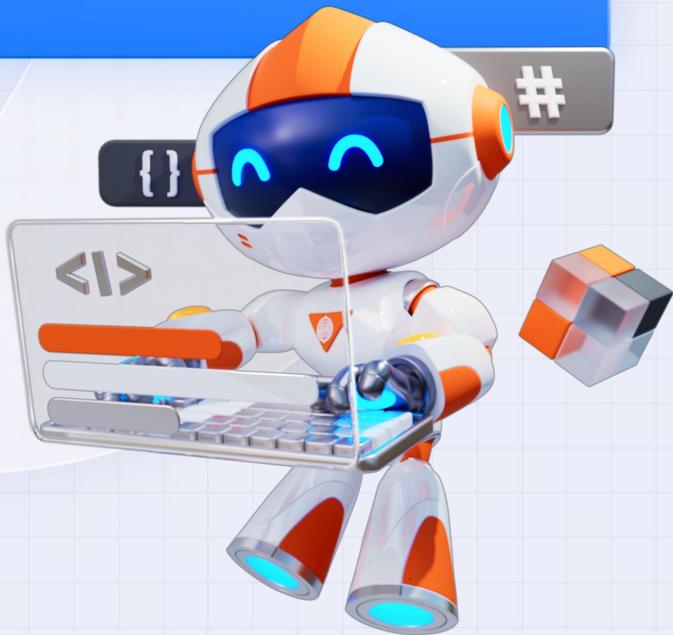


L10-3 | 丛林迷踪

模块29：列表闯关

模块30：手速大比拼



ANALYTICAL

INTELLECTUALLY CURIOUS

CREATIVE



## 项目内容

### ➊ 项目内容

列表闯关

手速大比拼

### 单词卡

animal:动物

score:分数

import:导入

random:随机的





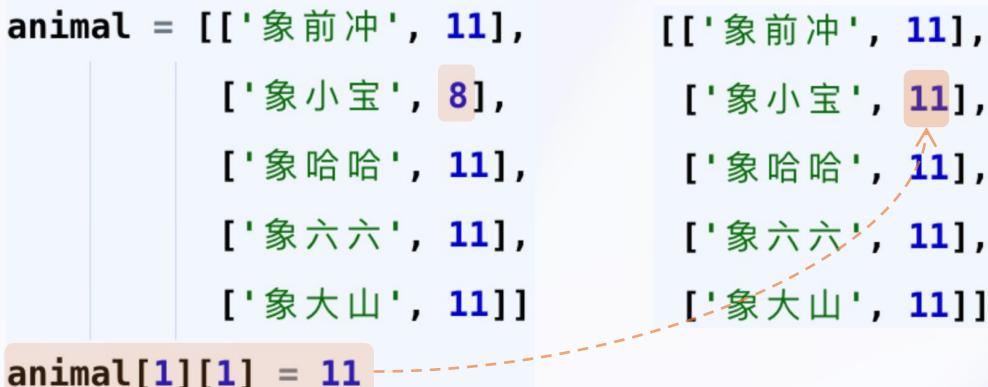
## 修改二维列表的某一个元素

### 概念详解

列表里面存列表就是**二维列表**。使用**列表[索引]**可以得到一个子列表。对于子列表，再使用一个索引：**列表[索引1][索引2]**，可以获取子列表的元素。

### 代码示例

```
animal = [[‘象前冲’, 11],  
          [‘象小宝’, 8],  
          [‘象哈哈’, 11],  
          [‘象六六’, 11],  
          [‘象大山’, 11]]  
  
animal[1][1] = 11
```



animal[1][1]可以得到第二个子列表的第二个元素。  
animal[1][1] = 11，就可以把8变成11。



## 修改二维列表的某一列元素

### 概念详解

二维列表就像是一个方阵，里面的元素按行按列地排布。对于同一列元素来说，它们的第一个索引依次增加，第二个索引相同。

### 代码示例

```
animal = [[['象前冲', 8],  
          ['象小宝', 8],  
          ['象哈哈', 8],  
          ['象六六', 8],  
          ['象大山', 8]]  
  
for i in range(5):  
    animal[i][1] = 11
```

animal[i][1]=11，可以把第二列所有的8变成11。



## 手速大比拼



项目介绍



手速大比拼

1 在4行4列的土地上出菜

2 收菜

3 记录分数



## 列表乘法

### 列表乘法



格式：列表 \* 数字

作用：可以把列表中的元素重复指定的次数。

### 代码示例

[1]\*3 → [1, 1, 1]

[0]\*4 → [0, 0, 0, 0]



## 二维列表应用

### 列表应用



```
[[0, 0, 0, 0],  
 [0, 0, 0, 0],  
 [0, 0, 0, 0],  
 [0, 0, 0, 0]]
```

土地有四行四列，可以用一个四行四列的二维列表来对应每一块土地。

列表中，用0表示没菜；1表示有菜。

初始的时候，所有土地都没菜，因此列表里都是0。

需要出菜的时候，就把二维列表的某一个元素变成1。



## 思考笔记

