```
import pandas as pd
import numpy as np
import scipy.stats as stats


df = pd.read_csv('/content/Mall_Customers.csv')
df
```

	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40
...	...	...	...	...	...
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

200 rows × 5 columns

Next steps: Generate code with df | View recommended plots | New interactive sheet

2. Display summary statistics for each column

```
df.describe()
```

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
count	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000
std	57.879185	13.969007	26.264721	25.823522
min	1.000000	18.000000	15.000000	1.000000
25%	50.750000	28.750000	41.500000	34.750000
50%	100.500000	36.000000	61.500000	50.000000
75%	150.250000	49.000000	78.000000	73.000000
max	200.000000	70.000000	137.000000	99.000000

3. Display Measures of Dispersion

```
numeric_df = df.select_dtypes(include=['float64', 'int64'])

dispersion_measures = {
    "Mean Absolute Deviation": numeric_df.mean(),
    "Variance": numeric_df.var(),
    "Standard Deviation": numeric_df.std(),
    "Range": numeric_df.max() - numeric_df.min(),
    "1st Quartile": numeric_df.quantile(0.25),
    "3rd Quartile": numeric_df.quantile(0.75),
    "Skewness": numeric_df.skew()
}

dispersion_df = pd.DataFrame(dispersion_measures)
dispersion_df
```

	Mean Absolute Deviation	Variance	Standard Deviation	Range	1st Quartile	3rd Quartile	Skewness
CustomerID	100.50	3350.000000	57.879185	199	50.75	150.25	0.000000
Age	38.85	195.133166	13.969007	52	28.75	49.00	0.485569
Annual	60.56	689.835578	26.264721	122	41.50	78.00	0.321843

Next steps: Generate code with dispersion_df | View recommended plots | New interactive sheet

4. Summary statistics of income grouped by age groups (categorical vs quantitative)

```
income_summary = df.groupby("Age")["Annual Income (k$)"].agg(["min", "max"])
income_summary.head()
```

Age	min	max
18	33	65
19	15	81
20	16	73
21	15	62
22	17	57

Next steps: Generate code with `income_summary` View recommended plots New interactive sheet

Start coding or generate with AI.