```python
# supply_chain_optimization.py

import pandas as pd
import numpy as np

# 1. Supplier Data
supplier_df = pd.DataFrame({
    'Supplier': ['A', 'B', 'C'],
    'Avg_Lead_Time': [7, 10, 6],
    'Std_Dev_Lead_Time': [2.0, 5.0, 1.5],
    'Cost_per_Unit': [8000, 7300, 8200],
    'Defect_Rate_%': [2.5, 6.2, 1.8],
    'SLA_Adherence_%': [96, 81, 93]
})

# 2. Simulated 6-month Inventory Data
inventory_df = pd.DataFrame({
    'Month': ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun'],
    'Demand': [120, 130, 140, 135, 120, 125],
    'Opening_Stock': [150, 130, 80, 50, 15, 15],
    'Received_from_Supplier': [100, 80, 50, 100, 120, 110],
})

inventory_df['Closing_Stock'] = inventory_df['Opening_Stock'] + inventory_df['Received_from_Supplier'] - inventory_df['Demand']
inventory_df['Stockout'] = inventory_df['Closing_Stock'].apply(lambda x: 'Yes' if x <= 0 else 'No')

# 3. Holding Cost Calculation
HOLDING_COST_PER_UNIT = 80  # □ per unit/month
inventory_df['Holding_Cost'] = inventory_df['Closing_Stock'].apply(lambda x: max(x, 0) * HOLDING_COST_PER_UNIT)

# 4. Safety Stock Calculation
Z_scores = {'A': 1.65, 'B': 1.28, 'C': 1.28}
avg_daily_demand = 5  # assumed constant
```

```python

safety_stock_df = pd.DataFrame({
    'Supplier': ['A', 'B', 'C'],
    'Z': [Z_scores['A'], Z_scores['B'], Z_scores['C']],
    'Std_Dev_Lead_Time': [2.0, 5.0, 1.5],
    'Avg_Daily_Demand': avg_daily_demand
})

safety_stock_df['Safety_Stock'] = (safety_stock_df['Z'] * safety_stock_df['Std_Dev_Lead_Time'] * safety_stock_df['Avg_Daily_Demand']).round().astype(int)

# 5. Summary Outputs
print("\n--- Supplier Summary ---")
print(supplier_df)

print("\n--- Inventory Simulation ---")
print(inventory_df[['Month', 'Demand', 'Opening_Stock', 'Received_from_Supplier', 'Closing_Stock', 'Stockout', 'Holding_Cost']])

print("\n--- Safety Stock per Supplier ---")
print(safety_stock_df[['Supplier', 'Safety_Stock']])

print("\n--- Key Results ---")
avg_holding_cost = inventory_df['Holding_Cost'].mean()
stockout_events = inventory_df[inventory_df['Stockout'] == 'Yes'].shape[0]

print(f"Average Monthly Holding Cost: □{avg_holding_cost:.0f}")
print(f"Stockout Events (6 months): {stockout_events}")
print("Preferred Suppliers: A (1st), C (2nd) based on SLA and defect rate.")

```