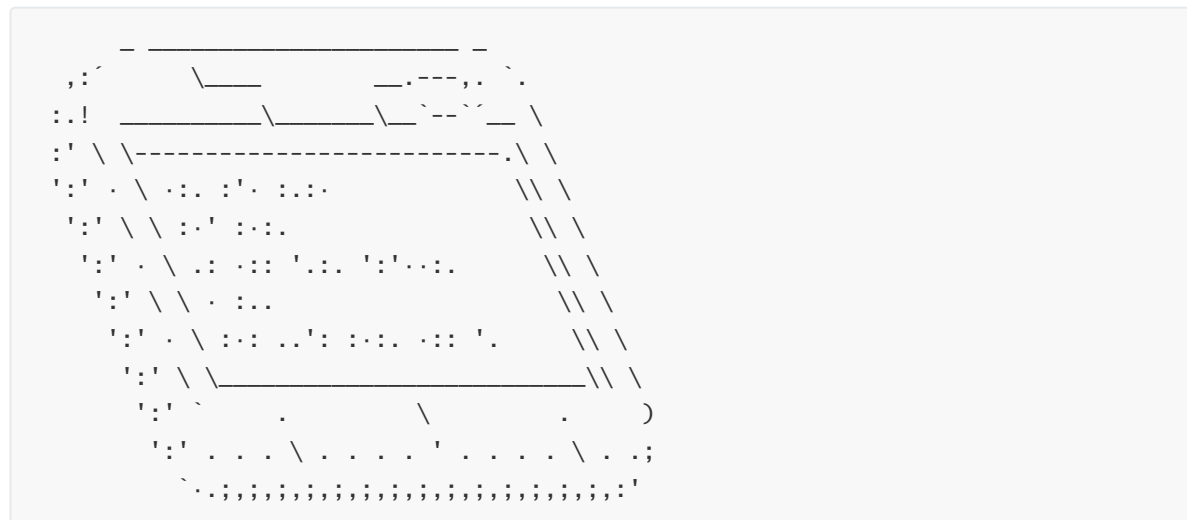


# Stone Story RPG 石语言官方文档翻译——简介

## · ∴ ∴ 意念石简介 ∴ ∴ ·



在这一教程中，你将学习到：利用**意念石mind stone**以及**石语言Stone Script**来自动通过枯木峡谷；

预计完成时间: 20分钟；

所需的先验知识:

- 玩石头纪Stone Story RPG，直到打败娜迦王；
- 了解在一个地点中途更换武器的战略价值；
- 建造坩埚并了解治疗药剂；
- 理解衔尾蛇之石会使你在一个关卡中循环。

所需项目:

- 至少为4星级的剑、盾和弩；
- 80滴焦油——用以炼制药水；
- 坩埚已启用自动续杯功能。

意念石是非常强大的遗物。它的目的是实现自动操作，例如在跑步过程中更换武器和使用药水。将指令输入游戏智能，告诉它该做什么。如果设置正确，意念石会能够让你的角色立即适应游戏中的变化，并实现超越玩家手操的操作。

这个遗物也是《石头纪Stone Story RPG》通向定制和个性装扮的大门。探索其他玩家完成的作品，甚至创建自己的作品！例如宠物，帽子，自定义界面，完整的迷你游戏等等。[点击这里了解定制和个性装扮。](#)

意念石中的指令是用石语言Stonescript编写的，这是一种为每个人设计的极简语言。只要你会打字，你就可以使用这种脚本——不需要编程基础！但是，如果您想深入研究，请参阅[Stonescript手册](#)，其中详细列出了所有可用的说明和示例。

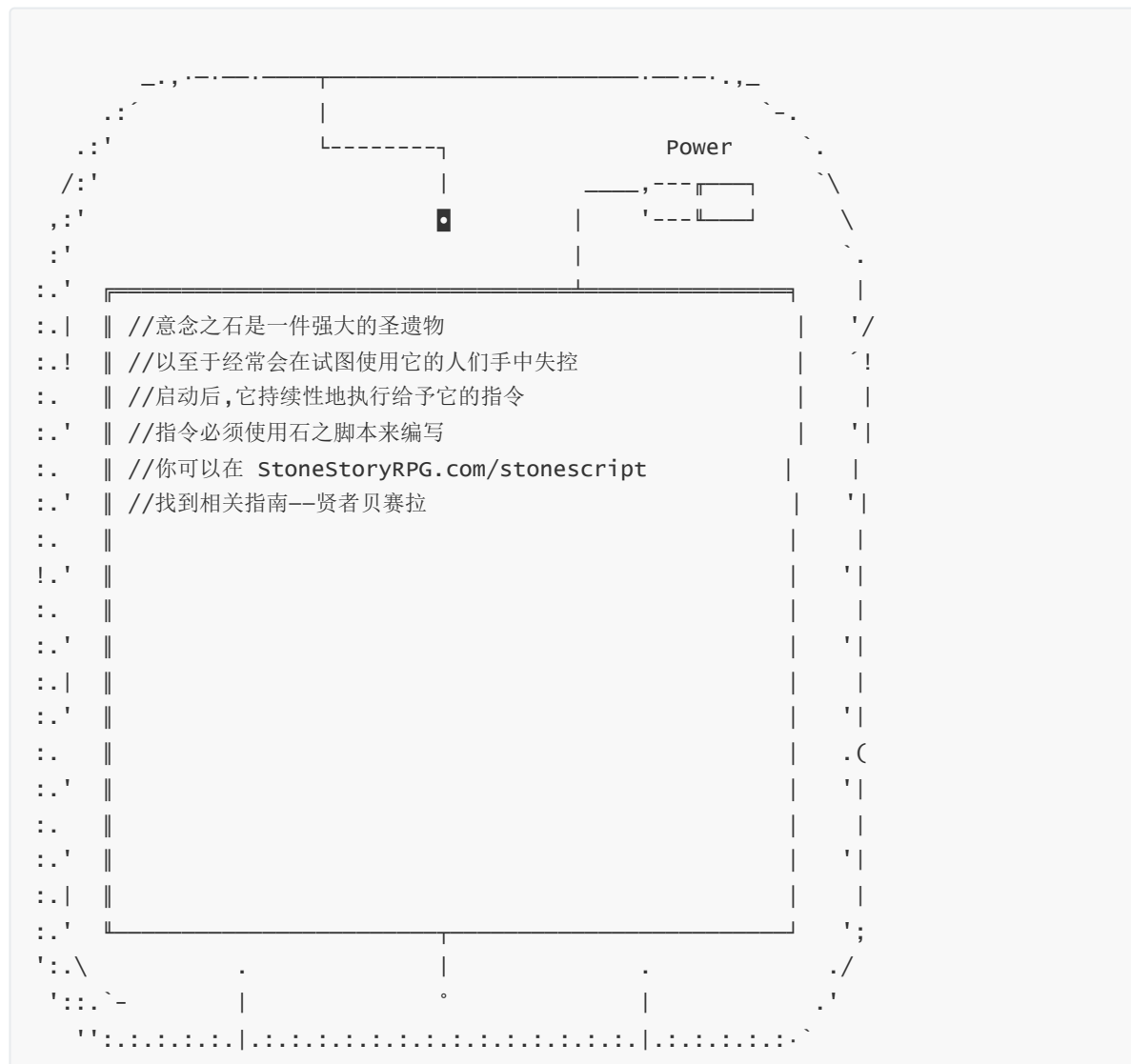
如果你想要更多帮助或是合作编写脚本，可以访问我们的[Discord](#)

好的，那让我们开始吧！

## ⌘ 1. 清除意念石 ⌘

选择所有默认出现在你的意念石上文本并删除。如果你愿意的话，你可以保留贤者贝塞拉Bezerra, the Sage留下的这一段开头，因为任何以 `//` 开头的文本都是无效的，会被AI忽略。

提示：全选的快速方式是Ctrl + A。



## ⌘ 2. 治疗 ⌘

```
)(  
(~  
—
```

指令可以将你的常识转化为游戏的系统可以理解的东西。例如，只有你快要死了的时候，使用治疗药剂 Healing Potion才有意义的，难道不是吗？

有一个命令可以这样做。在意念石中键入以下内容：

```
activate potion
```

这样的代码会告诉系统该做什么。当你访问一个地点时，它会在每一帧不断尝试激活药剂。一旦你失去一些生命值，它就会成功激活。如果你有一个不是治疗药剂的其他药剂，如幸运药剂，那么当你访问一个地点时，它会立即激活。但可能不是我们想要的结果，因此让我们改进它。

我们可以想出这样一个更完整的指令：

```
"如果 我的血量比较低 则 激活药水"  
"IF my hitpoints are low THEN activate potion"
```

把这句话翻译成Stonescript，修改意念石如下：

```
?hp < 7  
    activate potion
```

第一行 `?hp < 7` 在进行一个判断，“我的生命值是否小于7？”与人类语言相反，游戏要求问号作为一行的开端，这样系统马上知道这是一个判断，并知道如何阅读其余部分(有点像西班牙语？)。

**注意激活药剂的代码 `activate potion` 前面增加的空格。**空格很重要，因为它会将这一行分类至依赖于它的上一行。这样，只有当上面的判断等于“是 `true`”时，它才会激活。

**空格真的很重要！！！！一定要写对空格！！！！**

---

### ⋈ 3. 游玩 ⋈

```
\|' /',  
`| / ,  
^ / `-'  
\\ò ó' /'  
| , (  
'i'!| |  
|-' _'-.  
-/_`'\`
```

你可以带着你启动了的意念石和新调制的药剂前往任何危险的地方，看着你受到敌人的伤害。当你的生命值低于7时，它会自动激活药剂。

提示：要在一个地点的中间重新打开意念石，请按键盘上的“M”按键。你没必要回到工作台界面才能打开意念石。当你关闭意念石并回到你所在的地点时，系统将重新加载你的脚本，你写的新指令也将生效。

---

### ⋈ 4. 武器 ⋈

```
| o__  
+/( )_  
/ \
```

本教程中我们的目标是自动放置的刷枯木峡谷。一个基本的剑Sword+盾Shield的组合就足以完成本地点的简单难度关卡，这也会是一个好的开始。

在你的意念石的底部增加以下内容：

```
equipL sword
equipR shield
```

这个 `equip` 命令正是如你所期望的那样，是用来装备武器的。它从你的库存中找到一件武器并装备，而这个命令右侧的所有文本都是指令的附加详细信息。`L` 和 `R` 分别表示用哪只手放置武器，是左手（left）还是右手（right）。你可以添加更多的细节来指定装备哪种武器，因为你的库存中可能有很多剑。例如，您可以添加 `*7` 来指定你要装备的是一件7星武器：`equipL sword *7`。如果没有任何额外的细节，系统会找到你最好的武器并装备它，这也许是你默认想要的。（如果你额外细节仍无法区分武器，或者没有满足你描述的武器，那系统也会选择它认为符合的最好的武器并装备。所谓最好的武器，通常是你背包里面靠前位置的武器，默认会先按照附魔等级排序，随后是星级）

对于中文玩家来说，你可以使用中文来代替指令中的装备名字，但是一定要注意你打的名字是绝对准确的（例如长剑和大剑的区别）。

```
|/)
0.-.-
`/|`-`
`|`
```

一般来说，这很好，但是当你对付蚊子的时候，最好装备一把弩Crossbow来避免所有的伤害。虽然蚊子比较弱，但是可以对你造成密集攻击并造成很多伤害：

```
?foe = mosquito
equip crossbow
```

关键字 `foe` 允许我们检查当前被系统瞄准的敌人，然后根据敌人的类型来做出决定。

你可能会注意到，在这个 `equip` 命令中没有 `L` 或 `R`。对于弩，我们暂时先不考虑任何细节，让系统自己去决定什么更好吧。现在需要事情保持简单。

您的整个脚本应该如下所示：

```
?hp < 7
  activate potion

equipL sword
equipR shield
?foe = mosquito
  equip crossbow
```

此时有必要声明一下，你的脚本是自上而下执行的。它会先检查药水，然后装备剑和盾，最后，如果你面对的是蚊子它会装备弩。整个脚本每秒运行30次，即每秒有30帧frame。

---

## ⋯ 5. 木头 ⋯

```
?harvest.distance < 10
  equip hatchet
```

[illegible]

```
?foe ! boss & foe.distance > 8
  equipL ouroboros
```

第一行的第二部分是 `foe.distance > 8`，意思是“敌人到玩家的距离是否大于8？”。第1行中的两个判断用 `&` 符号合并成一个判断，表示“并且”。（学过电路或程序的玩家可知道这个是“与”符号）

"当下一个敌人不是**boss**，并且它与玩家的距离大于8的时候，在左手装备衔尾蛇之石。"

---

## ⋮⋮ 7. 自动拾取 ⋮⋮

```
,  
_/_-'
```

你可能会注意到，有一件事让我们的速度慢了下来——我们必须捡起地上的东西，例如在枯木峡谷里面散落的木头。幸运的是，星之石Star Stone(装备中)可以吸引所有的可拾取物，这样就能大大加快移动速度。

在代码中添加以下内容:

```
?pickup.distance < 10  
    equipL star
```

类似于 `foe.distance`，上面这个指令结构是在询问下一个可拾取物是否离玩家的距离少于10。如果满足条件的话，我们就在左手装备星之石。我们可以只写 `star`，这就足以识别星之石这件物品，因为没有任何其他物品的名字中有“星”。

---

## ⋮⋮ 8. 综上所述 ⋮⋮

至此，我们的枯木峡谷的脚本就完成了！但是这些指令目前将在所有的地点运行。我们可能希望为每个地点定制一组特定的脚本结构，所以我们需要把这些脚本指令分组，并且只在我们探索枯木峡谷时才运行这些指令。

为了获取我们的位置，我们可以使用关键字 `loc`。

在以上脚本的**顶部**，添加以下内容:

```
?loc = deadwood
```

同时也要在所有行的前面增加空格，这样它们就从属于这个位置判断语句了。整个脚本应该如下所示:

```
?loc = deadwood  
    ?hp < 7  
        activate potion  
  
    equipL sword  
    equipR shield  
    ?foe = mosquito  
        equip crossbow  
  
    ?harvest.distance < 10  
        equip hatchet  
  
    ?foe ! boss & foe.distance > 8  
        equipL ouroboros  
  
    ?pickup.distance < 10  
        equipL star
```

提示:你可以使用Ctrl+C / Ctrl+V快捷键从这里或其他玩家那里复制/粘贴脚本到意念石。

## ⋮⋮ 接下来该做什么? ⋮⋮



下面是外部资源和迷你教程:

### 链接:

- 对于这个教程有任何吗, 或者需要在其他关卡的帮助吗? 访问我们的[Discord](#)。(中文玩家请加入QQ群 641088515)
- 发现由石头纪社区创建的[有趣的定制、个性装扮和小游戏。](#)
- 准备好更加深入的使用意念石了吗? 从[Stonescript手册](#)中学习关于石语言的一切。

### 生命偷取:

如果你有一把比较好的吸血剑 (dL), 你可以用它来管理你的生命值, 而非仅仅依靠药剂。这将极大地提高你的生存能力(并节省焦油):

```
?hp < maxhp & foe.distance < 7  
equipL sword dL
```

### 毒属性:

通过使用一把比较好的毒属性武器 (dP) 你可以减轻大部分来自boss的伤害。与吸血能力类似, 这能够大大改善你的生存能力和对药剂的需求, 使得你能够用最少的装备到达更高难度的地方:

```
?foe = boss & foe.debuffs.count = 0  
equipR dP
```

在这个方法中, 我们通过 `foe.debuffs.count` 检查敌人的debuff的数量, 以避免在减益时间用完之前再次使用。

### 机动性:

玩家经常使用冲刺盾Dashing Shield, 三曲腿之石Triskelion Stone甚至是铁头长棍Quarterstaff的技能来大大提高他们的循环速度。例如:

```
?foe.distance >= 11 & foe.distance <= 16  
equipR dashing
```

### 使用技能:

从这个功能开始, 事情会更加复杂。玩家的一个常见问题是如何激活武器技能。虽然一些物品如斧子或者是堕神之剑Blade of the Fallen God的技能激活比较容易, 但是例如巴迪什Bardiche这样的武器会给玩家带来大麻烦:

```
?foe = boss & foe.distance <= 10 &
^item.GetCooldown("bardiche") <= 0 |
^item.GetCooldown("bardiche") > 870
    equip bardiche
    activate R
```

你会注意到这里有几个新符号，`^` 代表这行指令是上一行的延续，`|` 表示“或”。

在这个例子中，我们使用了巴迪什的技能来对抗boss（`?foe = boss`），当boss在我们武器的攻击范围内的时候（`foe.distance <= 10`），我们装备巴迪什（`equip bardiche`），然后使用技能（`activate R`）。我们也在使用 `item.GetCooldown()` 检查巴迪什的技能的冷却时间。

我们使用冷却信息cooldown来使巴迪什只有在能够使用技能时才装备。不像一些瞬间激活的技能，巴迪什有一个“施法”时间，在此期间我们需要让巴迪什保持装备。游戏允许你在使用技能时装备其他东西，但是这个技能也就会被取消，而且会继续进入冷却。这就是 `870` 的由来：冷却值用帧来表示，**每秒钟相当于30帧**。巴迪什有30秒的冷却时间，相当于900帧。减去大约1秒的施法时间，就等于870帧。

为了更形象的了解巴迪什的技能冷却时间，试着在脚本里面加上这个

```
var cd
cd = item.GetCooldown("bardiche")
>`0,0,Bardiche cooldown = @cd@
```

这段代码片段将在左上角显示巴迪什的冷却时间。激活巴迪什的技能，看到冷却时间上升到900，然后倒计时回0。

---

版权所有：Martian Rex, Inc. 2020

翻译：逃税奸商