**Lab Exercise 8 - Writing Data into an Excel File Using openpyxl in RPA with Python**

**Objective**

This lab exercise demonstrates how to use the openpyxl library to create and write data into an Excel file programmatically, a critical skill in Robotic Process Automation (RPA) workflows.

**Objective**

1. Create a new Excel file.
2. Write data into specific cells.
3. Format the Excel file for better readability.

**Prerequisites**

1. Install the openpyxl library:

pip install openpyxl

**Step 1: Create and Write Basic Data**

from openpyxl import Workbook

# Create a new workbook

workbook = Workbook()

# Select the active worksheet

sheet = workbook.active

# Rename the sheet

sheet.title = "Employee Data"

# Write headers to the Excel file

headers = ["Employee ID", "Name", "Department", "Salary"]

sheet.append(headers)

# Write some sample data

data = [

[101, "Alice", "HR", 50000],

[102, "Bob", "IT", 70000],

[103, "Charlie", "Finance", 60000],

[104, "Diana", "Marketing", 65000]

]

for row in data:

sheet.append(row)

# Save the workbook

workbook.save("employee\_data.xlsx")

print("Excel file 'employee\_data.xlsx' created successfully!")

**Step 2: Add Data to Specific Cells**

# Reopen the workbook to add more data

workbook = Workbook()

sheet = workbook.active

# Add a new employee record in specific cells

sheet["A6"] = 105

sheet["B6"] = "Eve"

sheet["C6"] = "Sales"

sheet["D6"] = 55000

# Save the workbook

workbook.save("employee\_data\_updated.xlsx")

print("New data added and saved in 'employee\_data\_updated.xlsx'")

**Step 3: Apply Formatting**

from openpyxl.styles import Font, Alignment

# Load the workbook

workbook = Workbook()

sheet = workbook.active

# Format the header row

header\_font = Font(bold=True, color="FFFFFF")

header\_alignment = Alignment(horizontal="center")

for cell in sheet[1]:

cell.font = header\_font

cell.alignment = header\_alignment

cell.fill = openpyxl.styles.PatternFill(start\_color="0000FF", end\_color="0000FF", fill\_type="solid")

# Save the formatted workbook

workbook.save("formatted\_employee\_data.xlsx")

print("Formatted Excel file saved as 'formatted\_employee\_data.xlsx'")

**Step 4: Add Formulas**

# Add a formula to calculate total salary

sheet["E1"] = "Total Salary"

sheet["E2"] = "=SUM(D2:D6)"

# Save the workbook

workbook.save("employee\_data\_with\_formula.xlsx")

print("Excel file with formula saved as 'employee\_data\_with\_formula.xlsx'")

**Expected Output**

1. **employee\_data.xlsx**: Contains employee data with headers.
2. **employee\_data\_updated.xlsx**: New data added to specific cells.
3. **formatted\_employee\_data.xlsx**: Contains formatting for headers.
4. **employee\_data\_with\_formula.xlsx**: Includes a formula to calculate the total salary.