DSL1-C5_S5_Practice

In [1]:

```
import pandas as pd
import warnings
warnings.filterwarnings('ignore')
```

Task 1:

In [22]:

```
X1=55
X2 =80
zscore55=(X1-70)/11.35
zscore80=(X2-70)/11.35

print(zscore55)
print(zscore80)

p_value55 = norm.cdf(zscore55)
p_value88 = norm.cdf(zscore80)

print('zscore80','-', 'zscore55 =',p_value88- p_value55)
```

```
-1.3215859030837005
0.881057268722467
zscore80 - zscore55 = 0.7177035479448073
```

Task 1.2

In [18]:

```
X3 = 40
zscore40=(X3-70)/11.35
```

In [23]:

```
from scipy.stats import norm
p_value = norm.cdf(zscore40)
#prob_above80 = 1- p_value
print("the probability of less than 40 is :", p_value)
```

```
the probability of less than 40 is : 0.004106667373140424
```

Task 2

In [2]:

```
car_df = pd.read_csv(r'E:\Aishwarya official\Aishwarya Data Scince\course 5\DS1_C5_S5_Smart
car_df
```

Out[2]:

ickup_datetime	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude	passenger_c
2015-05-07 19:52:06 UTC	-73.999817	40.738354	-73.999512	40.723217	
2009-07-17 20:04:56 UTC	-73.994355	40.728225	-73.994710	40.750325	
2009-08-24 21:45:00 UTC	-74.005043	40.740770	-73.962565	40.772647	
2009-06-26 08:22:21 UTC	-73.976124	40.790844	-73.965316	40.803349	
2014-08-28 17:47:00 UTC	-73.925023	40.744085	-73.973082	40.761247	
...	...	...	...	...	
2012-10-28 10:49:00 UTC	-73.987042	40.739367	-73.986525	40.740297	
2014-03-14 01:09:00 UTC	-73.984722	40.736837	-74.006672	40.739620	
2009-06-29 00:42:00 UTC	-73.986017	40.756487	-73.858957	40.692588	
2015-05-20 14:56:25 UTC	-73.997124	40.725452	-73.983215	40.695415	
2010-05-15 04:08:00 UTC	-73.984395	40.720077	-73.985508	40.768793	

In [3]:

```
car_df.isnull().sum()
```

Out[3]:

```
Unnamed: 0           0
key                  0
fare_amount          0
pickup_datetime      0
pickup_longitude     0
pickup_latitude      0
dropoff_longitude    1
dropoff_latitude     1
passenger_count      0
dtype: int64
```

Task 2.c

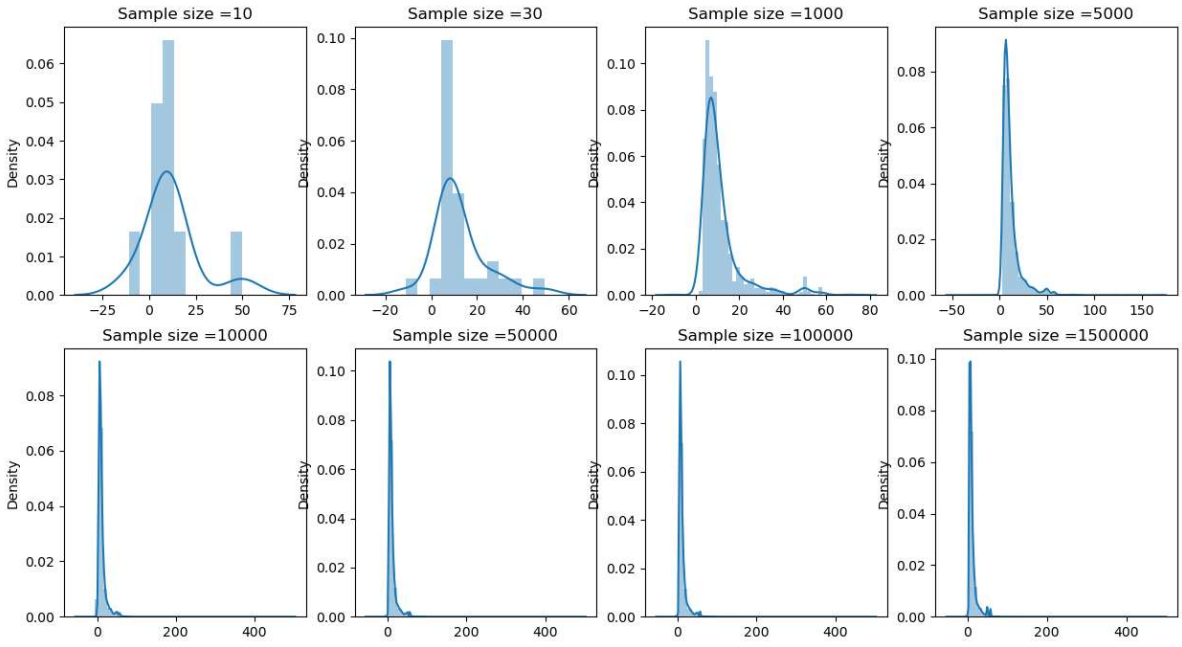
In [6]:

```
import seaborn as sns
num = [10,30,1000,5000,10000,50000,100000,1500000]
data_s = []
data_smean =[]
sample_df = pd.DataFrame()

for i in num :
    sample_df=car_df.sample(i, replace=True, random_state=1) #store each sample
    data_s.append(sample_df['fare_amount'].tolist())
    data_smean.append(sample_df['fare_amount'].mean())

fig, ax=plt.subplots(2,4, figsize=(15,8))

k=0
for i in range(0,2):
    for j in range(0,4):
        sns.distplot(data_s[k],ax=ax[i,j])
        ax[i,j].set_title(label='Sample size ='+ str(len(data_s[k])))
        k=k+1
plt.show()
```



Task 2.d

In [13]:

```
sample_df=pd.DataFrame()
for i in range(0,30): #take 20 random salaries
    sample_30 = pd.DataFrame(car_df['fare_amount'].sample(200, replace=True,ignore_index=Tr
    sample_df.insert(i,"Sample_"+ str(i+1),sample_30)
sample_df.head()
```

Out[13]:

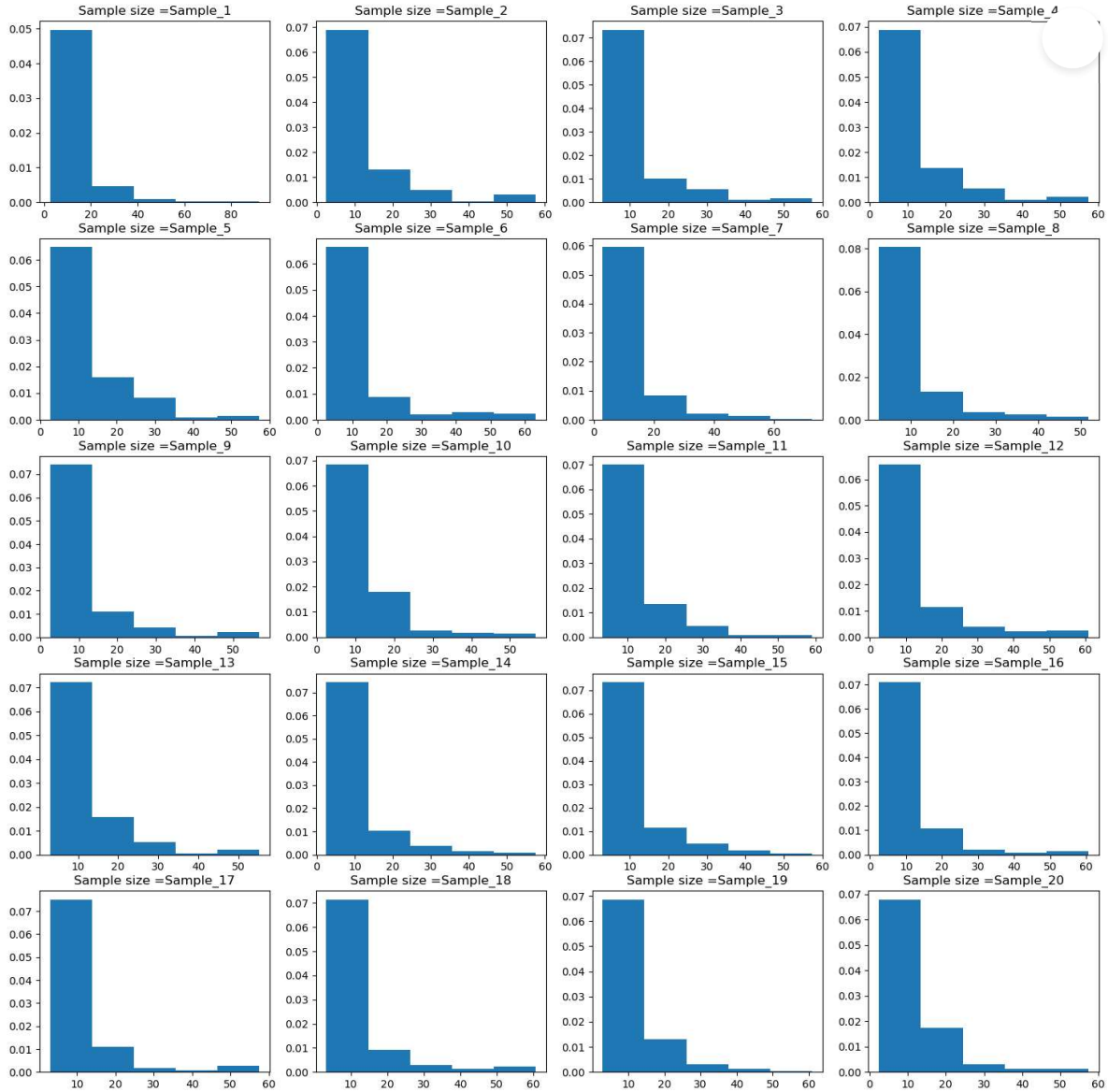
	Sample_1	Sample_2	Sample_3	Sample_4	Sample_5	Sample_6	Sample_7	Sample_8	San
0	7.70	7.5	8.0	7.70	8.0	9.0	22.0	21.5	
1	7.00	8.1	9.7	35.47	9.7	8.0	7.5	5.0	
2	7.70	6.0	14.5	4.50	25.5	6.5	8.5	3.0	
3	7.30	4.5	7.0	6.10	6.0	8.5	3.7	4.5	
4	49.57	6.9	8.5	12.10	34.9	7.7	12.5	12.5	

5 rows × 30 columns

In [14]:

```
sample_name = sample_df.columns
fig, ax=plt.subplots(5,4, figsize=(18,18))

k=0
for i in range(0,5):
    for j in range(0,4):
        ax[i,j].hist(sample_df[sample_name[k]],5,density=True)
        ax[i,j].set_title(label='Sample size ='+ sample_name[k])
        k=k+1
plt.show()
```



Task 2.e

In [10]:

```
import  scipy.stats as sts
import statistics as st
```

In [7]:

```
fare_amount= car_df['fare_amount']
```

In [8]:

```
samp1 = fare_amount.sample(10,replace=True , random_state=1)
samp2 = fare_amount.sample(30,replace=True , random_state=1)
samp3 = fare_amount.sample(50,replace=True , random_state=1)
```

In [11]:

```
tables=[samp1,samp2,samp3,fare_amount]
std=[]
mean=[]
mode=[]
median=[]
skew1=[]
kurt1=[]

for sample in tables:
    std.append(sample.std())
    mean.append(sample.mean())
    median.append(sample.median())
    mode.append(st.mode(sample))
    skew1.append(sample.skew())
    kurt1.append(sample.kurtosis())

pd.DataFrame([std,mean,median,mode,skew1,kurt1],
             columns=["Sample10_sal","Sample30_sal","Sample50_sal","Population"],index=["st
```

Out[11]:

	Sample10_sal	Sample30_sal	Sample50_sal	Population
std	15.120462	11.608228	11.533487	9.901776
Mean	11.687000	12.572333	12.272000	11.359955
Median	10.100000	8.700000	8.500000	8.500000
Mode	6.000000	6.000000	8.500000	6.500000
Skewness	1.699303	1.414679	2.126777	4.504847
Kurtosis	5.360300	3.054245	5.882639	63.884314

```
Iterpretation :-
```