In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from IPython import get_ipython
import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
data = pd.read_csv('Fertile_Man_2020.csv')
```

In [3]:

```
data.head()
```

Out[3]:

	PI	Semen Volume (ml)	Sperm Concentration (106/ml)	Total Number (106)	Total Motility (%)	Progressive Motility (%)	Non-progressive Motility (%)	Immotile Spermatozoa (%)
0	Aboutorabi	3.2	27.0	86.4	35	20	15	65
1	Aboutorabi	0.8	136.0	108.8	47	35	12	53
2	Aboutorabi	2.0	71.0	142.0	49	42	7	51
3	Aboutorabi	1.0	35.0	35.0	50	28	22	50
4	Aboutorabi	2.0	46.0	92.0	51	28	33	49

In [4]:

```
1 data.tail()
```

Out[4]:

	PI	Semen Volume (ml)	Sperm Concentration (106/ml)	Total Number (106)	Total Motility (%)	Progressive Motility (%)	Non-progressive Motility (%)	Immotile Spermatozoa (%)	V
3584	Tang	1.7	23.0	39.1	53	52	1	NO RESULT	
3585	Tang	2.5	110.0	275.0	66	66	0	NO RESULT	
3586	Tang	2.0	109.0	218.0	64	44	20	36	RE
3587	Tang	6.2	96.0	595.2	39	29	10	61	RE
3588	Tang	3.0	36.0	108.0	54	38	16	46	RE

In [5]:

```
1 data.shape
```

Out[5]:

```
(3589, 10)
```

In [6]:

```
1 data.columns
```

Out[6]:

```
Index(['PI', 'Semen Volume (ml)', 'Sperm Concentration (106/ml)',
       'Total Number (106)', 'Total Motility (%)', 'Progressive Motility
(%)',
       'Non-progressive Motility (%)', 'Immotile Spermatozoa (%)',
       'Vitality (%)', 'Normal Forms (%)'],
      dtype='object')
```

In [7]:

```
1 data.duplicated().sum()
```

Out[7]:

```
220
```

In [8]:

```
1 data = data.drop_duplicates()
```

In [9]:

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

Out[9]:

```
PI                              0
Semen Volume (ml)               0
Sperm Concentration (106/ml)    0
Total Number (106)              0
Total Motility (%)              0
Progressive Motility (%)        0
Non-progressive Motility (%)    0
Immotile Spermatozoa (%)        0
Vitality (%)                    0
Normal Forms (%)                0
dtype: int64
```

In [10]:

```
1 data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3369 entries, 0 to 3588
Data columns (total 10 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   PI                             3369 non-null   object
 1   Semen Volume (ml)              3369 non-null   object
 2   Sperm Concentration (106/ml)   3369 non-null   object
 3   Total Number (106)             3369 non-null   object
 4   Total Motility (%)             3369 non-null   object
 5   Progressive Motility (%)       3369 non-null   object
 6   Non-progressive Motility (%)   3369 non-null   object
 7   Immotile Spermatozoa (%)       3369 non-null   object
 8   Vitality (%)                   3369 non-null   object
 9   Normal Forms (%)               3369 non-null   object
dtypes: object(10)
memory usage: 289.5+ KB
```

In [11]:

```
1 data.nunique()
```

Out[11]:

```
PI                                10
Semen Volume (ml)                 97
Sperm Concentration (106/ml)     644
Total Number (106)              1652
Total Motility (%)                88
Progressive Motility (%)          91
Non-progressive Motility (%)      52
Immotile Spermatozoa (%)          85
Vitality (%)                      66
Normal Forms (%)                  67
dtype: int64
```

In [12]:

```
1 data['PI'].unique()
```

Out[12]:

```
array(['Aboutorabi', 'Auger', 'Baker', 'Evgeni', 'Haugen', 'Jensen',
       'Lotti', 'Swan', 'Zedan', 'Tang'], dtype=object)
```

In [13]:

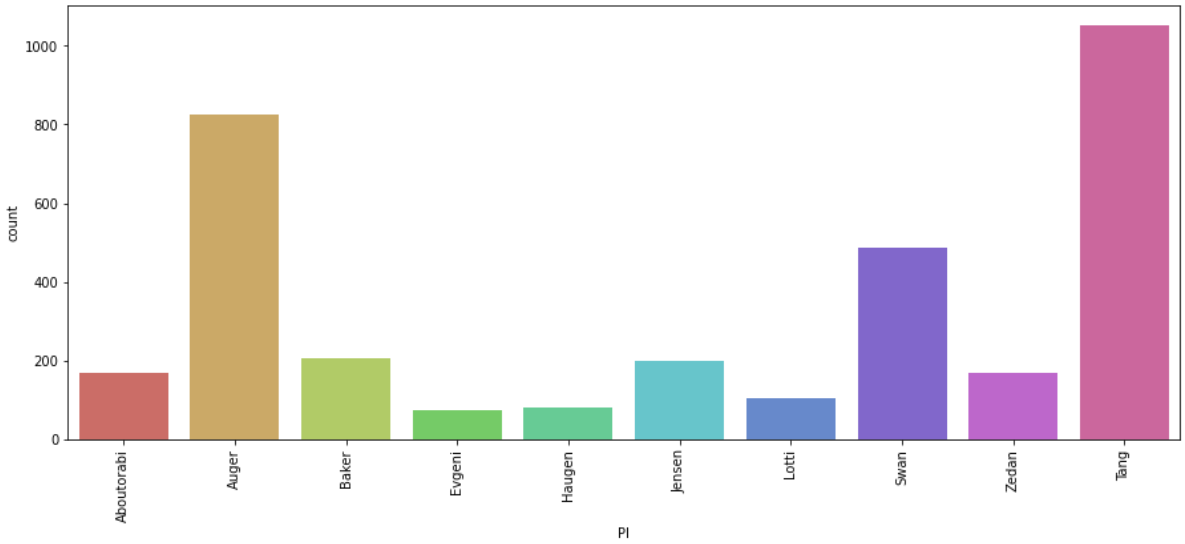
```
1 data['PI'].value_counts()
```

Out[13]:

```
Tang          1050
Auger          826
Swan           487
Baker          206
Jensen         199
Zedan          170
Aboutorabi     168
Lotti          105
Haugen          82
Evgeni          76
Name: PI, dtype: int64
```

In [14]:

```
plt.figure(figsize=(15,6))
sns.countplot('PI', data = data, palette = 'hls')
plt.xticks(rotation = 90)
plt.show()
```



In [15]:

```
data.columns
```

Out[15]:

```
Index(['PI', 'Semen Volume (ml)', 'Sperm Concentration (106/ml)',
       'Total Number (106)', 'Total Motility (%)', 'Progressive Motility
(%)',
       'Non-progressive Motility (%)', 'Immotile Spermatozoa (%)',
       'Vitality (%)', 'Normal Forms (%)'],
      dtype='object')
```

In [16]:

```
data_new = data[['Semen Volume (ml)', 'Sperm Concentration (106/ml)',
       'Total Number (106)', 'Total Motility (%)', 'Progressive Motility (%)',
       'Non-progressive Motility (%)', 'Immotile Spermatozoa (%)',
       'Vitality (%)', 'Normal Forms (%)']]
```

In [17]:

```
data_new = data_new.replace({'Semen Volume (ml)': {'NO RESULT': 0}})
data_new = data_new.replace({'Sperm Concentration (106/ml)': {'NO RESULT': 0}})
data_new = data_new.replace({'Total Number (106)': {'NO RESULT': 0}})
data_new = data_new.replace({'Total Motility (%)': {'NO RESULT': 0}})
data_new = data_new.replace({'Progressive Motility (%)': {'NO RESULT': 0}})
data_new = data_new.replace({'Non-progressive Motility (%)': {'NO RESULT': 0}})
data_new = data_new.replace({'Immotile Spermatozoa (%)': {'NO RESULT': 0}})
data_new = data_new.replace({'Normal Forms (%)': {'NO RESULT': 0}})
data_new = data_new.replace({'Vitality (%)': {'NO RESULT': 0}})
```

In [18]:

```
data_new
```

Out[18]:

	Semen Volume (ml)	Sperm Concentration (106/ml)	Total Number (106)	Total Motility (%)	Progressive Motility (%)	Non-progressive Motility (%)	Immotile Spermatozoa (%)	Vitality (%)
0	3.2	27.0	86.4	35	20	15	65	0
1	0.8	136.0	108.8	47	35	12	53	0
2	2.0	71.0	142.0	49	42	7	51	0
3	1.0	35.0	35.0	50	28	22	50	0
4	2.0	46.0	92.0	51	28	33	49	0
...	...	...	...	...	...	...	...	...
3579	2.0	115.0	230.0	79	77	2	0	82
3581	4.0	22.0	88.0	35	32	3	0	38
3586	2.0	109.0	218.0	64	44	20	36	0
3587	6.2	96.0	595.2	39	29	10	61	0
3588	3.0	36.0	108.0	54	38	16	46	0

3369 rows × 9 columns

In [19]:

```
for i in data_new.columns:
    data_new[i] = data_new[i].astype(float)
```

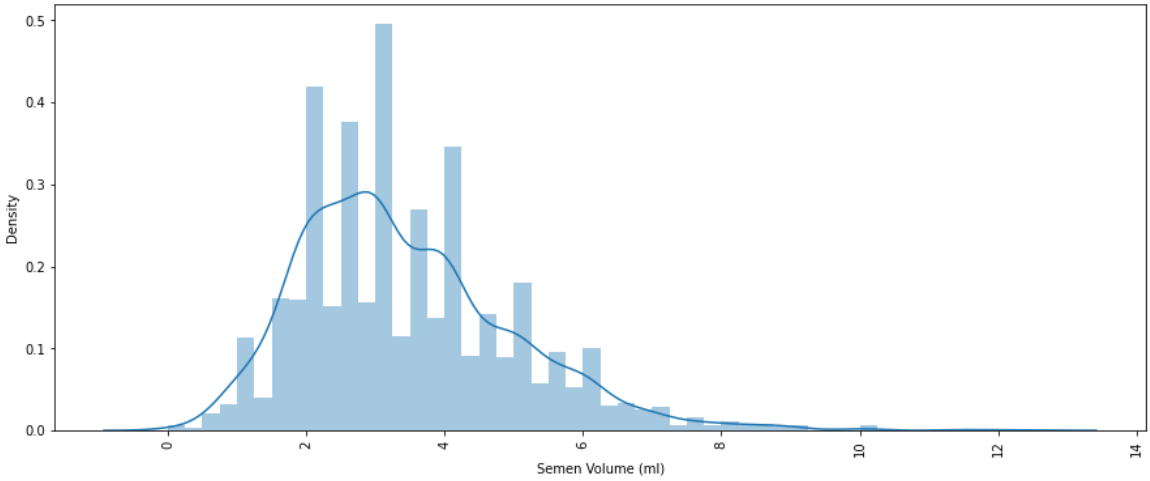
In [20]:

```
1 data_new.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3369 entries, 0 to 3588
Data columns (total 9 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Semen Volume (ml)               3369 non-null   float64
 1   Sperm Concentration (106/ml)    3369 non-null   float64
 2   Total Number (106)              3369 non-null   float64
 3   Total Motility (%)              3369 non-null   float64
 4   Progressive Motility (%)        3369 non-null   float64
 5   Non-progressive Motility (%)    3369 non-null   float64
 6   Immotile Spermatozoa (%)        3369 non-null   float64
 7   Vitality (%)                    3369 non-null   float64
 8   Normal Forms (%)                3369 non-null   float64
dtypes: float64(9)
memory usage: 263.2 KB
```

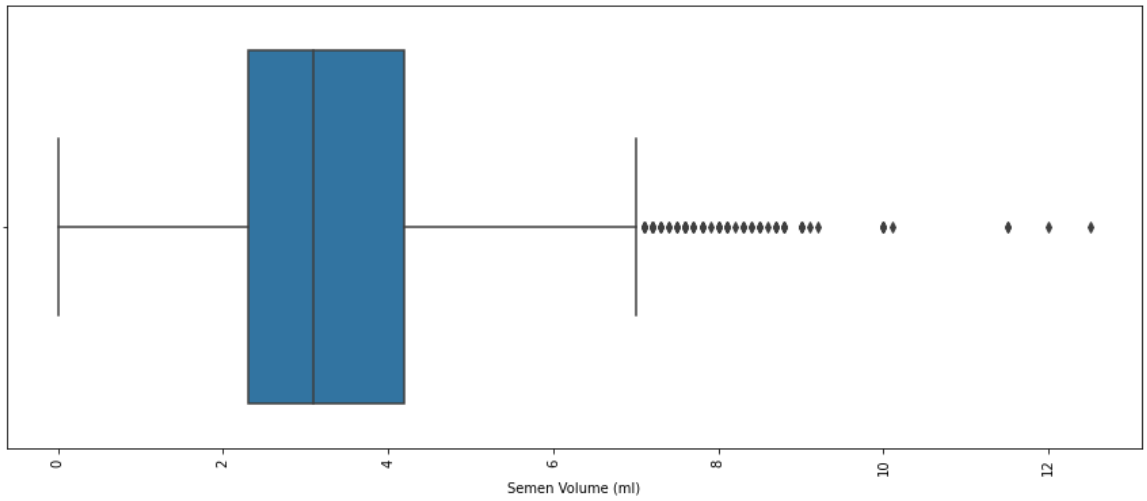
In [21]:

```
1 for i in data_new.columns:
2     plt.figure(figsize=(15,6))
3     sns.distplot(data_new[i])
4     plt.xticks(rotation = 90)
5     plt.show()
```



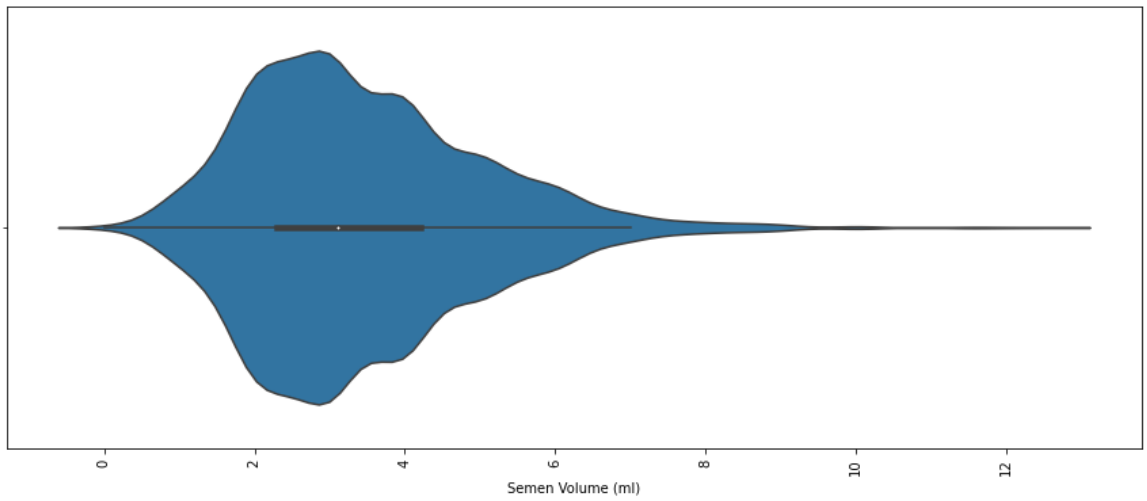
In [22]:

```
for i in data_new.columns:
    plt.figure(figsize=(15,6))
    sns.boxplot(data_new[i])
    plt.xticks(rotation = 90)
    plt.show()
```



In [23]:

```
for i in data_new.columns:
    plt.figure(figsize=(15,6))
    sns.violinplot(data_new[i])
    plt.xticks(rotation = 90)
    plt.show()
```



In [24]:

```
1 corrmat = data_new.corr()
2 corrmat
```

Out[24]:

	Semen Volume (ml)	Sperm Concentration (106/ml)	Total Number (106)	Total Motility (%)	Progressive Motility (%)	Non-progressive Motility (%)	Imn Spermat
Semen Volume (ml)	1.000000	-0.132747	0.394834	0.003056	0.093729	-0.163921	0.09
Sperm Concentration (106/ml)	-0.132747	1.000000	0.762657	0.070431	0.082049	-0.016940	0.03
Total Number (106)	0.394834	0.762657	1.000000	0.056822	0.127535	-0.116299	0.07
Total Motility (%)	0.003056	0.070431	0.056822	1.000000	0.772534	0.377100	-0.40
Progressive Motility (%)	0.093729	0.082049	0.127535	0.772534	1.000000	-0.077893	-0.38
Non-progressive Motility (%)	-0.163921	-0.016940	-0.116299	0.377100	-0.077893	1.000000	-0.08
Immotile Spermatozoa (%)	0.095168	0.034547	0.070889	-0.400565	-0.388460	-0.089031	1.00
Vitality (%)	-0.156211	-0.170896	-0.210887	0.364078	0.324613	0.148656	-0.52
Normal Forms (%)	0.002695	0.109208	0.117101	-0.226283	-0.237217	-0.290616	-0.17

In [25]:

```
cmap = sns.diverging_palette(260,-10,s=50, l=75, n=6,
                             as_cmap=True)
plt.subplots(figsize=(18,18))
sns.heatmap(corrmat,cmap= cmap,annot=True, square=True)
plt.show()
```