

# GDScript拾遗

## 枚举类型

枚举实质上是常量的简写，适用于为某些常量连续赋整数值。

```
enum {TILE_BRICK, TILE_FLOOR, TILE_SPIKE, TILE_TELEPORT}
```

# Is the same as:

```
const TILE_BRICK = 0
const TILE_FLOOR = 1
const TILE_SPIKE = 2
const TILE_TELEPORT = 3
```

有时为了给常量分类，会给枚举取名。若给枚举取名，则该枚举将会把所有键纳入该名称的字典中。

```
enum State {STATE_IDLE, STATE_JUMP = 5, STATE_SHOOT}
```

# Is the same as:

```
const State = {STATE_IDLE = 0, STATE_JUMP = 5, STATE_SHOOT = 6}
```

```
func _ready():
    # Access values with Name.KEY, prints '5'
    print(State.STATE_JUMP)
```

字典也是一个对象，其键值对就相当于对象的属性。

## @GlobalScope

文档里的这一部分写了引擎内置的全局范围的常量和函数。

[Key枚举](#)

```
func _unhandled_input(event):
    if event is InputEventKey:
        if event.pressed and event.keycode == KEY_ESCAPE:
            get_tree().quit()
```

# GDScript 导出属性

在 Godot 中，你可以导出一些变量，这些变量可以在编辑器中查看和修改。

```
@export var number: int = 5
```