Platform: Beginner

Knight's Quest in the Jungle

In this game, the Knight is on a quest to rescue a captured wizard from the dragon's lair located in a jungle. The player controls the Knight, who moves left and right using arrow keys and can jump over obstacles (like logs and mud puddles). The Knight can collect magical items (like potions) to gain points. When the Knight touches the wizard, the wizard changes his costume to a 'freed' version, and a victory sound plays. If the Knight collides with the dragon, he loses a life and must restart at a checkpoint.

Key Characters

- Knight
- Wizard
- Dragon

Guiding Prompts

- You are creating learning materials for students on Scratch coding.
  The following project idea have been presented with Jungle,Mansion theme:
  It a fun game
  The project has 3 characters: Knight,Wizard,Dragon.

Please check if they are age appropriate for elementary and middle school students. If the project asked is not age appropriate, give the reason and suggest some alternatives projects ideas.

- You are creating learning materials for students on Scratch coding.
  The students would like a project in Jungle,Mansion theme as follows:
  It a fun game
  The project has 3 characters: Knight,Wizard,Dragon
  This is a BEGINNER level project, focus on these concepts when providing solutions, do not use hard concepts: Simple motion, sound, switch costumes, repeat, conditional
  Create 2 project ideas using the theme, characters, and key storyline students have provided above, focusing on the specified scratch concepts.

Sprite Ideas

Knight

Wizard

Dragon

Step-By-Step Tutorial

Step 1: Create the Knight character and set up basic movement.

Start by creating a Knight sprite. Use the arrow keys to move the Knight left and right. Add a jump feature using the space bar. This will involve changing the y-position of the Knight to simulate jumping.

```
when green flag clicked
forever
  if <key [right arrow v] pressed?> then
    change x by (10)
  end
  if <key [left arrow v] pressed?> then
    change x by (-10)
  end
  if <key [space v] pressed?> then
    repeat (10)
      change y by (10)
    end
    repeat (10)
      change y by (-10)
    end
  end
end
```

Step 2: Add obstacles and magical items.

Create sprites for obstacles like logs and mud puddles. Use a simple conditional to check if the Knight is touching these obstacles and make the Knight jump over them. Also, create magical item sprites like potions that the Knight can collect to gain points.

```
when green flag clicked
forever
  if <touching [log v]?> then
    change y by (10)
  end
  if <touching [potion v]?> then
    change [score v] by (10)
    hide
  end
end
```

Step 3: Create the Wizard and Dragon characters.

Design a Wizard sprite with two costumes: one for captured and one for freed. Create a Dragon sprite that the Knight must avoid. If the Knight touches the Dragon, he loses a life and restarts at a checkpoint.

```
when green flag clicked
if <touching [wizard v]?> then
  switch costume to [freed v]
  play sound [victory v] until done
end

when green flag clicked
if <touching [dragon v]?> then
  broadcast [lose life v]
end
```

Step 4: Set up checkpoints and life system.

Create a variable for lives and set up checkpoints where the Knight will restart if he loses a life. Use a broadcast message to handle life loss and restarting.

```
when I receive [lose life v]
change [lives v] by (-1)
if <(lives) = [0]> then
  stop all
else
  go to [checkpoint v]
end
```

Step 5: Add victory conditions and sounds.

When the Knight reaches the Wizard and frees him, switch the Wizard's costume and play a victory sound. This signals the end of the game.

```
when green flag clicked
if <touching [wizard v]?> then
  switch costume to [freed v]
  play sound [victory v] until done
end
```

Potential Problems

- Ensuring the Knight's jump is smooth and returns to the ground correctly without getting stuck in the air.
- Making sure the Knight's collision detection with obstacles and items is accurate, so the game responds correctly.
- Balancing the difficulty of the game so that it is challenging but not too hard for beginners.

Extra Challenges

- Add a timer to increase the difficulty, requiring the Knight to complete the quest within a certain time limit.
- Introduce a new type of obstacle, such as a moving platform, that the Knight must navigate.