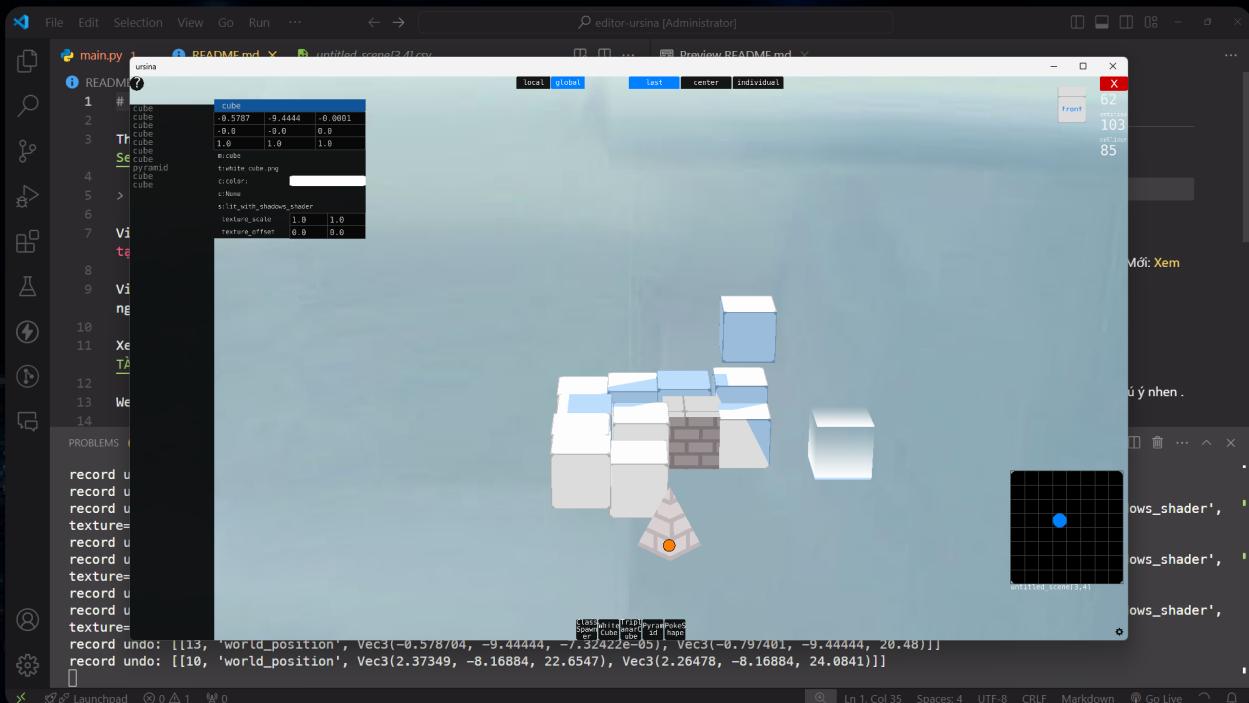


TÁC GIẢ: 8 SYNC DEV

PHẦN MỀM EDIT GAME URSINA PYTHON



Theo dõi kênh: [Tại đây](#).

Sẽ có cập nhật trong tương lai 🎉 !!!

Video Hướng Dẫn Chi Tiết Game Minecraft Ursina Phần 1: [Xem tại đây](#).

Video Hướng Dẫn Chi Tiết Game Minecraft Ursina Phần 2 (Đa người chơi) Mới: [Xem tại đây](#).

Xem Tài Liệu Đa Người Chơi: [Xem tại đây](#)

Website: [Xem tại đây](#)

Tài liệu sẽ được cập nhật định kì và thông báo trong group nên các bạn chú ý nhen

Các khóa học khác xem tại : [8 Sync Dev](#)

Hoặc liên hệ trực tiếp qua fb: [Kevin Nguyễn](#) để được hỗ trợ

GIẢI THÍCH MÃ NGUỒN

1. NHẬP THƯ VIỆN



{ 8 Sync Dev }

```
from ursina import *
from ursina.editor.level_editor import *
```

- `from ursina import *:` Nhập toàn bộ các module và class từ thư viện Ursina.
- `from ursina.editor.level_editor import *:` Nhập toàn bộ module liên quan đến trình chỉnh sửa cấp độ từ Ursina.

2. HÀM MAIN()



{ 8 Sync Dev }

```
def main():
    app = Ursina(vsync=False)
```

- `def main():`: Đây là hàm chính, nơi mọi thứ bắt đầu.
- `app = Ursina(vsync=False)`: Tạo một ứng dụng Ursina với vsync tắt. `vsync` (Vertical Synchronization) giúp đồng bộ hóa tốc độ khung hình của ứng dụng với tốc độ làm mới của màn hình, nhưng trong trường hợp này, nó bị tắt để tăng hiệu suất.

3. LỚP TREE



{ 8 Sync Dev }

```

class Tree(Entity):
    def __init__(self, **kwargs):
        super().__init__(**kwargs)
        self.model = 'cube'
        self.color = color.brown

        self.top = Entity(name='tree_top', parent=self, y=1.5,
model='cube', color=color.green, selectable=True)
        LEVEL_EDITOR.entities.append(self.top)

```

- `class Tree(Entity):`: Khai báo lớp Tree kế thừa từ Entity, là một đối tượng cơ bản trong Ursina.
- `def __init__(self, **kwargs):`: Hàm khởi tạo của lớp, cho phép truyền các tham số linh hoạt.
- `super().__init__(**kwargs)`: Gọi hàm khởi tạo của lớp cha (Entity).
- `self.model = 'cube'`: Thiết lập mô hình hình học của đối tượng là một hình lập phương.
- `self.color = color.brown`: Đặt màu của thân cây là màu nâu.
- `self.top = Entity(...)`: Tạo phần ngọn cây với màu xanh, đặt nó là con của đối tượng Tree.
- `LEVEL_EDITOR.entities.append(self.top)`: Thêm phần ngọn cây (tree_top) vào danh sách các thực thể trong trình chỉnh sửa cấp độ.

4. KHỞI TẠO TRÌNH CHỈNH SỬA CẤP ĐỘ



{ 8 Sync Dev }

```
level_editor = LevelEditor()
```

- `level_editor = LevelEditor()`: Tạo một đối tượng trình chỉnh sửa cấp độ (LevelEditor), cho phép người dùng tạo và chỉnh sửa các đối tượng trong không gian 3D.

5. THÊM CÁC LỚP VÀO MENU TRÌNH CHỈNH SỬA



{ 8 Sync Dev }

```
from ursina.prefabs.first_person_controller import  
FirstPersonController  
level_editor.class_menu.available_classes |= {'WhiteCube':  
WhiteCube, 'EditorCamera':EditorCamera,  
'FirstPersonController':FirstPersonController}
```

- `from ursina.prefabs.first_person_controller import FirstPersonController`: Nhập lớp FirstPersonController, cung cấp khả năng di chuyển góc nhìn như trong các game bắn súng góc nhìn thứ nhất.
- `level_editor.class_menu.available_classes |= ...`: Thêm các lớp (WhiteCube, EditorCamera, FirstPersonController) vào menu lớp của trình chỉnh sửa cấp độ. Người dùng có thể chọn và thêm các đối tượng này vào cảnh trong trình chỉnh sửa.

6. CẤU HÌNH VÀ CHẠY ỨNG DỤNG



{ 8 Sync Dev }

```
from ursina.shaders import ssao_shader  
window.borderless=False  
app.run()
```

- `from ursina.shaders import ssao_shader`: Nhập shader SSAO (Screen Space Ambient Occlusion) để tạo hiệu ứng bóng tối dựa trên không gian màn hình, giúp cải thiện chất lượng hình ảnh.
- `window.borderless=False`: Thiết lập cửa sổ không viền (fullscreen không viền), cho phép người dùng di chuyển và thay đổi kích thước cửa sổ.

- `app.run()`: Chạy ứng dụng Ursina.

7. ĐIỂM BẮT ĐẦU CỦA CHƯƠNG TRÌNH



{ 8 Sync Dev }

```
if __name__ == '__main__':
    main()
```

- `if __name__ == '__main__':`: Kiểm tra xem file này có phải là file chính được thực thi không. Nếu đúng, hàm `main()` sẽ được gọi để bắt đầu chương trình.

CỘNG ĐỒNG

KHÓA HỌC:



[Kevin Nguyễn](#)



[Fullstack Python](#)



[Nhóm Chia Sẻ Công Nghệ](#)



[Fullstack Nextjs](#)



[Nhóm BlockChain](#)



[Fullstack Android-IOS](#)



[Tiktok: 8 Sync](#)



[Youtube: 8 Sync Dev](#)



[Zalo](#)

Tài liệu sẽ được cập nhật định kì và thông báo trong group nên các bạn chú ý nhen .

Các khóa học khác xem tại : [8 Sync Dev](#)

Hoặc liên hệ trực tiếp qua fb: [Kevin Nguyễn](#) để được hỗ trợ

