Blender + Python Scene Generation Report

Date: 2026-02-07

1. Overview

This project builds a small room scene in Blender using a mix of manual modeling and procedural generation with a Python script. The final scene includes a checkerboard floor, walls, two scripted chairs, and two manually modeled assets (a chair and a table) that are appended from a separate .blend file and positioned via code.

2. Manual Modeling (manual.blend)

In the manual .blend file, I created two mesh assets named exactly **Chair** and **Table**. Each asset was assembled from primitive cubes scaled and positioned to form legs, seat/table-top, and backrest (chair). All parts were joined into a single object per asset (Object → Join), and the final objects were renamed in the Outliner.

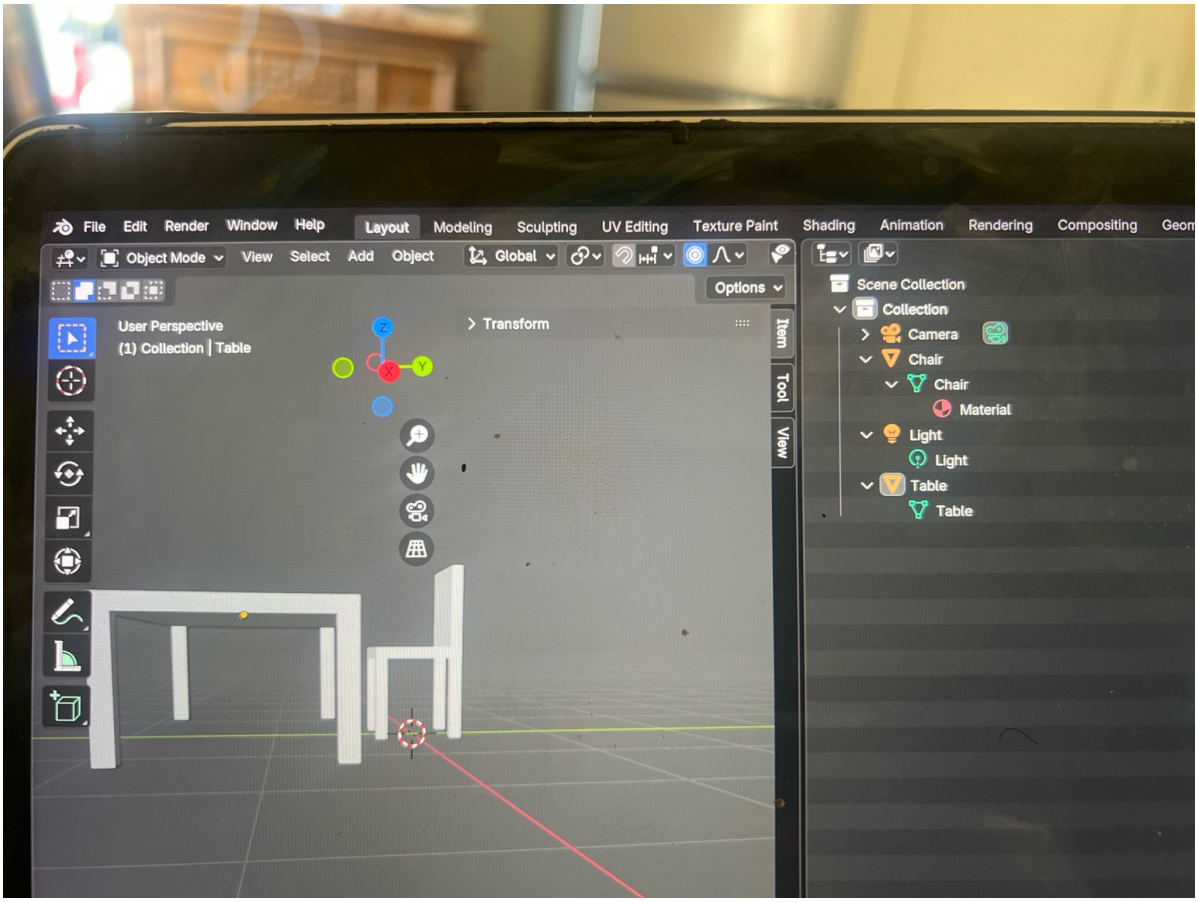

Figure 1: Manual Chair and Table assets created and named in the Outliner.

3. Scripted Scene Generation (main.py)

The script clears the scene, defines a helper **create_block()** to spawn colored cubes, and then generates:
• A checkerboard floor (10×10 tiles) using alternating materials
• Three walls (back/left/right)
• Two procedural chairs composed of cubes parented to an Empty root so the chair can be rotated as one unit
• Appends the manual Chair and Table from the manual .blend file and positions them in the room

How to run	Open Blender → Scripting workspace → Open Text (main.py) → Run Script
Naming requirement	Manual objects must be named exactly: Chair, Table (case-sensitive)
Manual blend path	MANUAL_BLEND should be an absolute path to manual.blend

4. Materials

Materials are assigned via Blender's Material Properties panel. The script creates materials programmatically and assigns them to objects. The floor alternates colors to form a checkerboard pattern, while walls and furniture use their own materials. The screenshot below shows a selected floor tile and its assigned material.

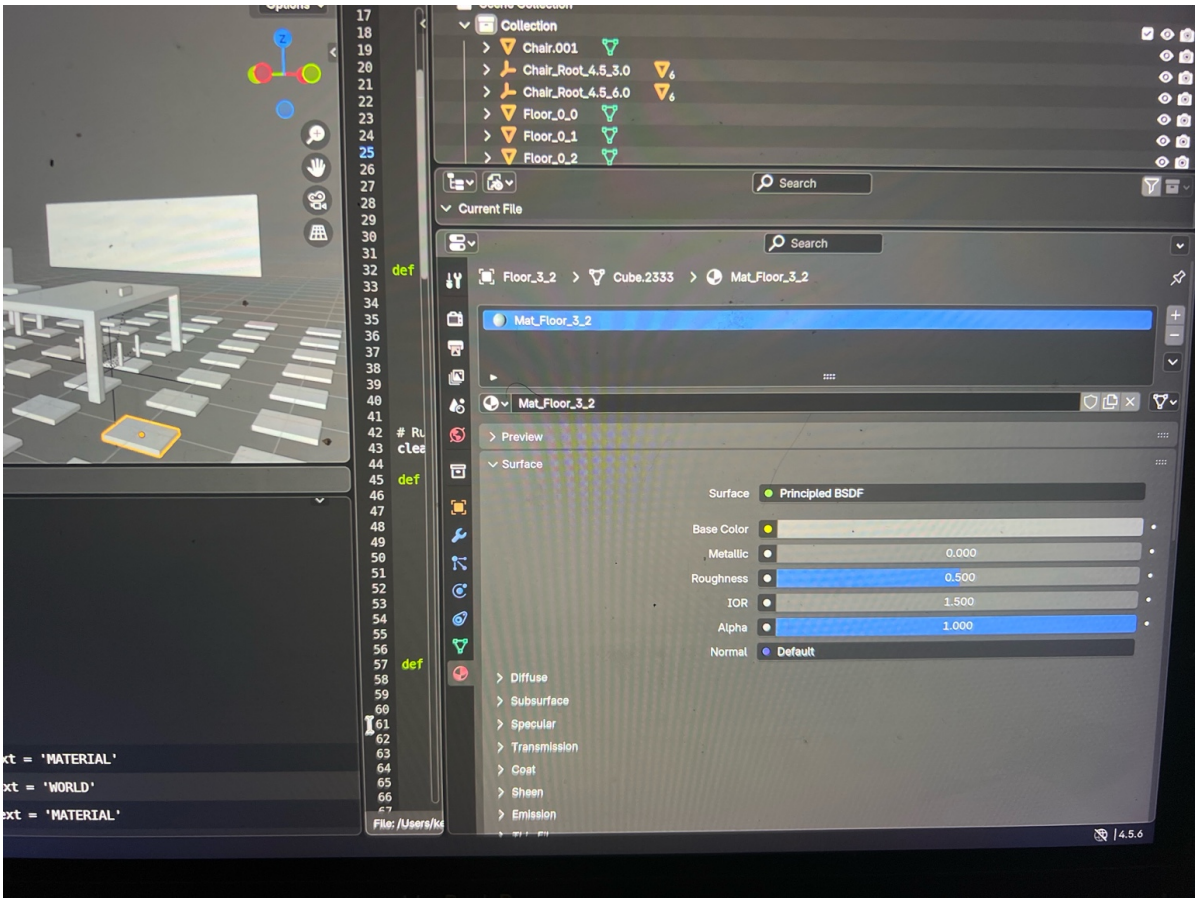

Figure 2: Material Properties panel showing the material assigned to a floor tile.

5. Script Output / Console Evidence

The Blender console shows calls to add primitives and confirms the script ran successfully to assemble the scene, providing evidence of procedural generation.

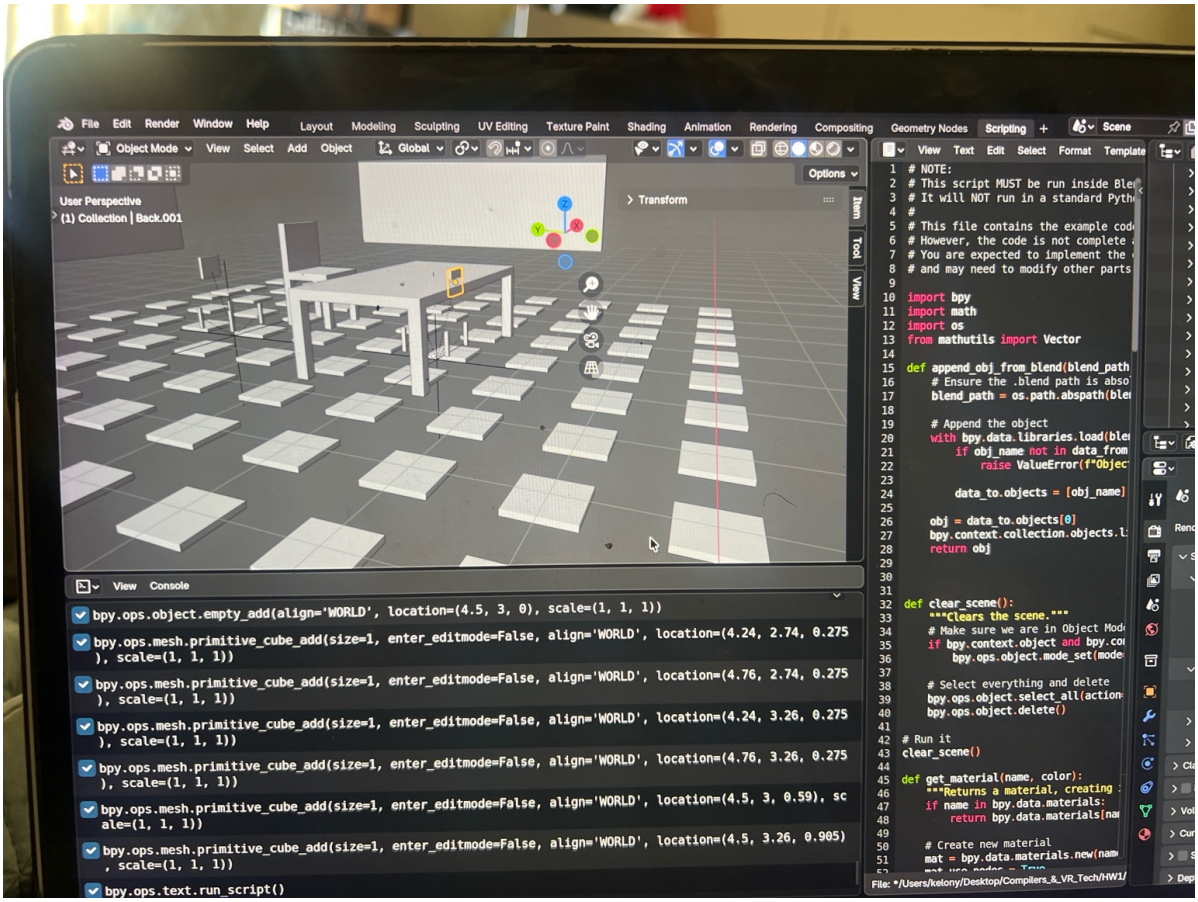

Figure 3: Blender console output while running main.py.

6. Final Scene Result (final.blend)

After generating the room and scripted chairs, the script appends the manual Chair and Table assets from the manual .blend file and places them in the room. Because the floor tiles are centered slightly below z=0, appended assets were placed with a small positive Z offset to avoid clipping.

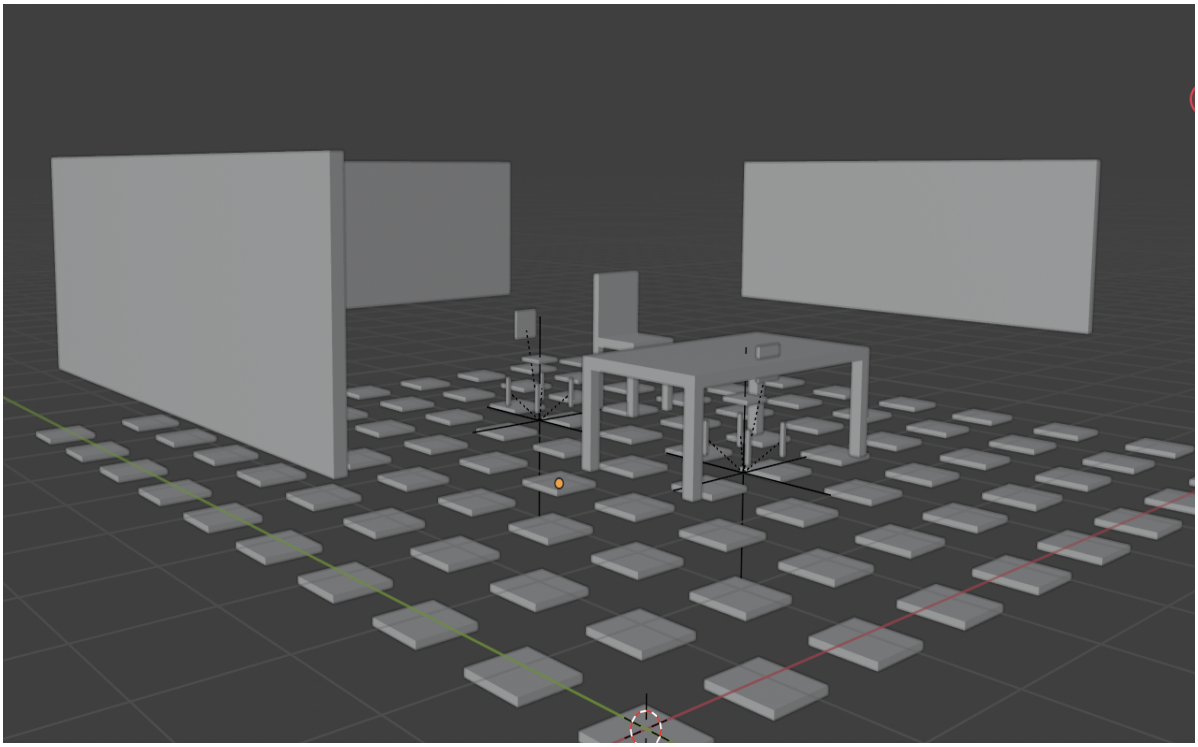

Figure 4: Final scene overview showing room layout and placed furniture.

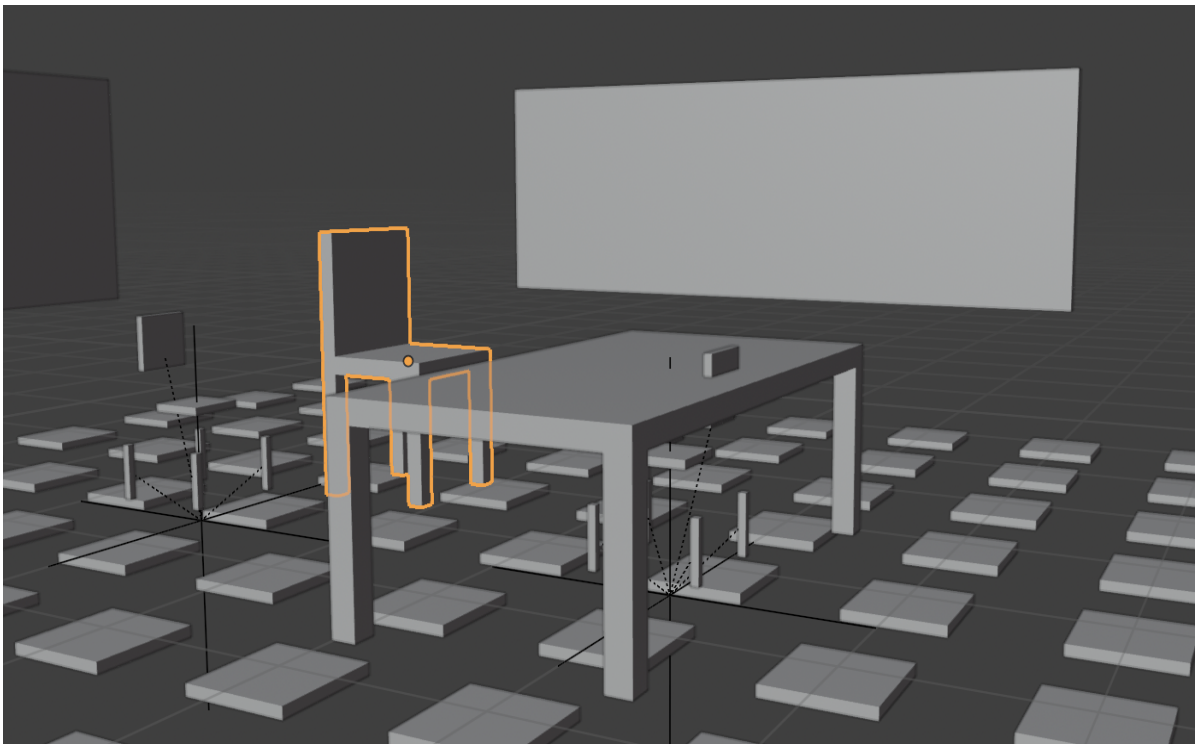

Figure 5: Close-up of the final furniture placement (chair and table).

7. Notes / Issues Encountered

• File path issues: a stray space in the manual .blend path prevented Blender from opening the file; using the correct absolute path fixed the load error.
• Object naming: appending is case-sensitive; the manual assets must be named exactly Chair and Table in the manual file.
• Floor intersection: because floor tiles sit slightly below z=0, appended assets were moved upward slightly (positive Z) to avoid clipping.