# TASK 4

Deploy detailed Python code to strategically bucket customers with various FICO scores in order to narrow in on the probability of default

# OBJECTIVE

* How to apply statistical formulas to business solutions
* The importance of breaking down a large dataset using machine learning methods

# CODE

# A simple program to analyze loan data

import pandas as pd

import os

# Print current working directory

print("Current working directory:", os.getcwd())

# Change to the directory where the CSV file is located

# You need to replace "\_\_\_\_\_\_\_\_" with the actual path

os.chdir("\_\_\_\_\_\_\_\_")

# Read the CSV file

data = pd.read\_csv('loan\_data\_created.csv')

# Get the 'default' and 'fico\_score' columns as lists

defaults = data['default'].tolist()

fico\_scores = data['fico\_score'].tolist()

print("Number of entries:", len(defaults))

# Initialize lists to store default counts and total counts for each FICO score

default\_counts = [0] \* 551 # FICO scores range from 300 to 850

total\_counts = [0] \* 551

# Count defaults and totals for each FICO score

for i in range(len(defaults)):

score\_index = int(fico\_scores[i]) - 300

default\_counts[score\_index] += defaults[i]

total\_counts[score\_index] += 1

# Calculate cumulative sums

for i in range(1, 551):

default\_counts[i] += default\_counts[i-1]

total\_counts[i] += total\_counts[i-1]

# Calculate default rates for each FICO score

default\_rates = []

for i in range(551):

if total\_counts[i] > 0:

rate = default\_counts[i] / total\_counts[i]

default\_rates.append((i + 300, rate))

# Sort default rates and find the top 5 FICO scores with highest default rates

top\_5\_default\_rates = sorted(default\_rates, key=lambda x: x[1], reverse=True)[:5]

print("\nTop 5 FICO scores with highest default rates:")

for score, rate in top\_5\_default\_rates:

print(f"FICO Score: {score}, Default Rate: {rate:.2%}")

# Calculate overall default rate

overall\_default\_rate = sum(defaults) / len(defaults)

print(f"\nOverall default rate: {overall\_default\_rate:.2%}")