Experiment 9

9. Simulating Semi-Rigid and Rigid Debris with Python.

Steps :

**🔹 Step 1: Open Blender and Prepare the Scene**

1. Open a new Blender project.
2. Delete the default cube (optional).
3. Add a mesh (like a cube or plane) to act as the **ground**:
   * Shift + A → Mesh → Plane
   * Scale it up (S → 10).
   * Set it as a **Rigid Body → Passive** (so it doesn’t move).

**🔹 Step 2: Add Rigid Debris (Fully Rigid Objects)**

1. Add a cube or sphere for debris:
   * Shift + A → Mesh → Cube (or Sphere).
   * Set it as **Rigid Body → Active**.
   * Keep **Rigid Body type = Active**, collision shape = Convex Hull (good for debris).
2. Duplicate (Shift + D) several times to create multiple debris pieces.
3. Now, if you press Space → Play, debris will fall and bounce rigidly.

**🔹 Step 3: Add Semi-Rigid Debris (Flexible but Breakable)**

Semi-rigid means **partly flexible, partly rigid**. We simulate this using:

* **Rigid Body Constraints** (like fixed joints or springs).
* **Breakable setups** with Python.

**Example setup:**

1. Add a chain of small cubes (like a beam).
2. Set each cube as **Rigid Body → Active**.
3. Add **Rigid Body Constraints** (Shift + A → Rigid Body Constraint → Point).
   * Connect each cube to the next one (like a hinge).
   * This creates semi-rigid behavior (bends, but still rigid).
4. Enable **Breakable constraints via Python** so they snap under force.

**🔹 Step 4: Python Script for Debris Simulation**

1. Open **Scripting tab** in Blender.
2. Create a new script and paste this:

import bpy

# Reset scene

bpy.ops.object.select\_all(action='SELECT')

bpy.ops.object.delete(use\_global=False)

# Add ground

bpy.ops.mesh.primitive\_plane\_add(size=20, location=(0,0,0))

ground = bpy.context.active\_object

bpy.ops.rigidbody.object\_add()

ground.rigid\_body.type = 'PASSIVE'

# Add rigid debris

for i in range(5):

bpy.ops.mesh.primitive\_cube\_add(size=1, location=(i\*2, 0, 5))

cube = bpy.context.active\_object

bpy.ops.rigidbody.object\_add()

cube.rigid\_body.mass = 1

cube.rigid\_body.collision\_shape = 'CONVEX\_HULL'

# Add semi-rigid chain (linked cubes)

prev\_obj = None

for i in range(5):

bpy.ops.mesh.primitive\_cube\_add(size=0.5, location=(0, i\*0.6, 8))

obj = bpy.context.active\_object

bpy.ops.rigidbody.object\_add()

obj.rigid\_body.mass = 0.5

obj.rigid\_body.collision\_shape = 'BOX'

# Add constraint if not the first

if prev\_obj:

bpy.ops.object.empty\_add(type='PLAIN\_AXES', location=((prev\_obj.location + obj.location)/2))

empty = bpy.context.active\_object

bpy.ops.rigidbody.constraint\_add()

empty.rigid\_body\_constraint.type = 'POINT'

empty.rigid\_body\_constraint.object1 = prev\_obj

empty.rigid\_body\_constraint.object2 = obj

prev\_obj = obj

👉 This script will:

* Create a ground plane.
* Spawn rigid cubes (debris).
* Create a **semi-rigid chain** of cubes with constraints.

**🔹 Step 5: Run the Simulation**

1. In the scripting editor → press **Run Script**.
2. Switch to the **Timeline** and press **Play**.
   * Rigid debris cubes will fall and bounce.
   * The semi-rigid chain will bend and swing before breaking apart if constraints are removed.

✅ At this point, you have:

* **Rigid debris** (independent, solid).
* **Semi-rigid debris** (connected, flexible pieces).
* A Python script that builds the whole setup automatically.