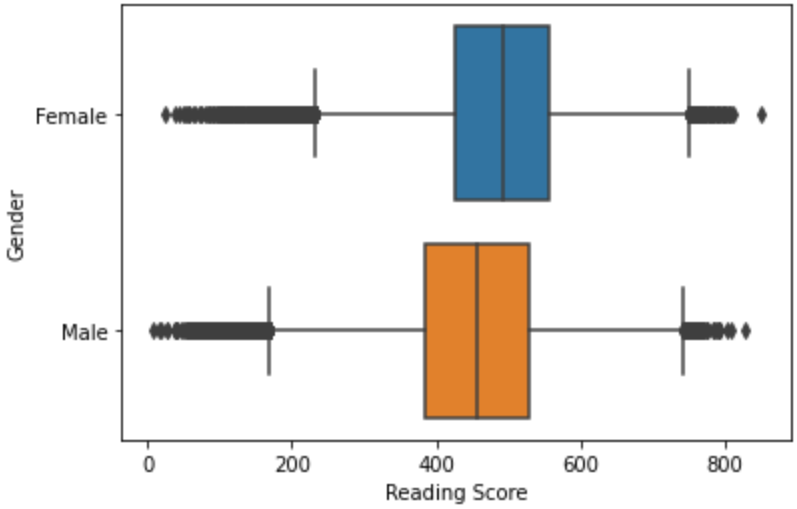
- Girls took the tests more than the boys but it's relatively the same!

Method 2: Bivariate Analysis

Bivariate analysis provide us the relationship between two variables in the dataset.

5) Which gender performed better in reading ?

In [54]:
```
import seaborn as sns
sns.boxplot(x = pisa_2012_clean['Reading Score'], y = pisa_2012_clean['Gender'] );
```

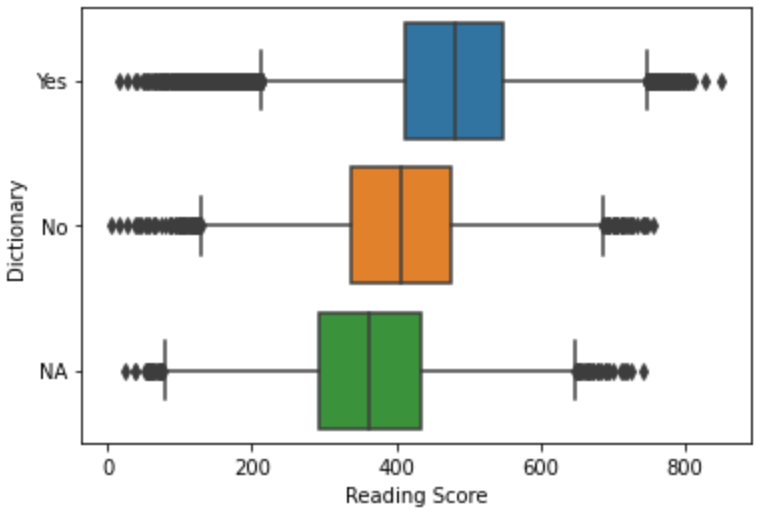


- Looking at the boxplots, we see that there's more outliers in the female on the left of the whisker. But they still outperformed their male counterparts greatly

- **Personally, I've heard that male students perform better than female students in subjects Math and Science. So I wanted to take this opporutnity to see if how female students compare to their male counterparts when it comes with Reading. Surpringsly, they outperform them by quite a margin.**

6) Did students who possess dictionaries perform better in reading section?

In [55]:
```
sns.boxplot(x = pisa_2012_clean['Reading Score'], y = pisa_2012_clean['Dictionary'] );
```

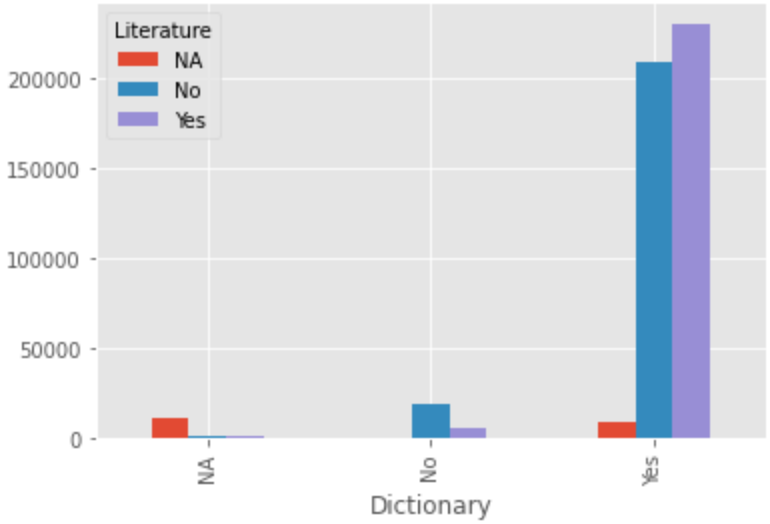


- Yes, students who possess dictionaries performed higher in reading section.

- **I expected this to be the answer and it was refreshing to see how having a possession of something leads to a either advantange/disadvantage in performance.Since dictionaries do carry our words and their meanings, it makes sense that we see the plot above. Although NA isn't a variable we are looking it at since we are doing Bivariate Analysis of two variables (Reading Score and Dictionary), it was interesting to see how NA scored the lowest.I think it may perhpas have to do with just not being able to read due to lack of resources (education,finance,support, etc.) as education is an investment and there's disaprities in our education system.**

7) Do students with dictionaries more likely to possess literature books?

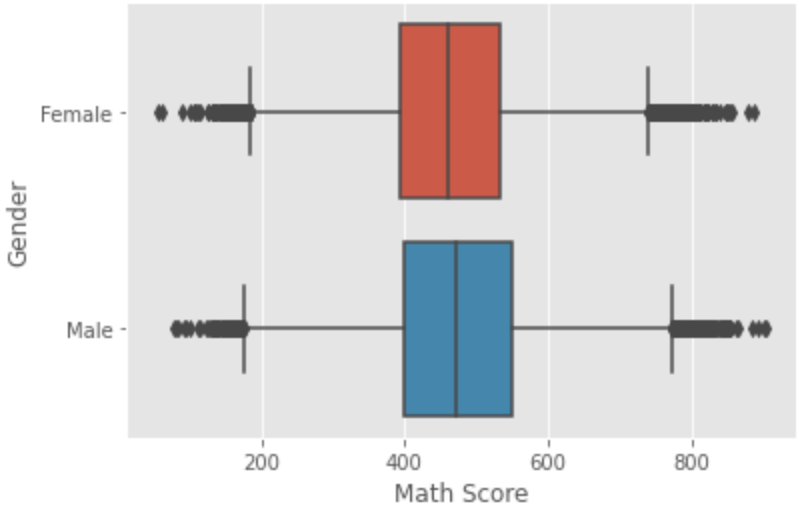
In [56]:
```
#https://stackoverflow.com/questions/47809646/how-to-make-a-histogram-for-non-numeric-vari

plt.style.use('ggplot')

pisa_2012_clean.groupby(['Dictionary', 'Literature'])\
      .Literature.count().unstack().plot.bar(legend=True)

plt.show()
```



- As expected, students who possess dictionaries also possess more literatures books. And likewise, there's more students who don't possess literature books among those who don't possess dictionaries.

8) Which gender performed better in Math ?

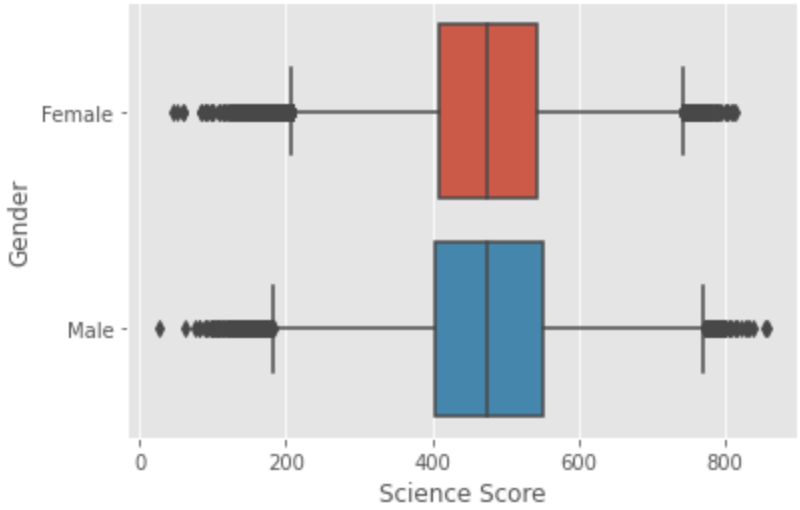
In [57]:
```
import seaborn as sns
sns.boxplot(x = pisa_2012_clean['Math Score'], y = pisa_2012_clean['Gender'] );
```



- Boys perfommed better than girls slightly better and we see that there's more outliers in girls pulling to the left, while for the boys, they have more outliers pulling to the right (of the whiskers).

9) Which gender performed better in Science ?

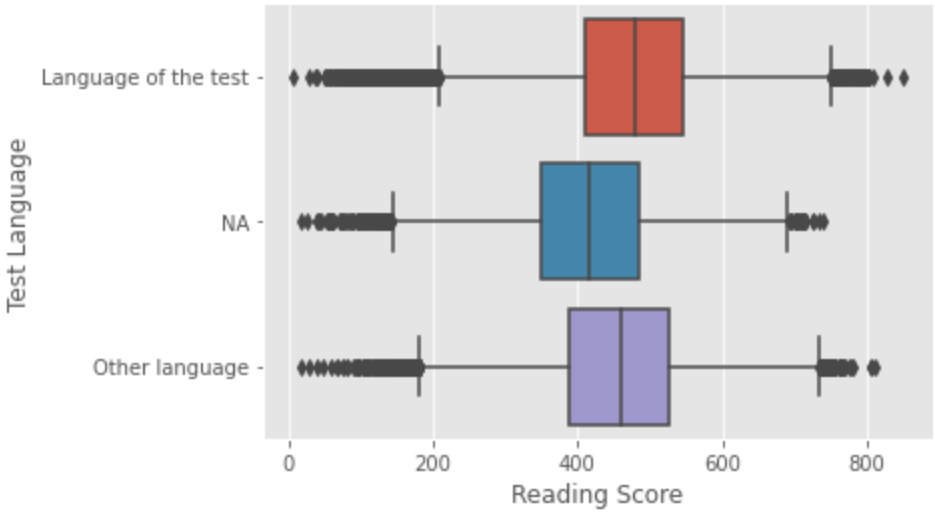
In [58]:
```
import seaborn as sns
sns.boxplot(x = pisa_2012_clean['Science Score'], y = pisa_2012_clean['Gender'] );
```



- By looking at the median of the boxes they scored just about the same.

10) How did non native speakers perform compared to non-native speakers, in Reading section ?

In [59]:
```
import seaborn as sns
sns.boxplot(x = pisa_2012_clean['Reading Score'], y = pisa_2012_clean['Test Language'] );
```

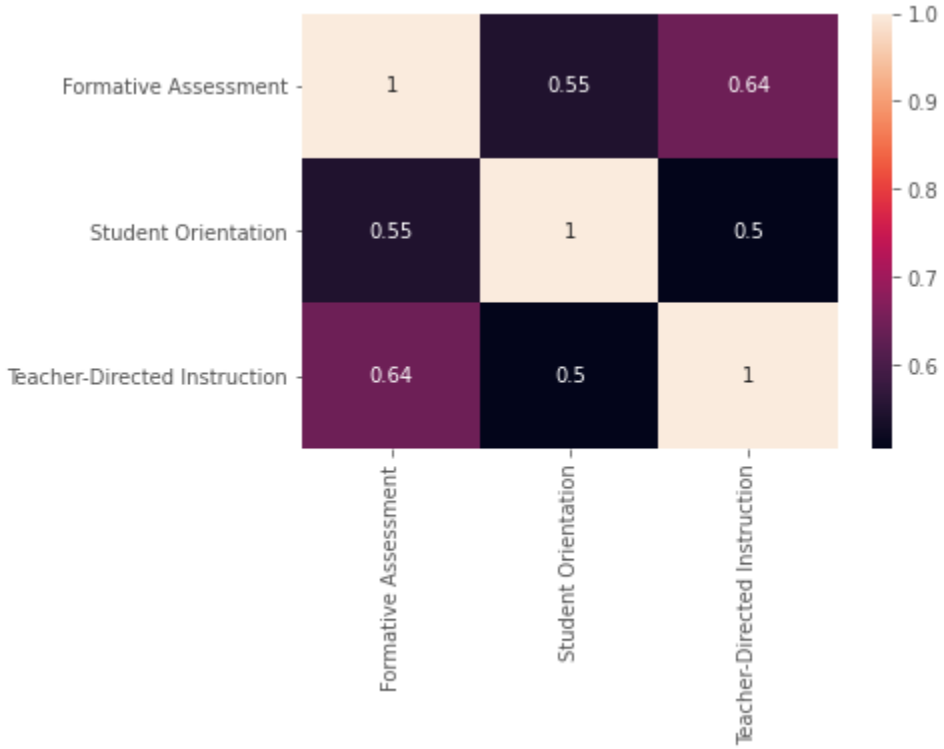


- Non-native speakers performed a little below than their counterparts in the Reading Section. This is very interesting to see.

Method : Multivariate Analysis

11) What's the correlation between the three teacher behavior measurements?

In [72]:
```
#https://datatofish.com/correlation-matrix-pandas/

df_1 = pd.DataFrame(pisa_2012_clean,columns=['Formative Assessment','Student Orientation',
corrMatrix = df_1.corr()
sns.heatmap(corrMatrix, annot=True)
plt.show()
```

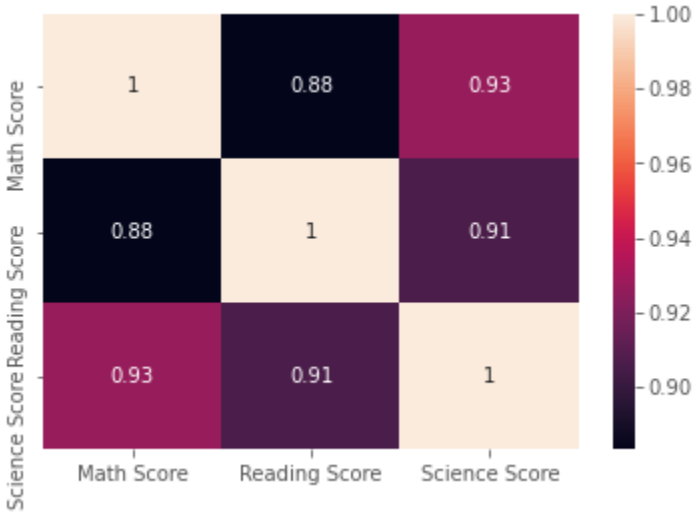


- **We see that Formative Assessment and Teacher-Directed Instruction have the highest correlation. It makes sense because Formative Asessment include things like diagnostic tests which are conducted by teachers. And Teacher-Directed Instruction goes with that notion where students are instructed to take exams/tests, etc.**

12) What's the correlation between the three subject scores?

In [73]:

```
df_2 = pd.DataFrame(pisa_2012_clean,columns=['Math Score','Reading Score','Science Score'])
corrMatrix = df_2.corr()
sns.heatmap(corrMatrix, annot=True)
plt.show()
```

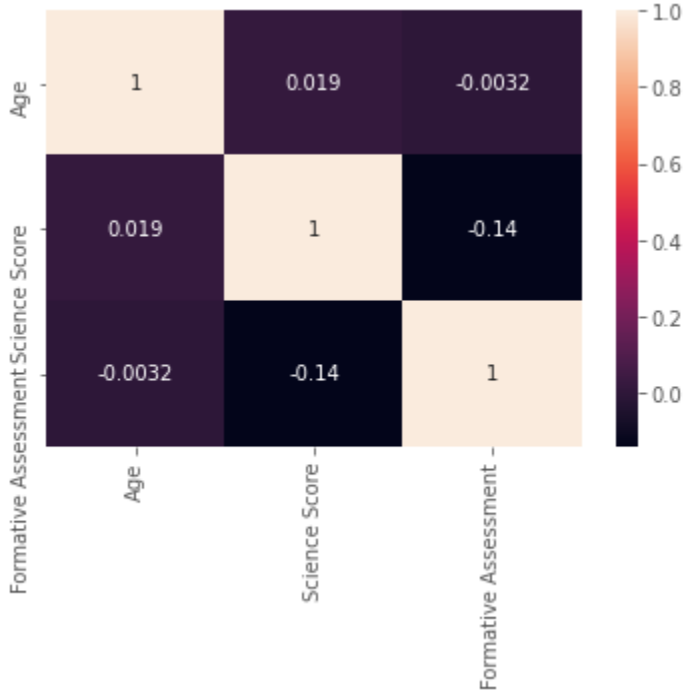


- **We can see that the correlation coefficient between Science Score and Math Score is the highest, and this is something that most people may already know. Further, Reading Score and Math Score had the lowest correlation out of the three, possibly explaning the phenomenon how some people are analytical while others are more creative.**

13) What's the correlation between Age, Science Score, and Formative Assessment?

In [62]:

```
df_3 = pd.DataFrame(pisa_2012_clean,columns=['Age','Science Score','Formative Assessment'])
corrMatrix = df_3.corr()
sns.heatmap(corrMatrix, annot=True)
plt.show()
```



- As we expected, this is the lowest correlation matrix we've seen. It's a mixture of Age which isn't a score, and two scores that's not related. One is a test score which a student earns and the other is measurement of Teacher Behavior which student is instructed with.