The Chai Point stall at Bengaluru airport estimates that each person visiting the store drinks an average of 1.7 small cups of tea.

Assume a population standard deviation of 0.5 small cups. A sample of 30 customers collected over a few days averaged 1.85 small cups of tea per person.

Test the claim using an appropriate test at an alpha = 0.05 significance value, with a critical z-score value of .

Note: Round off the z-score to two decimal places.

A) The computed z-score is 1.64, and since 1.64 is less than 1.96, the null hypothesis cannot be rejected.

B) The computed z-score is 1.64, and since 1.64 is less than 1.96, the null hypothesis is rejected.

C) The computed z-score is 2.33, and since 1.96 is less than 2.33, the null hypothesis cannot be rejected.

D) The computed z-score is 2.33, and since 1.96 is less than 2.33, the null hypothesis is rejected.

```
## H0:the average small number of cups that are cosnumed equals 1.7
## Ha: not equals 1.7
import pandas as pd
import numpy as np

# z=x-u/std

# z= 1.85-1.7/0.5/

sample_mean=1.85
population_mean=1.7
population_std=0.5
sample_size=30

# Calculate z score

z_score=(sample_mean-population_mean)/(population_std/np.sqrt(sample_size))
z_score=np.round(z_score,2)
print (z_score)
```

```
1.64
```

p_value approach

```
#P(Z< 1.64)
from scipy.stats import norm
1-norm.cdf(z_score)
```

```
np.float64(0.050502583474103746)
```

```
p_value=2*0.05
```

```
p_value
```

```
0.1
```

```
#is 0.1 <0.05?
```

The Zumba trainer claims to the customers, that their new dance routine helps to reduce more weight.

Weight of 8 people were recorded before and after following the new Zumba training for a month:

wt_before = [85, 74, 63.5, 69.4, 71.6, 65,90,78]

wt_after = [82, 71, 64, 65.2, 67.8, 64.7,95,77]

Test the trainer's claim with 90% confidence. Further, what would be the pvalue?

A) P value: 0.854, Customers did not reduce their weight

B) P value: 0.145, Customers did not reduce their weight

C) P value: 0.854, Customers have reduced their weight

D) P value: 0.145, Customers have reduced their weight

```
from os import terminal_size
# Ho: Customer did not reduce their weight average weight before and remains same
# Ha: Customer have reduce their weight ubefore > uafter
from scipy.stats import ttest_rel
wt_before = [85, 74, 63.5, 69.4, 71.6, 65,90,78]
```

```
wt_after = [82, 71, 64, 65.2, 67.8, 64.7,95,77]
alpha=0.10
t_stat,p_value=ttest_rel(wt_before,wt_after)
print(t_stat,p_value)
```

```
1.1421853793555032 0.2909361700265277
```

```
p_value =0.29/2
```

```
p_value
```

```
0.145
```

```
if p_value < alpha:
  print("Reject null hypothesis")
else:
  print("Fail to reject null hypothesis")
```

```
Fail to reject null hypothesis
```

```
from os import terminal_size
# Ho: Customer did not reduce their weight average weight before and remains same
# Ha: Customer have reduce their weight ubefore > uafter
from scipy.stats import ttest_rel
wt_before = [85, 74, 63.5, 69.4, 71.6, 65,90,78]

wt_after = [82, 71, 64, 65.2, 67.8, 64.7,95,77]
alpha=0.10
t_stat,p_value=ttest_rel(wt_before,wt_after,alternative="greater")
print(t_stat,p_value)
```

```
1.1421853793555032 0.14546808501326386
```

The quality assurance department claims that on average the non-fat milk contains more than 190 mg of Calcium per 500 ml packet.

To check this claim 45 packets of milk are collected and the content of calcium is recorded.

Perform an appropriate test to check the claim with a 90% confidence level.

data = [193, 321, 222, 158, 176, 149, 154, 223, 233, 177, 280, 244, 138, 210, 167, 129, 254, 167, 194, 191, 128, 191, 144, 184, 330, 216, 212, 142, 216, 197, 231, 133, 205, 192, 195, 243, 224, 137, 234, 171, 176, 249, 222, 234, 191]

Note: Round off the answer to four decimal places.

A)Test statistic: 1.3689 , Reject null hypothesis

B) Test statistic: 1.3689 , Fail to reject null hypothesis

C) Test statistic: 1.2851, Reject null hypothesis

D) Test statistic: 1.2851 , Fail to reject null hypothesis

```
#Ha: u > 190
#H0 : u <=190

from scipy.stats import ttest_1samp
data=pd.Series([193, 321, 222, 158, 176, 149, 154, 223, 233, 177, 280, 244, 138, 210, 167, 129, 254, 167, 194, 1
print("Obs sample mean",round(data.mean(),2))

t_stat,p_value=ttest_1samp(data,190,alternative="greater")
print(t_stat,p_value)
if p_value < 0.10:
  print("Reject null hypothesis")
else:
  print("Fail to reject null hypothesis")
```

```
Obs sample mean 199.49
1.3689029903414232 0.08898891556150607
Reject null hypothesis
```

Child development researchers studying growth patterns of children collect data on the height of fathers and sons.

Analyse the correlation between the father's height and their son's height using the given data

Father Height = [169.39, 161.91, 159.23, 161.72, 167.52, 152.13, 169.64, 162.56, 154.92, 158.57, 153.17, 159.56, 153.77, 168.02, 157.75, 157.42, 160.65, 160.09, 151.4, 151.05, 136.94, 163.56, 160.39, 146.92, 171.66, 150.48, 158.12, 157.83, 163.99, 164.95]

Son Height = [187.35, 177.8, 181.85, 190.69, 188.07, 168.16, 181.65, 173.94, 174.28, 177.87, 176.01, 185.18, 180.33, 175.85, 178.11, 177.34, 185.46, 173.56, 177.19, 169.02, 157.13, 179.58, 181.05, 169.8, 190.89, 164.82, 175.32, 173.69, 185.73, 185.29]

A) Negative Correlation

B) Positive Correlation

C) No Correlation

D) Correlation coefficient > 1

```
Father_Height = [169.39, 161.91, 159.23, 161.72, 167.52, 152.13, 169.64, 162.56, 154.92, 158.57, 153.17, 159.56,
```

```
Son_Height = [187.35, 177.8, 181.85, 190.69, 188.07, 168.16, 181.65, 173.94, 174.28, 177.87, 176.01, 185.18, 180
```

```
df=pd.DataFrame({"Father_Height":Father_Height,"Son_Height":Son_Height})
```

```
df.head()
```

	Father_Height	Son_Height
0	169.39	187.35
1	161.91	177.80
2	159.23	181.85
3	161.72	190.69
4	167.52	188.07

Next steps: Generate code with df New interactive sheet

```
df.corr()
```

	Father_Height	Son_Height
Father_Height	1.000000	0.800205
Son_Height	0.800205	1.000000

A researcher is investigating the distribution of response times (in seconds) for two different versions of a mobile app, i.e. the time taken for a mobile app to respond to a user action, measured in seconds.

The goal is to determine if the response time distributions significantly differ between the two versions.

Data for 20 users for each app version is collected.

response_times_version_A = [1.2, 1.3, 1.1, 1.4, 1.2, 1.3, 1.0, 1.5, 1.2, 1.3, 1.2, 1.4, 1.1, 1.3, 1.2, 1.5, 1.3, 1.4, 1.2, 1.3]

response_times_version_B = [1.6, 1.2, 1.3, 1.4, 1.1, 1.3, 1.2, 1.5, 1.3, 1.4, 1.2, 1.3, 1.2, 1.4, 1.1, 1.3, 1.5, 1.2, 1.3, 1.4] Choose the appropriate test for the given scenario

A) One -Way ANOVA

B) Two Sample Z Proportion

C) Paired T-Test

D) Two-Sample Z-Test

E) KS Test

```
Start coding or generate with AI.
```