Assignment-3

Problem Statement 1: In a survey conducted by a non-banking financial company, a sample of 200 customers yielded that x of them were highly satisfied with the timely disbursal of their loans.

Write a Python code to perform the following operations:

1. Read an integer input that specifies the number of highly satisfied customers
2. Calculate an approximate 90% confidence interval for the proportion of the loan customers who are highly satisfied with disbursal time

- Find out the Margin of Error using scipy.stats.norm.ppf
- Calculate and print the confidence interval values rounded up to five decimal places and separated by a space

Note:

- Margin of Error = Critical Value * Standard Error of Statistic
- Confidence Interval = Sample Statistic ± Margin of Error

Example: Let's say 172 out of 200 customers were highly satisfied.

```
Sample Input:
172
```

The confidence interval should be printed as -

```
Sample Output:
0.82856 0.89144
```

Problem Statement 2: A radar unit is used to measure the speeds of cars on a motorway. The speeds are normally distributed with a mean of 75 km/hr and a standard deviation of 15 km/hr.

Write a Python code to perform the following operations:

1. Find the probability that a car picked at random is traveling at more than X km/hr

- Take the speed X as an input
- Print the probability value rounded up to four decimal places

Hint:

Use Normal Distribution.

Sample Input:

```
100
```

Sample Output:

0.0478

Problem Statement 3: Write a Python program to load the *"kerala.csv"* data into a DataFrame and perform the following tasks:

1. Explore the DataFrame using info() and describe() functions
2. June and July are the peak months of rainfall. Consider that if it rains more than 500mm, then chances of flood become more; create a Datarame with columns – "YEAR", "JUN_GT_500" (Contains a boolean value to show whether it rained more than 500 mm in the month of June) , "JUL_GT_500" (Contains a boolean value to show whether it rained more than 500 mm in the month of July), and "FLOODS" (Contains a boolean value to show whether it flooded that year)
3. Calculate the probability of flood given it rained more than 500 mm in June (P(A|B))
4. Calculate the probability of rain more than 500 mm in June, given it flooded that year (P(B|A))
5. Probability of flood given it rained more than 500 mm in July
6. Probability of rain more than 500 mm in July given it flooded that year (P(B|A))

Sample Input:

	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL RAINFALL	FLOODS
0	KERALA	1901	28.7	44.7	51.6	160.0	174.7	824.6	743.0	357.5	197.7	266.9	350.8	48.4	3248.6	YES
1	KERALA	1902	6.7	2.6	57.3	83.9	134.5	390.9	1205.0	315.8	491.6	358.4	158.3	121.5	3326.6	YES
2	KERALA	1903	3.2	18.6	3.1	83.6	249.7	558.6	1022.5	420.2	341.8	354.1	157.0	59.0	3271.2	YES
3	KERALA	1904	23.7	3.0	32.2	71.5	235.7	1098.2	725.5	351.8	222.7	328.1	33.9	3.3	3129.7	YES
4	KERALA	1905	1.2	22.3	9.4	105.9	263.3	850.2	520.5	293.6	217.2	383.5	74.4	0.2	2741.6	NO

Sample Output:

1. June and July are the peak months of rainfall. Consider that if it rains more than 500mm, then chances of flood become more; create a Datarame with columns – *"YEAR", "JUN_GT_500", "JUL_GT_500",* and *"FLOODS"*

	YEAR	JUN_GT_500	JUL_GT_500	FLOODS	COUNT
0	1901	1	1	1	1
1	1902	0	1	1	1
2	1903	1	1	1	1
3	1904	1	1	1	1
4	1905	1	1	0	1

2. Calculate the probability of flood given it rained more than 500 mm in June (P(A|B))

```
Probailitity of flood given it rained more than 500 mm in June (P(A|B)):
P(Flood|June): 0.59375
```

3. Calculate the probability of rain more than 500 mm in June, given it flooded that year (P(B|A))

```
Probability of rain more than 500 mm in June given it flooded that year (P(B|A)):
P(June|Flood): 0.9000000000000001
```

4. Probability of flood given it rained more than 500 mm in July

```
Probabilitity of flood given it rained more than 500 mm in July:
P(Flood|July): 0.59375
```

5. Probability of rain more than 500 mm in July given it flooded that year (P(B|A))

```
Probability of rain more than 500 mm in July given it flooded that year (P(B|A)):
P(July|Flood): 0.9500000000000002
```

Problem Statement 4: Write a Python program to load the wine dataset using the Sklearn library to a DataFrame and perform the following tasks:

1. Convert the dataset into DataFrame using pandas.
2. Generate the sample size of 50 and give a random state as 100.
3. Calculate Z-critical, Margin of Error, and Confidence Interval for alcohol at 95% significance interval on generated sample data.

Sample Input:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	hue	od280/od315_of_diluted_wines	proline
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065.0
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050.0
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185.0
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480.0
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	0.52	1.06	7.70	0.64	1.74	740.0
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	0.43	1.41	7.30	0.70	1.56	750.0
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	0.43	1.35	10.20	0.59	1.56	835.0
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	0.53	1.46	9.30	0.60	1.62	840.0
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	0.56	1.35	9.20	0.61	1.60	560.0

178 rows × 13 columns

Sample Output:

```
Z-critical value: 1.6448536269514722
Margin of Error: 0.17880519784197366
Confidence Interval: (12.794794802158027, 13.152405197841976)
```