Problem Definition: Transportation Optimization

A company needs to transport goods from **three warehouses** (W1, W2, W3) to **four retail stores** (S1, S2, S3, S4) while **minimizing transportation costs**. Each warehouse has a **limited supply**, and each store has a **specific demand**. The transportation cost per unit between each warehouse and store is given.

1. Transportation Cost (in $ per unit)

The cost to transport one unit of goods from a warehouse to a store is given in the table below:

	S1	**S2**	**S3**	**S4**
W1	4	8	8	6
W2	6	7	3	5
W3	9	4	7	3

For example, the cost of transporting one unit from **W1 to S1** is **$4**, while from **W3 to S2** is **$4**.

Mathematical Formulation:

Decision Variables:

Let x_{ij} represent the **number of units transported from warehouse i to store j**:

- x_{11} = Units transported from **W1 → S1**
- x_{12} = Units transported from **W1 → S2**
- x_{13} = Units transported from **W1 → S3**
- x_{14} = Units transported from **W1 → S4**
- ... (similar variables for W2 and W3)

Objective Function:

Minimize total transportation cost:

$$\min Z = 4x_{11} + 8x_{12} + 8x_{13} + 6x_{14} + 6x_{21} + 7x_{22} + 3x_{23} + 5x_{24} + 9x_{31} + 4x_{32} + 7x_{33} + 3x_{34}$$

Constraints:

1. **Supply Constraints (Each warehouse has a limited supply)**:
   - $x_{11} + x_{12} + x_{13} + x_{14} \leq 100$ (Supply from W1)
   - $x_{21} + x_{22} + x_{23} + x_{24} \leq 120$ (Supply from W2)
   - $x_{31} + x_{32} + x_{33} + x_{34} \leq 130$ (Supply from W3)
2. **Demand Constraints (Each store must receive the required quantity)**:
   - $x_{11} + x_{21} + x_{31} = 80$ (Demand for S1)
   - $x_{12} + x_{22} + x_{32} = 70$ (Demand for S2)
   - $x_{13} + x_{23} + x_{33} = 90$ (Demand for S3)
   - $x_{14} + x_{24} + x_{34} = 110$ (Demand for S4)
3. **Non-Negativity Constraints**:
   $x_{ij} \geq 0 \quad \forall i, j$

Brief Explanation of the Code

This Python script solves a **transportation optimization problem** using **Linear Programming (LP)** with the **PuLP** library. The goal is to **minimize transportation costs** while satisfying supply and demand constraints.

Libraries Used:

- **pulp**: A Python library for Linear Programming (LP) and Mixed-Integer Programming (MIP).

Key Functions & Their Purpose:

1. **LpProblem("Transportation_Optimization", LpMinimize)**
   - Creates an LP problem named *"Transportation_Optimization"* with a **minimization** objective.
2. **LpVariable(f"x_{i}_{j}", lowBound=0)**
   - Defines decision variables x_{i,j}, representing **units transported** from warehouse i to store j.
   - `lowBound=0` ensures variables are **non-negative**.

3. **Objective Function:**

```
model += sum(x[i, j] for i in range(1, 4)) == demand[j], f"Demand_Constraint_S{j}"
```

Minimizes **total transportation cost** using a summation of cost per unit multiplied by transported units.

4. **Constraints:**
   - **Supply Constraints** ensure each warehouse does not exceed its available supply.
   - **Demand Constraints** ensure each store receives exactly its required amount.
5. **model.solve()**
   - Solves the LP problem using an appropriate solver.
6. **Printing Results:**
   - Displays **optimal transportation plan** and **minimum cost**.
7. **Sensitivity Analysis (Shadow Prices & Slack):**

```
# Sensitivity Analysis (Shadow Prices)
print("\nSensitivity Analysis (Shadow Prices):")
for constraint in model.constraints:
    print(f"{constraint}: Shadow Price = {model.constraints[constraint].pi}, Slack = {model.constraints[constraint].slack}")
```

•**Shadow price**: Impact of a unit increase in supply/demand.

•**Slack**: Unused capacity in a constraint.

Results:

GLPSOL--GLPK LP/MIP Solver 5.0

Parameter(s) specified in the command line:

--cpxlp /var/folders/4n/6ydjnj8d16x_m4by160ndjw80000gn/T/d69e0c74262d456d9ad4c475dfcc2757-pulp.lp

-o /var/folders/4n/6ydjnj8d16x_m4by160ndjw80000gn/T/d69e0c74262d456d9ad4c475dfcc2757-pulp.sol

Reading problem data from '/var/folders/4n/6ydjnj8d16x_m4by160ndjw80000gn/T/d69e0c74262d456d9ad4c475dfcc2757-pulp.lp'...

7 rows, 12 columns, 24 non-zeros

13 lines were read

GLPK Simplex Optimizer 5.0

7 rows, 12 columns, 24 non-zeros

Preprocessing...

7 rows, 12 columns, 24 non-zeros

Scaling...

A: min|aij| = 1.000e+00 max|aij| = 1.000e+00 ratio = 1.000e+00

Problem data seem to be well scaled

Constructing initial basis...

Size of triangular part is 7

0: obj = 1.960000000e+03 inf = 2.200e+02 (1)

4: obj = 1.510000000e+03 inf = 0.000e+00 (0)

* 8: obj = 1.320000000e+03 inf = 0.000e+00 (0)

OPTIMAL LP SOLUTION FOUND

Time used: 0.0 secs

Memory used: 0.0 Mb (33757 bytes)

Writing basic solution to '/var/folders/4n/6ydjnj8d16x_m4by160ndjw80000gn/T/d69e0c74262d456d9ad4c475dfcc2757-pulp.sol'...

Optimal Transportation Plan:

Warehouse 1 to Store 1: 80.0 units

Warehouse 1 to Store 2: 0.0 units

Warehouse 1 to Store 3: 0.0 units

Warehouse 1 to Store 4: 20.0 units

Warehouse 2 to Store 1: 0.0 units

Warehouse 2 to Store 2: 0.0 units

Warehouse 2 to Store 3: 90.0 units

Warehouse 2 to Store 4: 30.0 units

Warehouse 3 to Store 1: 0.0 units

Warehouse 3 to Store 2: 70.0 units

Warehouse 3 to Store 3: 0.0 units

Warehouse 3 to Store 4: 60.0 units

Minimum Transportation Cost: $1320.0

Sensitivity Analysis (Shadow Prices):

Supply_Constraint_W1: Shadow Price = None, Slack = None

Supply_Constraint_W2: Shadow Price = None, Slack = None

Supply_Constraint_W3: Shadow Price = None, Slack = None

Demand_Constraint_S1: Shadow Price = None, Slack = None

Demand_Constraint_S2: Shadow Price = None, Slack = None

Demand_Constraint_S3: Shadow Price = None, Slack = None

Demand_Constraint_S4: Shadow Price = None, Slack = None