Awesome 🚀 Let’s build a **Daily Breakdown for Week 1** so you can start strong.  
Week 1 is all about **Graph Basics & Representations**.

**📅 Week 1 – Graph Foundations (Daily Plan)**

**Day 1 – Graph Basics**

* 📖 Study:
  + What is a graph? (nodes, edges, vertices).
  + Types of graphs (directed vs undirected, weighted vs unweighted, cyclic vs acyclic).
* 📝 Task:
  + Write definitions in your own words.
  + Draw a small directed graph & undirected graph on paper.
* 💻 Code:
* # adjacency list
* graph = {
* 0: [1, 2],
* 1: [2],
* 2: [0, 3],
* 3: [3]
* }
* print(graph)
* 🎯 Output: Understand how nodes connect.

**Day 2 – Graph Representation**

* 📖 Study:
  + Adjacency List vs Adjacency Matrix.
  + When to use which? (Sparse vs Dense graph).
* 💻 Code:
  + Build adjacency **list** from an edge list.
  + Build adjacency **matrix** from an edge list.
* edges = [(0,1), (0,2), (1,2), (2,0), (2,3), (3,3)]
* 🎯 Practice:
  + Convert the above edge list into both representations.

**Day 3 – Hands-On Graph Building**

* 📝 Task:
  + Implement a simple **Graph class** with:
    - add\_edge(u, v)
    - print\_graph()
  + Support both **directed** and **undirected**.
* 🎯 Practice Problem:
  + HackerRank – *Roads and Libraries* (just read problem statement & understand representation).

**Day 4 – Intro Problems (Easy)**

* 🧩 Solve:
  + LeetCode 1971 – *Find if Path Exists in Graph* (undirected graph, BFS/DFS).
  + LeetCode 841 – *Keys and Rooms* (DFS).
* 💡 Hint: Just use adjacency list + DFS/BFS.

**Day 5 – Grid as Graph**

* 📖 Study:
  + How 2D grids are just graphs in disguise.
* 💻 Code:
  + Write a DFS to explore all connected cells in a grid.
* 🧩 Solve:
  + LeetCode 200 – *Number of Islands*.

**Day 6 – Mixed Practice**

* 🧩 Solve:
  + LeetCode 695 – *Max Area of Island*.
  + Practice converting problems from **grid → graph representation**.

**Day 7 – Review & Recap**

* 📝 Task:
  + Write summary notes:
    - Graph definitions
    - Adjacency list vs matrix
    - DFS/BFS basics (just high-level idea)
* 🎯 Mini Quiz (self-check):
  + When to use adjacency list vs matrix?
  + How to build graph from edge list?
  + What’s the space complexity difference?

✅ By the end of Week 1:

* You’ll be **very comfortable building and representing graphs**.
* You’ll have solved 3–4 real LeetCode problems.
* You’ll be ready to jump into **DFS & BFS deeply in Week 2**.

Would you like me to also **make a Python template** for graph problems (with adjacency list + DFS + BFS functions) so you can reuse it every week?