11. ZIP OPERATION ON FOLDER

```
import shutil

# 1. Backup photos folder
shutil.make_archive("photos_backup", 'zip', "photos")

# 2. Zip project folder
shutil.make_archive("project_zip", 'zip', "project")

# 3. Zip lecture notes
shutil.make_archive("lecture_notes", 'zip', "lecture_notes")
```

✅ Creates .zip files automatically.

12. INHERITANCE – AREAS

```
import math

class Shape:
    def area(self):
        pass

class Triangle(Shape):
    def __init__(self, base, height):
        self.base = base
        self.height = height
    def area(self):
        return 0.5 * self.base * self.height

class Circle(Shape):
    def __init__(self, radius):
        self.radius = radius
    def area(self):
        return math.pi * self.radius ** 2

class Rectangle(Shape):
    def __init__(self, length, width):
        self.length = length
        self.width = width
    def area(self):
        return self.length * self.width

# Example
```

15. SPREADSHEET OPERATIONS (USING OPENPYXL)

```
from openpyxl import Workbook, load_workbook

# 1. Read marks & write average
wb = Workbook()
ws = wb.active
ws.append(["Name", "Mark1", "Mark2", "Mark3"])
ws.append(["Ravi", 80, 90, 85])
ws.append(["Anita", 70, 75, 80])

for row in ws.iter_rows(min_row=2, max_col=4):
    marks = [cell.value for cell in row[1:]]
    avg = sum(marks) / len(marks)
    ws.cell(row=row[0].row, column=5, value=avg)

wb.save("marks.xlsx")

# 2. Store sales and update totals
wb = Workbook()
ws = wb.active
ws.append(["Product", "Qty", "Price", "Total"])
ws.append(["Laptop", 2, 50000, None])
ws.append(["Mouse", 5, 500, None])

for row in ws.iter_rows(min_row=2, max_col=4):
    qty, price = row[1].value, row[2].value
    ws.cell(row=row[0].row, column=4, value=qty * price)

wb.save("sales.xlsx")

# 3. Copy sheet
wb = Workbook()
ws1 = wb.active
ws1.title = "Original"
ws1.append(["A", "B", "C"])

ws2 = wb.create_sheet("Copy")
for row in ws1.iter_rows(values_only=True):
    ws2.append(row)

wb.save("copy.xlsx")
```

11. ZIP OPERATION ON FOLDER

```
import shutil

# 1. Backup photos folder
shutil.make_archive("photos_backup", 'zip', "photos")

# 2. Zip project folder
shutil.make_archive("project_zip", 'zip', "project")

# 3. Zip lecture notes
shutil.make_archive("lecture_notes", 'zip', "lecture_notes")
```

✅ Creates .zip files automatically.

12. INHERITANCE – AREAS

```
import math

class Shape:
    def area(self):
        pass

class Triangle(Shape):
    def __init__(self, base, height):
        self.base = base
        self.height = height
    def area(self):
        return 0.5 * self.base * self.height

class Circle(Shape):
    def __init__(self, radius):
        self.radius = radius
    def area(self):
        return math.pi * self.radius ** 2

class Rectangle(Shape):
    def __init__(self, length, width):
        self.length = length
        self.width = width
    def area(self):
        return self.length * self.width

# Example
```

15. SPREADSHEET OPERATIONS (USING OPENPYXL)

```
from openpyxl import Workbook, load_workbook

# 1. Read marks & write average
wb = Workbook()
ws = wb.active
ws.append(["Name", "Mark1", "Mark2", "Mark3"])
ws.append(["Ravi", 80, 90, 85])
ws.append(["Anita", 70, 75, 80])

for row in ws.iter_rows(min_row=2, max_col=4):
    marks = [cell.value for cell in row[1:]]
    avg = sum(marks) / len(marks)
    ws.cell(row=row[0].row, column=5, value=avg)

wb.save("marks.xlsx")

# 2. Store sales and update totals
wb = Workbook()
ws = wb.active
ws.append(["Product", "Qty", "Price", "Total"])
ws.append(["Laptop", 2, 50000, None])
ws.append(["Mouse", 5, 500, None])

for row in ws.iter_rows(min_row=2, max_col=4):
    qty, price = row[1].value, row[2].value
    ws.cell(row=row[0].row, column=4, value=qty * price)

wb.save("sales.xlsx")

# 3. Copy sheet
wb = Workbook()
ws1 = wb.active
ws1.title = "Original"
ws1.append(["A", "B", "C"])

ws2 = wb.create_sheet("Copy")
for row in ws1.iter_rows(values_only=True):
    ws2.append(row)

wb.save("copy.xlsx")
```