

Script overview

This file is provided to give an overview of the included scripts.

Script:

- Dismember\Demo\Scripts**BallShooter.cs**

Purpose:

- Demonstrate how to add collision damage by shooting objects.

Usage:

- Attach the script to the main camera.
Press LMB (Left mouse button) to shoot a ball towards the target.

Script:

- Dismember\Demo\Scripts**Damage.cs**

Purpose:

- General purpose damage handler.
Adds damage to limbs by using raycast and adds force to rigidbodies.

Usage:

- Inherit this class to create custom damage handler.

Script:

- Dismember\Demo\Scripts**DamageOnCollision.cs**

Purpose:

- Demonstrate how to add collision damage.

Usage:

- Attach to the gameobject you want to be dealing damage.
Include Collider and Rigidbody.

Script:

- Dismember\Demo\Scripts**RayShooter.cs**

Purpose:

- Demonstrates how to use the "Damage" script to shoot rays from the mouseposition that will damage the target.

Usage:

- Attach to the main camera and shoot with LMB (Fire1)

Script:

- Dismember\Demo\Scripts**Respawner.cs**

Purpose:

- Respawns a model prefab after it "dies"

Usage:

- Add to an empty gameobject in the scene and drag your prefab to the slot in the Inspector.
-

Script:

- Dismember\Editor**DismemberManagerEditor.cs**

Purpose:

- Adds the buttons to the Inspector used by the DismemberManager.

Usage:

- Just needs to be in a Editor folder for unity to find and use it.
-

Script:

- Dismember\Scripts**ColliderHelper.cs**

Purpose:

- Generates colliders based on mesh sizes.

Usage:

- See the AddLimb function in DismemberManager.
-

Script:

- Dismember\Scripts**DismemberManager.cs**

Purpose:

- Controls overall health, events and sets up ragdoll with or without dismembering.

Usage:

- See documentation.
-

Script:

- Dismember\Scripts**EventTypes.cs**

Purpose:

- Extends UnityEvent to make them visible in the Inspector.

Usage:

- Used internally in DismemberManager (See events in there)
-

Script:

- Dismember\Scripts**GenericDismembering.cs**

Purpose:

- This is the limb script that handles the health of the limb, and takes care of creating the new mesh when a limb is shot off and disable the part that has been dismembered.

Usage:

- Add to the limbs you want positional damage on, and check wether you want the limb to dismember, select the appropriate mesh and select what type it is (For damage multiplier)
Will be added and configured automaticly, if you use the DismemberManager's "Setup dismembering" on a valid humanoid model.
-

Script:

- Dismember\Scripts**LimbData.cs**

Purpose:

- A data container class used internally to pass relevant data between functions.

Usage:

- Not being used.
-

Script:

- Dismember\Scripts**RagdollCreator.cs**

Purpose:

- Create rigidbodies and character joints for a simple ragdoll. Please note that the ragdoll is without colliders as they are added by the colliderhelper class in DismemberManager.

Usage:

- From DismemberManager:
RagdollCreator ragdoll = new RagdollCreator(gameObject);
ragdoll.Create();
-

Script:

- Dismember\Scripts**TPSA_AI_Integration.cs**

Purpose:

- Easy integration with the deprecated asset TPS_AIM
Implemented per request.

Usage:

- Please see "TPSA" under "Integration" in the documentation
-

Script:

- Dismember\Scripts**AdvancedDismembering.cs**

Purpose:

- Adds options for dismembering of individual limbs.

Usage:

- Add the script where the DismemberManager is added and use the inspector to setup custom behavior.