Simulacion de Monte Carlo

```
1 import pandas as pd
2 import numpy as np
3 import seaborn as sns
```

```
1 sns.set_style('whitegrid')
```

```
1 # Definir las variables de porcentajes
2 avg = 1
3 std_dev = .1
4 num_reps = 500
5 num_simulations = 1000
```

```
1 # Generar datos
2
3 pct_to_target = np.random.normal(avg, std_dev, num_reps).round(2)
```

```
1 pct_to_target[0:20]
```

```
array([0.97, 1.05, 0.94, 0.82, 1.05, 0.85, 0.88, 1.1 , 1.08, 1.16, 1.04,
       1.05, 0.68, 0.95, 0.99, 0.99, 1.1 , 1.01, 1.16, 1.01])
```

```
1 # Mas datos
2 sales_target_values = [75_000, 100_000, 200_000, 300_000 ,400_000, 500_000]
3 sales_target_prob = [.3, .3, .2 , .1, .05, .05]
4 sales_target = np.random.choice(sales_target_values, num_reps, p=sales_targe
```

```
1 sales_target[0:3]
```

```
array([300000,  75000,  75000])
```

```
1 df = pd.DataFrame(index=range(num_reps), data={'PCT_To_Target': pct_to_targe
2 df.head()
```

	PCT_To_Target	SALES_Target
0	0.97	300000
1	1.05	75000
2	0.94	75000
3	0.82	500000
4	1.05	75000

Next steps: Generate code with df View recommended plots

```
1 df['PCT_To_Target'].plot(kind='hist', title='Distribucion Historica')
```

```
<Axes: title={'center': 'Distribucion Historica'}, ylabel='Frequency'>
```