**布娃娃向导**

Unity 有一个简单的向导可用于快速创建布娃娃。只需在向导中的各种属性上拖动不同的肢体即可。然后，选择 Create，Unity 便会自动生成构成布娃娃的所有\_\_碰撞体**、**刚体\_\_和\_\_关节\_\_。

**创建角色**

布娃娃使用\_\_蒙皮网格\_\_；这是在 3D 建模应用程序中使用骨骼装配的角色网格。因此，必须在 Maya 或 Cinema4D 等 3D 软件包中构建布娃娃角色。

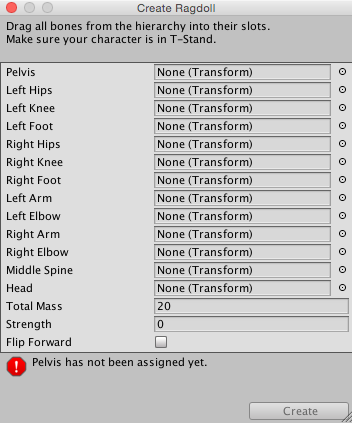
创建角色并进行装配后，将资源正常保存在 **Project 文件夹\_\_中。切换到 Unity 时，您将看到该角色资源文件。选择该文件，然后 Inspector 中将显示**Import Settings\_\_ 对话框。确保未启用 **Mesh Colliders**。

**使用向导**

不太可能将实际的源资源变成一个布娃娃。这将需要修改源资源文件，因此是不可能的。正确的做法是将角色资源的实例变成一个布娃娃，然后保存为\_\_预制件\_\_以供重复使用。

创建角色实例的方法是将角色从 **Project 视图\_\_拖动到**Hierarchy 视图**。单击层级视图中实例名称左侧的小箭头，展开其\_\_变换层级视图**。现在已准备好开始分配布娃娃部件。

从菜单栏中选择 **GameObject > 3D Object > Ragdoll…** 以打开布娃娃向导。现在将看到向导本身。

布娃娃向导

为布娃娃分配部件的过程不言自明。将角色实例的不同变换拖动到向导上的相应属性。如果已自行创建角色资源，此过程应该特别容易。

完成后，单击 **Create** 按钮。现在进入\_\_播放模式\_\_时，应该会看到角色像布娃娃一样跛行。

最后一步是将设置的布娃娃保存为预制件。从菜单栏中选择\_\_Assets > Create > Prefab\_\_。随后将看到在 Project 视图中显示了一个新预制件 (New Prefab)。将该预制件重命名为“Ragdoll Prefab”。将布娃娃角色实例从层级视图拖动到“Ragdoll Prefab”上。现在有了一个完整设置并可重复使用的布娃娃角色，可在游戏中任意多次使用。

**注意**

对于通过布娃娃向导制作的角色关节，请注意设置：关节的扭转轴与肢体的最大摆动轴对应，关节的 Swing 1 轴与肢体的较小摆动轴对应，而关节的 Swing 2 表示扭转肢体。采用这种命名方案是为了沿用旧版。

# 关节和布娃娃稳定性

本页提供有关改善[关节](file:///E:\UnityDocumentation\Manual\Joints.html)和[布娃娃](file:///E:\UnityDocumentation\Manual\wizard-RagdollWizard.html)稳定性的技巧。

* 避免将 **Angular Y Limit** 和 **Angular Z Limit** 设置为较小的关节角度。根据具体设置，为了保持稳定，最小角度应在 5 到 15 度左右。尝试将角度设置为零，而不是使用小角度。这样会锁定轴并提供稳定的模拟。
* 取消选中关节的 **Enable Preprocessing** 属性。通过禁用预处理，可在关节被强制进入无法满足关节约束条件的情况时帮助防止关节不规律地分离或移动。如果通过关节连接的[刚体](file:///E:\UnityDocumentation\Manual\class-Rigidbody.html)组件被静态碰撞几何体拉开（例如，在墙内不完整生成布娃娃），则会发生这种情况。
* 在极端情况下（例如在墙内不完整生成布娃娃或用大力推动布娃娃），关节解算器无法将布娃娃的刚体组件保持在一起。这种情况下可能导致拉伸。要解决此问题，请使用 [ConfigurableJoint.projectionMode](file:///E:\\UnityDocumentation\\ScriptReference\\ConfigurableJoint-projectionMode.html) 或 [CharacterJoint.enableProjection](file:///E:\\UnityDocumentation\\ScriptReference\\CharacterJoint-enableProjection.html) 在关节上启用投影。
* 如果与关节连接的刚体组件抖动，请打开 Physics Manager (**Edit** > **Project Settings** > **Physics**) 并尝试将 **Default Solver Iterations** 值增加到 10 到 20 之间。
* 如果与关节连接的刚体组件未准确响应反弹，请打开 Physics Manager (**Edit** > **Project Settings** > **Physics**) 并尝试将 **Default Solver Velocity Iterations** 值增加到 10 到 20 之间。
* 在运动刚体组件由关节连接到其他刚体组件情况下，切勿使用直接变换访问。这样做会跳过 PhysX 计算相应刚体组件的内部速度的步骤，导致解算器提供意外的结果。一种常见的错误实践例子是在 2D 项目中使用直接变换访问通过在骨架的根节点上更改 [Transform.TransformDirection](file:///E:\\UnityDocumentation\\ScriptReference\\Transform.TransformDirection.html) 来翻转角色。如果改用 [Rigidbody2D.MovePosition](file:///E:\UnityDocumentation\ScriptReference\Rigidbody2D.MovePosition.html) 和 [Rigidbody2D.MoveRotation](file:///E:\UnityDocumentation\ScriptReference\Rigidbody2D.MoveRotation.html)，那么此行为会改善很多。
* 避免关节连接的刚体组件之间的质量差异过大。一个刚体质量是另一个刚体质量的两倍是可以的，但当一个刚体质量比另一个刚体质量大十倍时，模拟就会变得不稳定。
* 尽量避免在包含刚体或关节的变换组件中使用不等于 1 的缩放。这样的缩放不可能在所有情况下都可靠。
* 如果刚体组件在插入到世界后发生重叠，并且无法避免重叠，请尝试降低 [Rigidbody.maxDepenetrationVelocity](file:///E:\\UnityDocumentation\\ScriptReference\\Rigidbody-maxDepenetrationVelocity.html) 使刚体组件更加平滑地相互退出。