

```
In [1]:
import numpy as np
import pandas as pd
```

```
In [2]:
df = pd.read_csv('osic.csv')
```

```
In [3]:
df.head()
```

Out[3]:

	Patient	Weeks	FVC	Percent	Age	Sex	SmokingStatus
0	ID00007637202177411956430	-4	2315	58.253649	79	Male	Ex-smoker
1	ID00007637202177411956430	5	2214	55.712129	79	Male	Ex-smoker
2	ID00007637202177411956430	7	2061	51.862104	79	Male	Ex-smoker
3	ID00007637202177411956430	9	2144	53.950679	79	Male	Ex-smoker
4	ID00007637202177411956430	11	2069	52.063412	79	Male	Ex-smoker

```
In [4]:
df.tail()
```

Out[4]:

	Patient	Weeks	FVC	Percent	Age	Sex	SmokingStatus
1544	ID00426637202313170790466	13	2712	66.594637	73	Male	Never smoked
1545	ID00426637202313170790466	19	2978	73.126412	73	Male	Never smoked
1546	ID00426637202313170790466	31	2908	71.407524	73	Male	Never smoked
1547	ID00426637202313170790466	43	2975	73.052745	73	Male	Never smoked
1548	ID00426637202313170790466	59	2774	68.117081	73	Male	Never smoked

```
In [5]:
df.shape
```

Out[5]:
(1549, 7)

```
In [6]:
df.columns
```

Out[6]:

```
Index(['Patient', 'Weeks', 'FVC', 'Percent', 'Age', 'Sex', 'SmokingStatus'], dtype='object')
```

```
In [7]:
df.duplicated().sum()
```

Out[7]:

```
0
```

```
In [8]:
df.isnull().sum()
```

Out[8]:

```
Patient      0
Weeks        0
FVC          0
Percent      0
Age          0
Sex          0
SmokingStatus 0
dtype: int64
```

```
In [9]:
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1549 entries, 0 to 1548
Data columns (total 7 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Patient         1549 non-null  object
1   Weeks           1549 non-null  int64
2   FVC             1549 non-null  int64
3   Percent         1549 non-null  float64
4   Age             1549 non-null  int64
5   Sex             1549 non-null  object
6   SmokingStatus   1549 non-null  object
dtypes: float64(1), int64(3), object(3)
memory usage: 84.8+ KB
```

```
In [10]:
df.describe()
```

Out[10]:

	Weeks	FVC	Percent	Age
count	1549.000000	1549.000000	1549.000000	1549.000000
mean	31.861846	2690.479019	77.672654	67.188509
std	23.247550	832.770959	19.823261	7.057395
min	-5.000000	827.000000	28.877577	49.000000
25%	12.000000	2109.000000	62.832700	63.000000
50%	28.000000	2641.000000	75.676937	68.000000
75%	47.000000	3171.000000	88.621065	72.000000
max	133.000000	6399.000000	153.145378	88.000000

```
In [11]:
df.nunique()
```

Out[11]:

Patient	176
Weeks	112
FVC	1202
Percent	1536
Age	34
Sex	2
SmokingStatus	3
dtype:	int64

```
In [12]:
df1 = df.copy()
```

```
In [13]:
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

```
In [14]:
df['Sex'].unique()
```

Out[14]:

array(['Male', 'Female'], dtype=object)

In [15]:

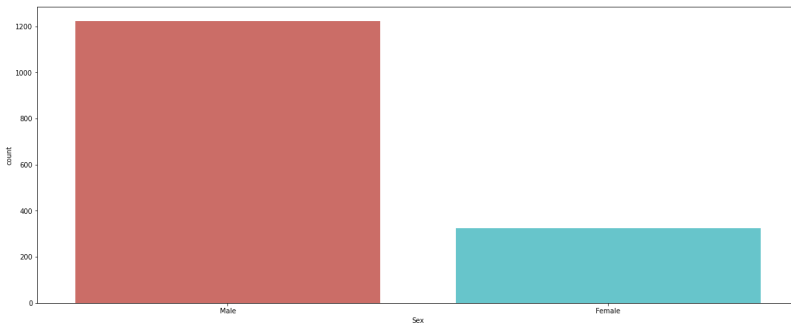
```
df['Sex'].value_counts()
```

Out[15]:

```
Male      1224  
Female    325  
Name: Sex, dtype: int64
```

In [16]:

```
plt.figure(figsize=(20,8))  
sns.countplot('Sex', data = df, palette = 'hls')  
plt.show()
```



In [17]:

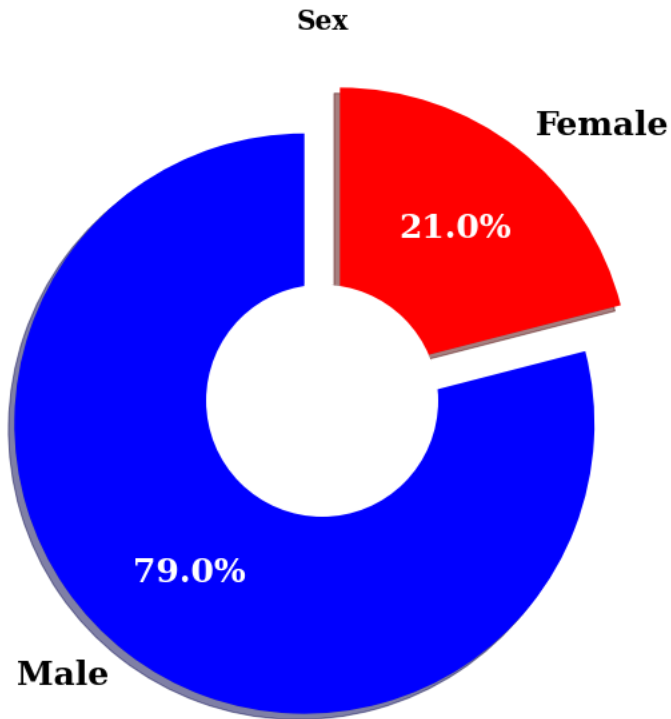
```
label_data = df['Sex'].value_counts()

explode = (0.1, 0.1)
plt.figure(figsize=(14, 10))
patches, texts, pcts = plt.pie(label_data,
                                labels = label_data.index,
                                colors = ['blue', 'red'],
                                pctdistance = 0.65,
                                shadow = True,
                                startangle = 90,
                                explode = explode,
                                autopct = '%1.1f%%',
                                textprops={ 'fontsize': 25,
                                              'color': 'black',
                                              'weight': 'bold',
                                              'family': 'serif' })

plt.setp(pcts, color='white')

hfont = {'fontname': 'serif', 'weight': 'bold'}
plt.title('Sex', size=20, **hfont)

centre_circle = plt.Circle((0,0),0.40,fc='white')
fig = plt.gcf()
fig.gca().add_artist(centre_circle)
plt.show()
```



In [18]:

```
df['SmokingStatus'].unique()
```

Out[18]:

```
array(['Ex-smoker', 'Never smoked', 'Currently smokes'], dtype=object)
```

In [19]:

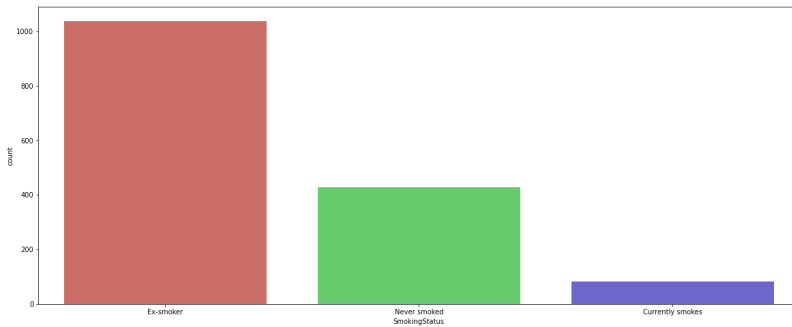
```
df['SmokingStatus'].value_counts()
```

Out[19]:

```
Ex-smoker      1038
Never smoked    429
Currently smokes    82
Name: SmokingStatus, dtype: int64
```

In [20]:

```
plt.figure(figsize=(20,8))  
sns.countplot('SmokingStatus', data = df, palette = 'hls')  
plt.show()
```



In [21]:

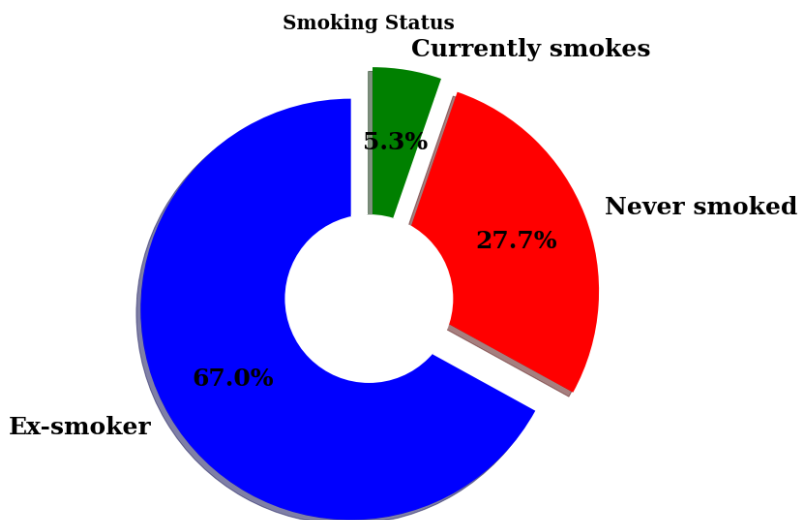
```
label_data = df['SmokingStatus'].value_counts()

explode = (0.1, 0.1, 0.1)
plt.figure(figsize=(14, 10))
patches, texts, pcts = plt.pie(label_data,
                                labels = label_data.index,
                                colors = ['blue', 'red', 'green'],
                                pctdistance = 0.65,
                                shadow = True,
                                startangle = 90,
                                explode = explode,
                                autopct = '%1.1f%%',
                                textprops={ 'fontsize': 25,
                                              'color': 'black',
                                              'weight': 'bold',
                                              'family': 'serif' })

plt.setp(pcts, color='black')

hfont = {'fontname':'serif', 'weight': 'bold'}
plt.title('Smoking Status', size=20, **hfont)

centre_circle = plt.Circle((0,0),0.40,fc='white')
fig = plt.gcf()
fig.gca().add_artist(centre_circle)
plt.show()
```



In [22]:

```
df = df.set_index('Patient')
```



```
In [23]:  
  
df
```

Out[23]:

	Weeks	FVC	Percent	Age	Sex	SmokingStatus
Patient						
ID00007637202177411956430	-4	2315	58.253649	79	Male	Ex-smoker
ID00007637202177411956430	5	2214	55.712129	79	Male	Ex-smoker
ID00007637202177411956430	7	2061	51.862104	79	Male	Ex-smoker
ID00007637202177411956430	9	2144	53.950679	79	Male	Ex-smoker
ID00007637202177411956430	11	2069	52.063412	79	Male	Ex-smoker
...
ID00426637202313170790466	13	2712	66.594637	73	Male	Never smoked
ID00426637202313170790466	19	2978	73.126412	73	Male	Never smoked
ID00426637202313170790466	31	2908	71.407524	73	Male	Never smoked
ID00426637202313170790466	43	2975	73.052745	73	Male	Never smoked
ID00426637202313170790466	59	2774	68.117081	73	Male	Never smoked

1549 rows × 6 columns

```
In [24]:  
  
from sklearn import preprocessing  
label_encoder = preprocessing.LabelEncoder()  
df['Sex']= label_encoder.fit_transform(df['Sex'])  
df['SmokingStatus']= label_encoder.fit_transform(df['SmokingStatus'])
```

```
In [25]:  
  
df['Weeks'].unique()
```

Out[25]:

array([-4,	5,	7,	9,	11,	17,	29,	41,	57,	8,	13,	15,	22,
	33,	45,	60,	0,	1,	3,	25,	37,	54,	6,	19,	32,	43,
	58,	35,	39,	47,	71,	87,	2,	4,	14,	26,	12,	21,	31,
	40,	52,	69,	16,	18,	20,	38,	53,	66,	23,	44,	70,	-3,
	27,	55,	49,	51,	81,	98,	30,	34,	36,	42,	65,	59,	24,
	63,	10,	61,	-1,	48,	56,	75,	28,	76,	46,	50,	83,	62,
	79,	-5,	73,	102,	82,	97,	100,	67,	91,	107,	85,	72,	84,
	78,	64,	68,	89,	101,	116,	-2,	74,	92,	104,	117,	133,	96,
	77,	94,	86,	88,	95,	93,	80,	99],	dtype=int64)				

In [26]:

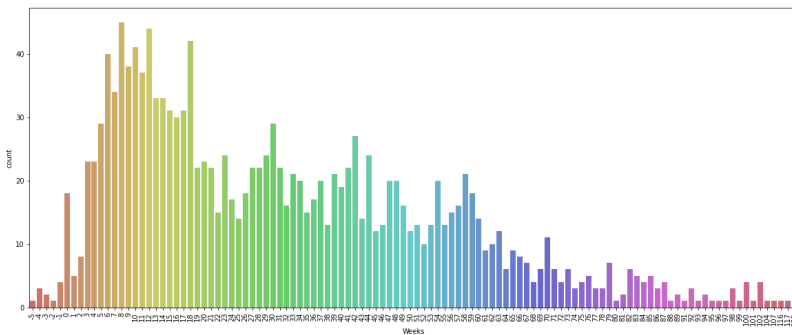
```
df['Weeks'].value_counts()
```

Out[26]:

```
8      45
12     44
18     42
10     41
6      40
..
104     1
91      1
107     1
-2      1
99      1
Name: Weeks, Length: 112, dtype: int64
```

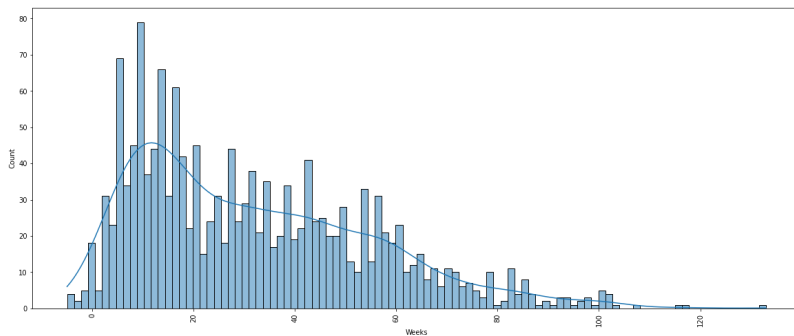
In [27]:

```
plt.figure(figsize=(20,8))
sns.countplot('Weeks', data = df, palette = 'hls')
plt.xticks(rotation = 90)
plt.show()
```



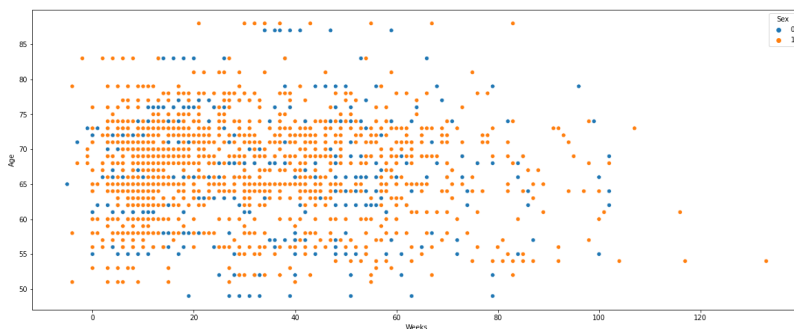
In [28]:

```
plt.figure(figsize=(20,8))
sns.histplot(df['Weeks'], bins = 100, kde = True, palette = 'hls')
plt.xticks(rotation = 90)
plt.show()
```



In [29]:

```
plt.figure(figsize=(20,8))
sns.scatterplot(data = df, x="Weeks", y="Age", hue = 'Sex')
plt.show()
```



In [30]:

```
df['FVC'].unique()
```

Out[30]:

```
array([2315, 2214, 2061, ..., 2712, 2978, 2774], dtype=int64)
```

In [31]:

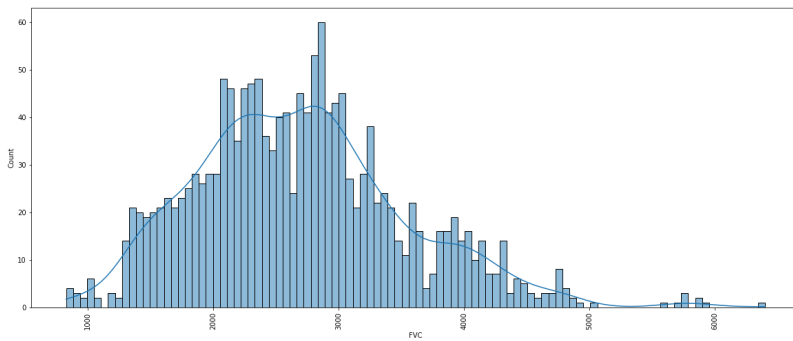
```
df['FVC'].value_counts()
```

Out[31]:

```
2474    4
2965    4
2889    4
2095    4
2708    4
..
3906    1
3780    1
3925    1
3907    1
2774    1
Name: FVC, Length: 1202, dtype: int64
```

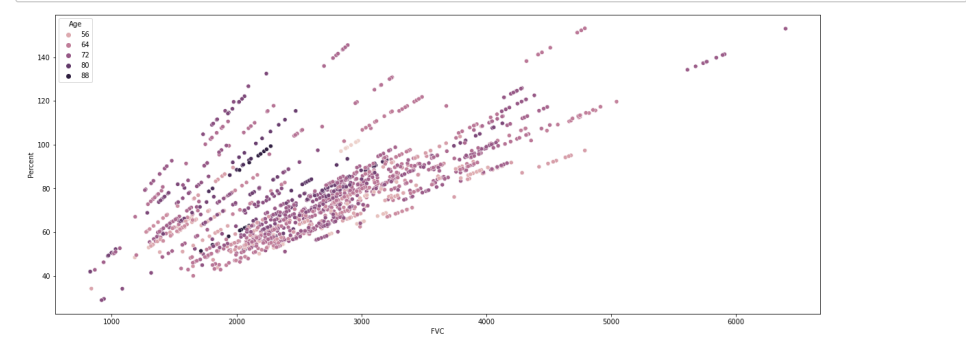
In [32]:

```
plt.figure(figsize=(20,8))
sns.histplot(df['FVC'], bins = 100, kde = True, palette = 'hls')
plt.xticks(rotation = 90)
plt.show()
```



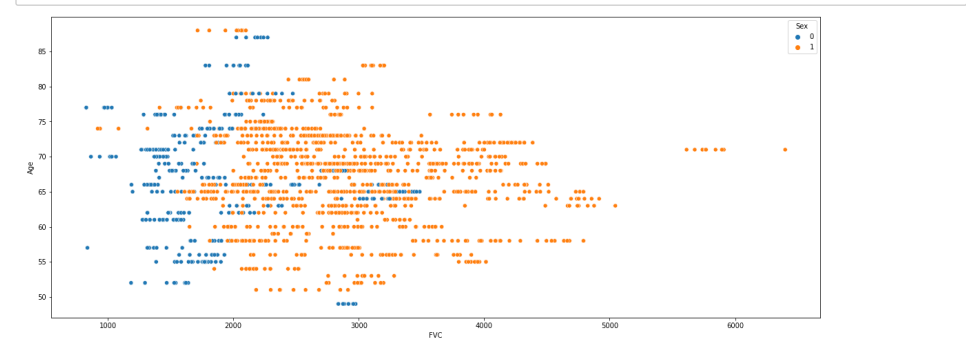
In [33]:

```
plt.figure(figsize=(20,8))
sns.scatterplot(data = df, x="FVC", y="Percent", hue ='Age')
plt.show()
```



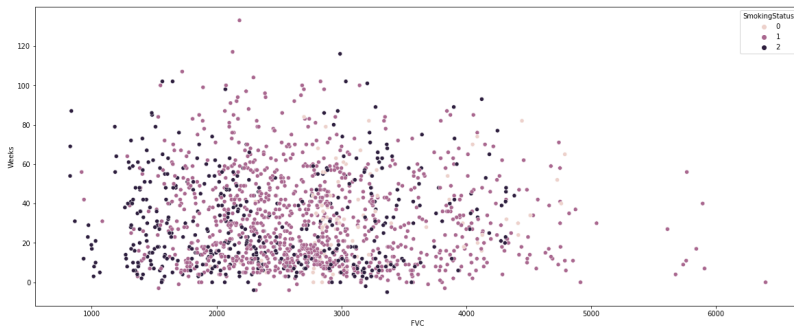
In [34]:

```
plt.figure(figsize=(20,8))
sns.scatterplot(data = df, x="FVC", y="Age", hue ='Sex')
plt.show()
```



In [35]:

```
plt.figure(figsize=(20,8))
sns.scatterplot(data = df, x="FVC", y="Weeks", hue = 'SmokingStatus')
plt.show()
```



In [36]:

```
df['Percent'].unique()
```

Out[36]:

```
array([58.25364872, 55.71212884, 51.86210367, ..., 71.40752382,
       73.05274531, 68.11708084])
```

In [37]:

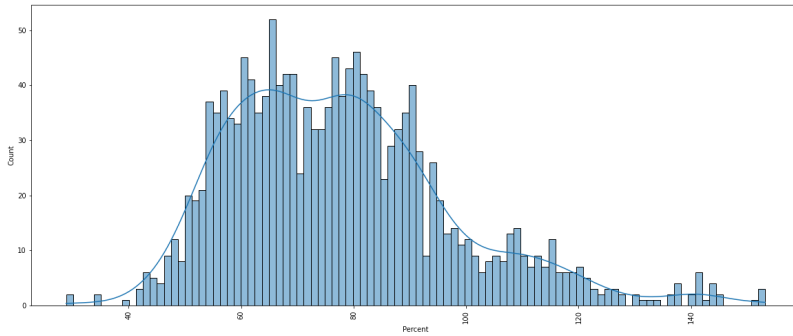
```
df['Percent'].value_counts()
```

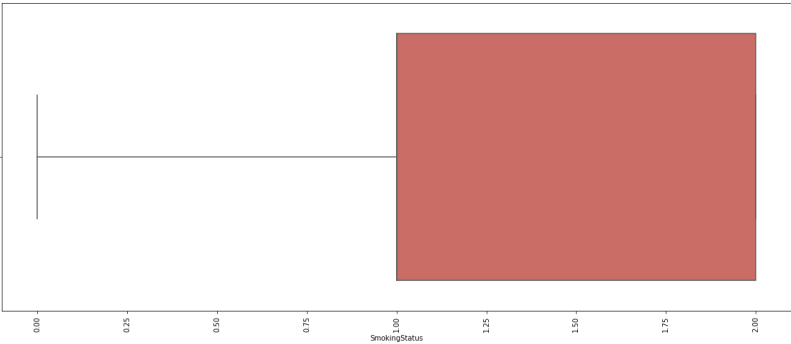
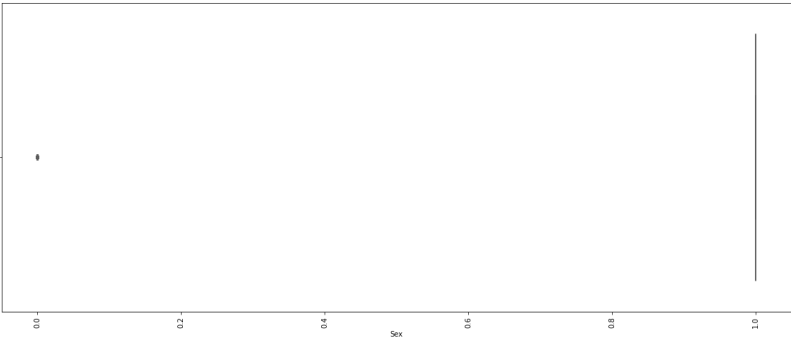
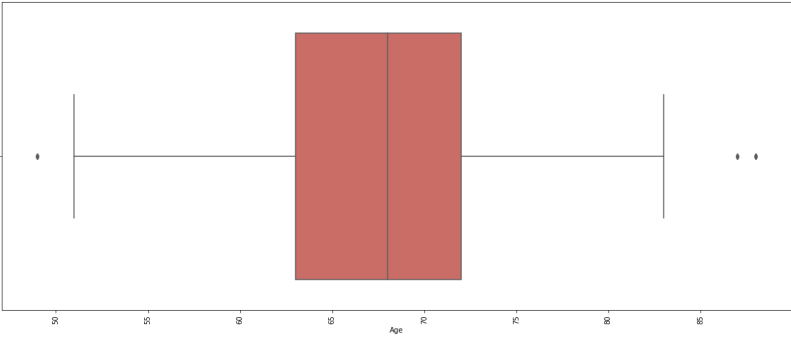
Out[37]:

```
87.795153    2
83.282505    2
80.474296    2
57.897831    2
94.644367    2
..
69.492021    1
49.475076    1
55.840291    1
57.534926    1
68.117081    1
Name: Percent, Length: 1536, dtype: int64
```

In [38]:

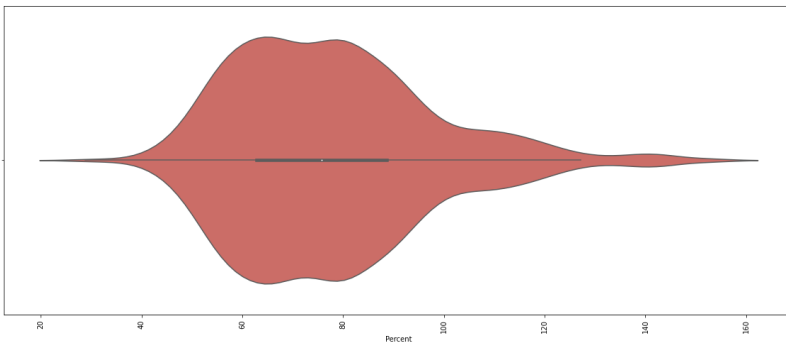
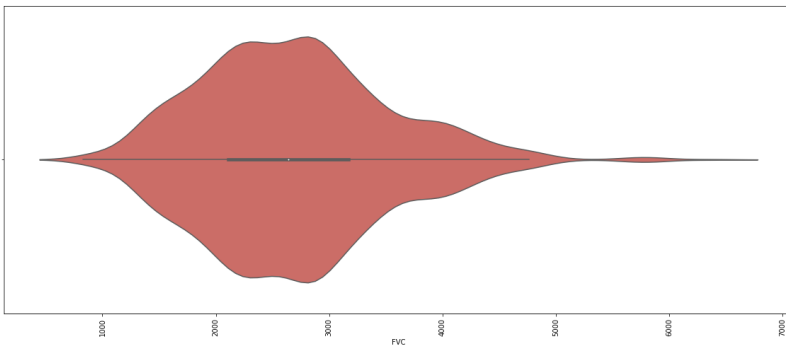
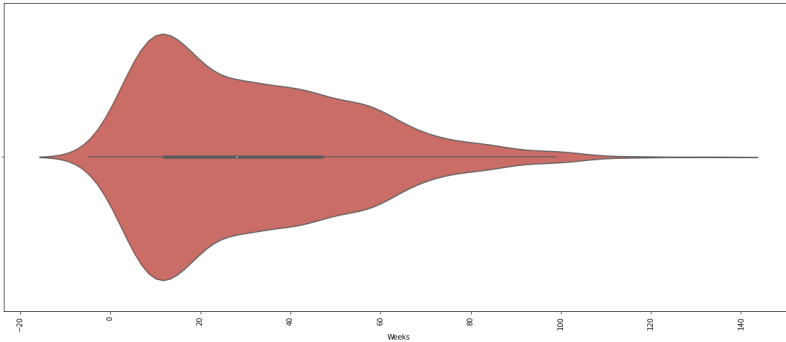
```
plt.figure(figsize=(20,8))
sns.histplot(df['Percent'], bins = 100, kde = True, palette = 'hls')
plt.xticks(rotation = 90)
plt.show()
```

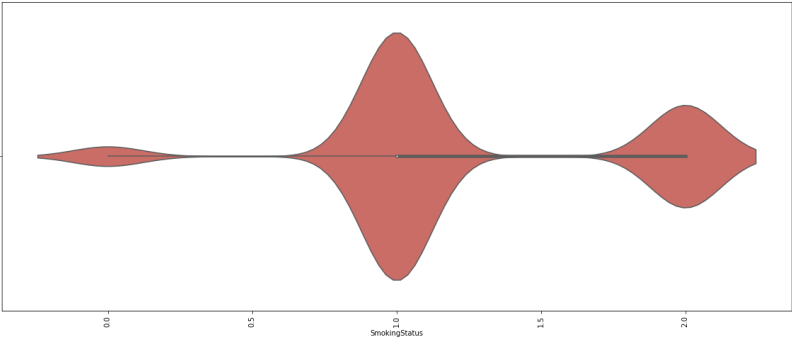
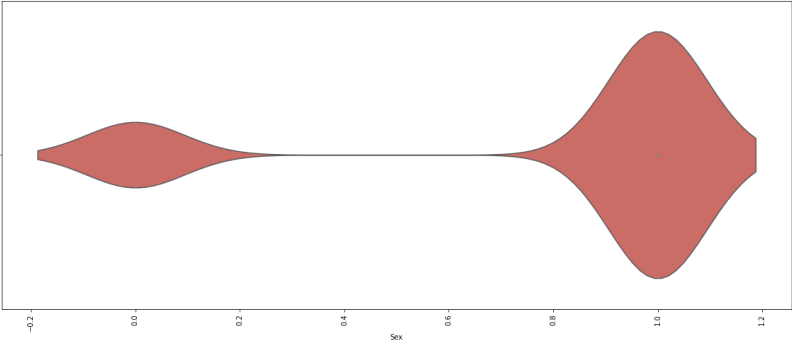
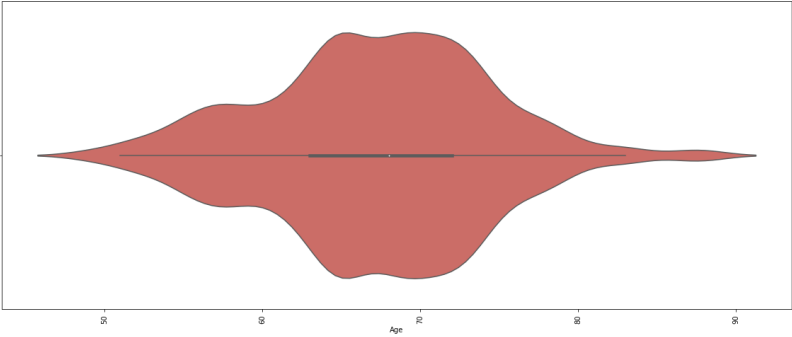




In [40]:

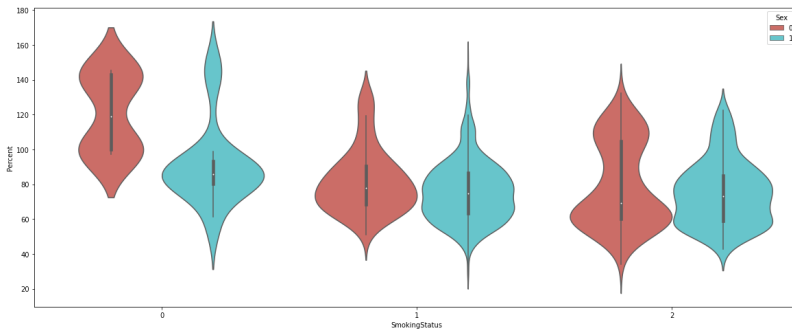
```
for i in df.columns:  
    plt.figure(figsize=(20,8))  
    sns.violinplot(df[i], palette = 'hls')  
    plt.xticks(rotation = 90)  
    plt.show()
```





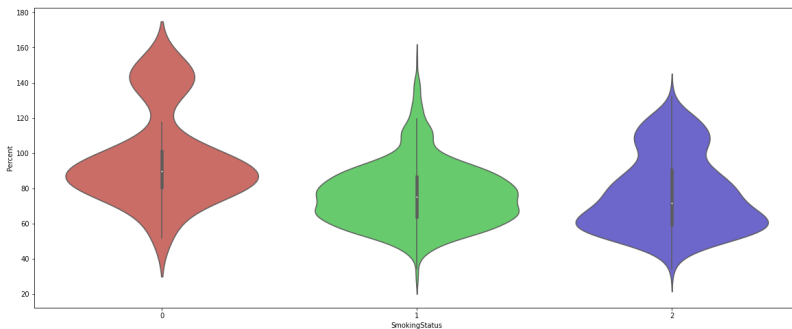
In [41]:

```
plt.figure(figsize=(20,8))
sns.violinplot(data = df, y='Percent', x='SmokingStatus', hue = 'Sex',
               palette = 'hls')
plt.show()
```



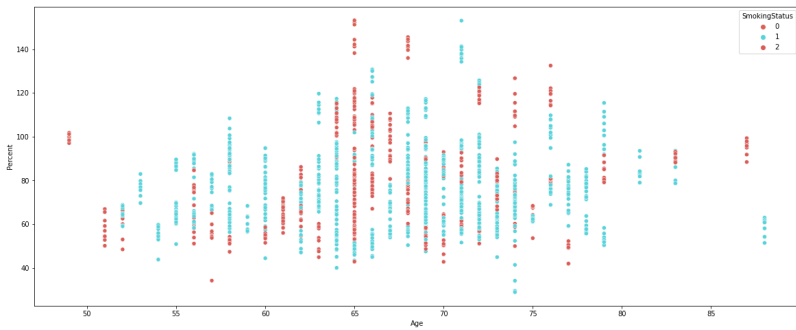
In [42]:

```
plt.figure(figsize=(20,8))
sns.violinplot(data = df, y='Percent', x='SmokingStatus',
               palette = 'hls')
plt.show()
```



In [43]:

```
plt.figure(figsize=(20,8))
sns.scatterplot(data = df, y='Percent', x='Age', hue = 'SmokingStatus',
               palette = 'hls')
plt.show()
```



In [44]:

```
df1['Patient'].unique()
```

Out[44]:

```

array(['ID00007637202177411956430', 'ID00009637202177434476278',
      'ID00010637202177584971671', 'ID00011637202177653955184',
      'ID00012637202177665765362', 'ID00014637202177757139317',
      'ID00015637202177877247924', 'ID00019637202178323708467',
      'ID00020637202178344345685', 'ID00023637202179104603099',
      'ID00025637202179541264076', 'ID00026637202179561894768',
      'ID00027637202179689871102', 'ID00030637202181211009029',
      'ID00032637202181710233084', 'ID00035637202182204917484',
      'ID00038637202182690843176', 'ID00042637202184406822975',
      'ID00047637202184938901501', 'ID00048637202185016727717',
      'ID00051637202185848464638', 'ID00052637202186188008618',
      'ID00060637202187965290703', 'ID00061637202188184085559',
      'ID00062637202188654068490', 'ID00067637202189903532242',
      'ID00068637202190879923934', 'ID00072637202198161894406',
      'ID00073637202198167792918', 'ID00075637202198610425520',
      'ID00076637202199015035026', 'ID00077637202199102000916',
      'ID00078637202199415319443', 'ID00082637202201836229724',
      'ID00086637202203494931510', 'ID00089637202204675567570',
      'ID00090637202204766623410', 'ID00093637202205278167493',
      'ID00094637202205333947361', 'ID00099637202206203808121',
      'ID00102637202206574119190', 'ID00104637202208063407045',
      'ID00105637202208831864134', 'ID00108637202209619669361',
      'ID00109637202210454292264', 'ID00110637202210673668310',
      'ID00111637202210956877205', 'ID00115637202211874187958',
      'ID00117637202212360228007', 'ID00119637202215426335765',
      'ID00122637202216437668965', 'ID00123637202217151272140',
      'ID00124637202217596410344', 'ID00125637202218590429387',
      'ID00126637202218610655908', 'ID00127637202219096738943',
      'ID00128637202219474716089', 'ID00129637202219868188000',
      'ID00130637202220059448013', 'ID00131637202220424084844',
      'ID0013263720222178761324', 'ID00133637202223847701934',
      'ID00134637202223873059688', 'ID00135637202224630271439',
      'ID00136637202224951350618', 'ID00138637202231603868088',
      'ID00139637202231703564336', 'ID00140637202231728595149',
      'ID00149637202232704462834', 'ID00161637202235731948764',
      'ID00168837202237320314458', 'ID00167637202237397919352',
      'ID00168837202239852027833', 'ID00169637202238024117706',
      'ID002296372022700354249580', 'ID00172637202238316925179',
      'ID00170837202238079193844', 'ID00180637202240177410333',
      'ID001734372022398329754031', 'ID00180637202240177410333',
      'ID0038863720221801804014195351050', 'ID00184637202242062969203',
      'ID00186637202242472088675', 'ID00190637202244450116191',
      'ID000526372022195619909961493238298', 'ID00196637202246668775836',
      'ID0034463720221976594017473', 'ID00199637202248141386743',
      'ID0018663720220763720225256380974', 'ID0020763720225256380974',
      'ID00267637202270399551595', 'ID00213637202257692916109',
      'ID000476372022180378092159820847190', 'ID00216637202257988213445',
      Name: PatientID, dtype: object, int64)

```

In [45]:

```

df1['PatientID'] = df1['PatientID'].astype(int)

```

Out[45]:

```

ID00105637202208831864134
ID00119637202210673668310
ID002296372022700354249580
ID0014063720221734372022398329754031
ID0038863720221801804014195351050
ID000526372022195619909961493238298
ID0034463720221976594017473
ID0018663720220763720225256380974
ID00267637202270399551595
ID000476372022180378092159820847190
Name: PatientID, dtype: object, int64
ID00009637202177434476278
ID00010637202177584971671
ID00011637202177653955184
ID00012637202177665765362
ID00014637202177757139317
ID00015637202177877247924
ID00019637202178323708467
ID00020637202178344345685
ID00023637202179104603099
ID00025637202179541264076
ID00026637202179561894768
ID00027637202179689871102
ID00030637202181211009029
ID00032637202181710233084
ID00035637202182204917484
ID00038637202182690843176
ID00042637202184406822975
ID00047637202184938901501
ID00048637202185016727717
ID00051637202185848464638
ID00052637202186188008618
ID00060637202187965290703
ID00061637202188184085559
ID00062637202188654068490
ID00067637202189903532242
ID00068637202190879923934
ID00072637202198161894406
ID00073637202198167792918
ID00075637202198610425520
ID00076637202199015035026
ID00077637202199102000916
ID00078637202199415319443
ID00082637202201836229724
ID00086637202203494931510
ID00089637202204675567570
ID00090637202204766623410
ID00093637202205278167493
ID00099637202206203808121
ID00104637202208063407045
ID00108637202209619669361
ID00109637202210454292264
ID00110637202210673668310
ID00111637202210956877205
ID00115637202211874187958
ID00117637202212360228007
ID00119637202215426335765
ID00122637202216437668965
ID00123637202217151272140
ID00125637202218590429387
ID00127637202219096738943
ID00129637202219868188000
ID00131637202220424084844
ID00133637202223847701934
ID00135637202224630271439
ID00138637202231603868088
ID00140637202231728595149
ID00161637202235731948764
ID00167637202237397919352
ID00169637202238024117706
ID00172637202238316925179
ID00180637202240177410333
ID00184637202242062969203
ID00190637202244450116191
ID00196637202246668775836
ID00199637202248141386743
ID0020763720225256380974
ID00213637202257692916109
ID00216637202257988213445
ID00219637202258203123958
ID00222637202259066229764
ID00225637202259339837603
ID00228637202259965313869
ID00232637202260377586117
ID00234637202261078001846
ID00235637202261451839085
ID00240637202264138860065
ID00241637202264294508775
ID00242637202264759739921
ID00243637202264759739921
ID00249637202266730854017
ID00251637202267455595113
ID00255637202267923028520
ID00264637202270643353440
ID00267637202270790561585
ID00273637202271319294586
ID00276637202271694539978
ID00279637202272164826258
ID00283637202278714365037
ID00285637202278913507108
ID00288637202279148973731
ID00290637202279304677843
ID00291637202279398396106
ID00294637202279614924243
ID00296637202279895784347

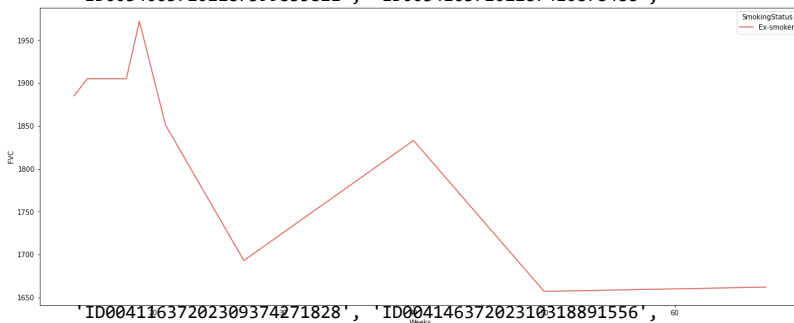
```



```

ID00298637202280361773446', 'ID00299637202280383305867',
In [46]: ID00305637202281772703145', 'ID00307637202282126172865',
ID00309637202282195513787', 'ID00312637202282607344793',
patient1 = df[df['Patient'] == 'ID00228637202259965313869']
ID0031763720228194142136', 'ID00319637202283897208687',
plt.figure(figsize=(20,8))
ID00322637202284842245491', 'ID00323637202285211956970',
sns.lineplot(data = patient1, y='FVC', x='Weeks', hue = 'SmokingStatus',
ID00328637202285906759848', 'ID00331637202286306023714',
plt.show()
ID00335637202286784464927', 'ID00336637202286801879145',
ID00337637202286839091062', 'ID00339637202287377736231',
ID00340637202287399835821', 'ID00341637202287410878488',

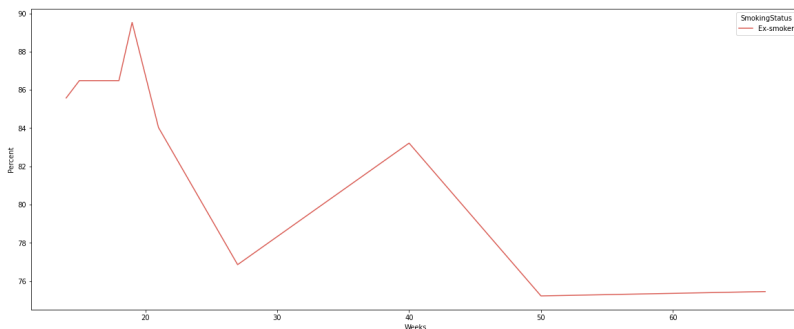
```



```

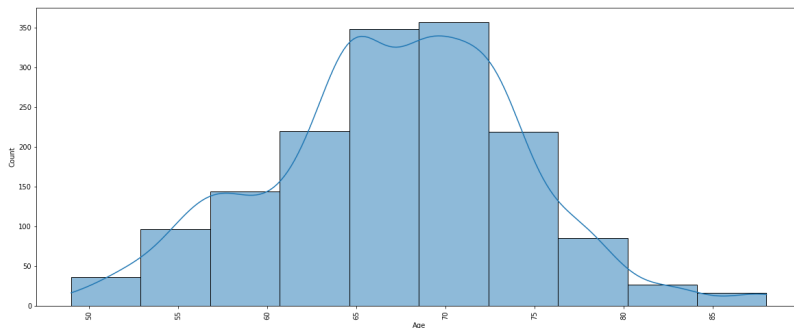
ID00411637202309374271828', 'ID00414637202310318891556',
ID00417637202310901214011', 'ID00419637202311204720264',
In [47]: ID00421637202311550012437', 'ID00422637202311677017371',
ID00423637202312137826377', 'ID00426637202313170790466'],
dtype=object)
patient1 = df[df['Patient'] == 'ID00228637202259965313869']
plt.figure(figsize=(20,8))
sns.lineplot(data = patient1, y='Percent', x='Weeks', hue = 'SmokingStatus',
palette = 'hls')
plt.show()

```



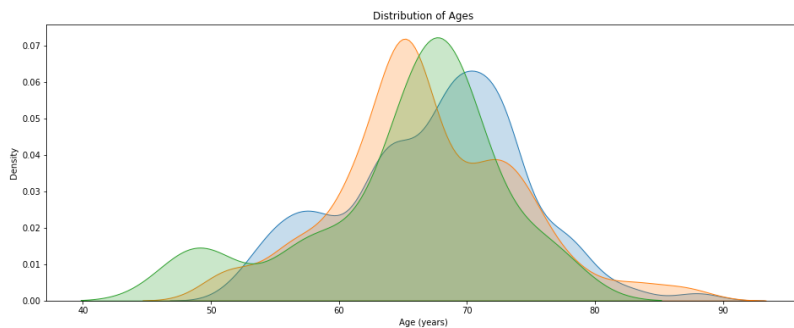
In [48]:

```
plt.figure(figsize=(20,8))
sns.histplot(df['Age'], bins = 10, kde = True, palette = 'hls')
plt.xticks(rotation = 90)
plt.show()
```



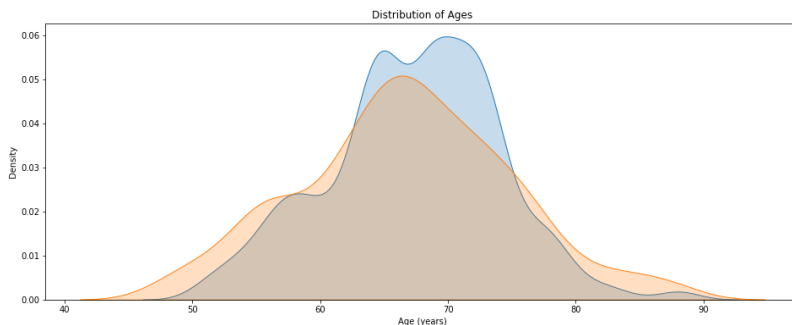
In [49]:

```
plt.figure(figsize=(16, 6))
sns.kdeplot(df1.loc[df1['SmokingStatus'] == 'Ex-smoker', 'Age'],
            label = 'Ex-smoker', shade=True)
sns.kdeplot(df1.loc[df1['SmokingStatus'] == 'Never smoked', 'Age'],
            label = 'Never smoked', shade=True)
sns.kdeplot(df1.loc[df1['SmokingStatus'] == 'Currently smokes', 'Age'],
            label = 'Currently smokes', shade=True)
plt.xlabel('Age (years)');
plt.ylabel('Density');
plt.title('Distribution of Ages');
plt.show()
```



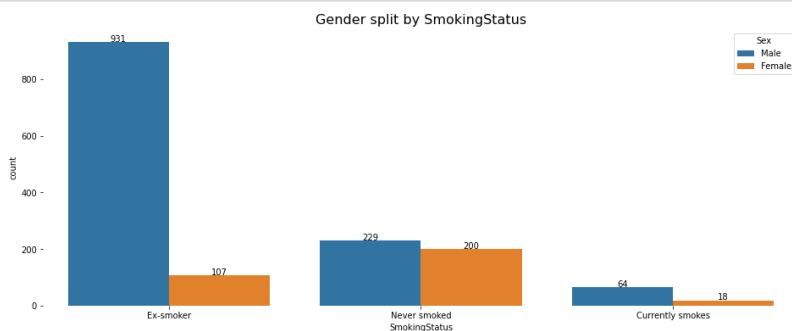
In [50]:

```
plt.figure(figsize=(16, 6))
sns.kdeplot(df1.loc[df1['Sex'] == 'Male', 'Age'], label = 'Male',shade=True)
sns.kdeplot(df1.loc[df1['Sex'] == 'Female', 'Age'], label = 'Female',shade=True)
plt.xlabel('Age (years)'); plt.ylabel('Density'); plt.title('Distribution of Ages');
plt.show()
```



In [51]:

```
plt.figure(figsize=(16, 6))
a = sns.countplot(data=df1, x='SmokingStatus', hue='Sex')
for p in a.patches:
    a.annotate(format(p.get_height(), ','),
               (p.get_x() + p.get_width() / 2.,
                p.get_height(), ha = 'center', va = 'center',
                xytext = (0, 4), textcoords = 'offset points'))
plt.title('Gender split by SmokingStatus', fontsize=16)
sns.despine(left=True, bottom=True);
```



In [52]:

```
corrmat = df.corr()  
f, ax = plt.subplots(figsize=(9, 8))  
sns.heatmap(corrmat, ax = ax, cmap = 'RdYlBu_r', linewidths = 0.5)  
plt.show()
```

