# TextBasedGame.py

# Author: Mariana Zornes

import sys

# Function to display instructions

def show\_instructions():

print("Welcome to the Cursed Temple Adventure Game!")

print("Collect all 6 relics to defeat the Guardian Spirit and escape the temple without being caught.")

print("Commands:")

print(" go North, go South, go East, go West")

print(" get [item name]")

print(" quit - to exit the game at any time\n")

# Function to show player status

def show\_status(current\_room, inventory, rooms):

print(f"You are in the {current\_room}")

print(f"Inventory: {inventory}")

# Show item in the room if available

if "item" in rooms[current\_room]:

print(f"You see {rooms[current\_room]['item']}")

print("----------------------------")

# Function to restart the game

def restart\_game():

replay = input("Would you like to play again? (yes/no): ").strip().lower()

if replay == 'yes':

main()

else:

print("Thanks for playing! Goodbye!")

sys.exit()

# Main function to handle the game

def main():

# Dictionary linking rooms and items within each room

rooms = {

'Entrance Hall': {'North': 'Treasure Vault', 'South': 'Tomb of the Pharaoh', 'East': 'Hall of Mirrors', 'West': 'Garden of Statues', 'item': 'No Relic'},

'Treasure Vault': {'South': 'Entrance Hall', 'East': "Guardian's Lair", 'item': ‘Guardian's Lair Key’},

"Guardian's Lair": {'West': 'Treasure Vault', 'item': 'No Relic'}, #Villain's room

'Garden of Statues': {'East': 'Entrance Hall', 'item': 'Emerald Necklace'},

'Chamber of Secrets': {'South': 'Hall of Mirrors', 'item': 'Ancient Scroll'},

'Forgotten Library': {'West': 'Tomb of the Pharaoh', 'item': 'Crystal Skull'},

'Tomb of the Pharaoh': {'East': 'Forgotten Library', 'North': 'Entrance Hall', 'item': 'Scepter of the Pharaoh'},

'Hall of Mirrors': {'West': 'Entrance Hall', 'North': 'Chamber of Secrets', 'item': 'Golden Idol'}

}

# Initialize game state variables

inventory = []

current\_room = 'Entrance Hall'

relics\_needed = 6 # Number of relics needed to win

game\_over = False

# Show game instructions

show\_instructions()

# Main game loop

while not game\_over:

# Display the current status of the player

show\_status(current\_room, inventory, rooms)

# Get the player's next move

move = input("Enter your move: ").split()

if len(move) < 1:

print("Invalid command. Please enter a command in the form 'go [direction]', 'get [item]', or 'quit'.")

continue

action = move[0].lower()

target = ' '.join(move[1:])

# Handle quit command

if action == "quit":

print("Thanks for playing! Goodbye!")

sys.exit()

# Handle room movement

elif action == "go":

# Check if movement direction is valid

if target in rooms[current\_room]:

current\_room = rooms[current\_room][target]

# Check if player has entered the villain's room

if current\_room == 'Guardian’s Lair':

if len(inventory) == relics\_needed:

print("You have all relics and can now defeat the Guardian Spirit!")

print("Congratulations! You have escaped the temple with all relics!")

game\_over = True

else:

print("You encountered the Guardian Spirit without all relics! Game Over.")

game\_over = True

else:

print(f"You can't go {target} from {current\_room}. Check for valid directions.")

# Handle item collection

elif action == "get":

# Check if the item is in the current room and matches the target item

if "item" in rooms[current\_room] and rooms[current\_room]["item"].lower() == target.lower():

if target not in inventory:

# Add item to inventory and remove it from the room

inventory.append(rooms[current\_room]["item"])

print(f"{rooms[current\_room]['item']} has been added to your inventory.")

del rooms[current\_room]["item"]

# Check if all relics have been collected

if len(inventory) == relics\_needed:

print("You have collected all relics! Find the Guardians Lair to defeat the Guardian.")

else:

print("Item already in inventory.")

else:

print(f"'{target}' doesn't seem quite right. Make sure you are entering the correct command or word.")

# Handle invalid commands

else:

print("Invalid command. Try again.")

# Prompt replay if game over

if game\_over:

restart\_game()

# Run the game

if \_\_name\_\_ == "\_\_main\_\_":

main()