

钥匙、门和箱子

本页介绍了如何使用自上而下引擎中的钥匙来解锁门、打开箱子、从中获取战利品以及与一般事物进行交互。

- [介绍](#)
- [创建密钥](#)
- [创建一扇门](#)
- [重点经营区](#)
- [创建一个胸部](#)

介绍

多亏了库存引擎，自上而下引擎本身就支持密钥的创建、收集和使用。钥匙是库存物品，与场景元素上的一些组件相结合，可以让您与门互动、打开箱子、掠夺其中的内容，并通常激活场景中的东西。如果您想查看实际示例，您可以查看（除其他外）位于 Demos/Minimal2D 中的 MinimalSandbox2D 演示场景或 KoalaDungeon 场景，它们将提供关键激活门和箱子的工作示例。

创建密钥

如果您熟悉库存引擎，这将很容易。如果没有，您可能应该查看其[文档](#)。要创建密钥，只需右键单击您的资源文件夹之一，然后转到"创建">"更多山脉">"自上而下引擎">"InventoryEngineKey"。为这个新创建的可编写脚本的对象实例命名，您就可以在其检查器中填写其详细信息。

创建密钥

密钥的检查器与您可以在库存引擎中创建的任何基本项目相同，因此我们不会在此处再次详细介绍，但请注意，密钥的名称 (ItemID) 非常重要，因为它是您将要创建的字符串用来表示什么钥匙可以打开什么门/箱子/等，所以它需要是唯一的。

一旦您创建了该可编写脚本的对象实例，剩下要做的就是为它创建一个可选项目。如果您不记得这是如何完成的，[请查看此文档](#)。

创建一扇门

您可以用钥匙打开的最常见的东西之一是门。所以让我们从那个开始。在 TopDown Engine 中创建可打开门的最简单方法是重复使用 2D 或 3D 门预制件。它也可以是移动平台、墙壁或任何您想要的东西。唯一的强制性元素是关键操作区组件。

钥匙操作区域

关键操作区示例

一旦你有了移动门/平台，你需要一个区域来与钥匙交互并打开它。为此，创建一个空的游戏对象，向其中添加一个 Box Collider 2D，将其设置为 IsTrigger，根据您的喜好调整其大小并将其放置在您的钥匙选择器项目和您的门之间的某个位置。然后，向其添加 KeyOperatedZone 组件。

KeyOperatedZones 是 ButtonActivated 类的扩展，这意味着它们将结合 ButtonActivated 元素（例如对话区）的优点和选项，并为它们添加键支持。

从区域的检查器中，您可以设置区域是否自动激活（在这种情况下需要您按下按钮）。您还可以指定激活次数以及每次使用之间的延迟。如果您的区域未自动激活，您可以决定始终或仅在碰撞时显示按钮提示，以及它应该出现的位置。最后，您可以决定该区域需要一个密钥，在这种情况下，该区域应该在您角色的库存中查找什么 KeyID。

最后要做的是将一个动作绑定到这个区域。此操作是场景中任何对象的方法，当区域被激活时（当您进入它并满足您刚刚定义的激活它的条件时：按下按钮，输入库存等）将被触发。对于 2D 门，您必须将 DungeonDoor 组件拖到我们的操作字段中，然后选择其 ToggleDoor 方法。

使用这个简单的组件，您可以控制门，但实际上任何具有可绑定到此 Key Action 字段的公共方法的对象。您可以让门在您靠近时或按下按钮时打开，或者没有钥匙就无法打开等。

创建一个箱子

创建一个箱子就像创建一个门一样简单。像以前一样创建一个关键操作区，然后将精灵或 3D 模型放入场景中的胸部。在那个箱子对象上，添加一个 `InventoryEngineChest` 组件。然后你所要做的就是向它添加 `ItemPicker` 组件（每个你想要的物品一个）。然后像以前一样将您的 `Chest` 绑定到 `KeyOperatedZone` 的检查器，这次使用 `InventoryEngineChest.OpenChest` 方法。你完成了！激活后，`InventoryEngineChest` 组件将自动选取您添加到其中的所有 `ItemPickers`，并将它们的物品放入您角色的库存中。此外，您可以向其添加动画师以触发打开动画。然后，您的动画师将需要一个名为 `"OpenChest"` 的触发器动画参数。

您可以在 `KoalaDungeon` 演示场景中找到一个可用的 `Chest` 示例。