

PRACTICE MATERIAL FOR SPELLATHON

(GRADE 9)

1. Develop a Strategy Game

Task: Create a simple turn-based strategy game where players control units to achieve objectives.

Steps:

1. Design sprites for different units (e.g., soldier, tank) and a map background.
2. Use variables to track player health, resources, and turn count.
3. Implement movement controls using arrow keys or mouse clicks.
4. Use "if" statements to determine attack outcomes based on unit stats and player decisions.
5. Include win/lose conditions and display the results at the end of the game.

2. Create a Weather Simulation

Task: Simulate weather patterns over time, allowing users to interact with the simulation.

Steps:

1. Use sprites for sun, clouds, rain, and snow, with a backdrop representing a landscape.
2. Create variables for temperature, humidity, and wind speed.
3. Implement a "forever" loop to change weather conditions based on variable values.
4. Use "if" statements to trigger weather changes (e.g., if temperature is low, show snow).
5. Allow user interaction to set weather parameters and see the effects.

3. Build a Simple E-commerce Store

Task: Create a virtual store where users can browse products and make purchases.

Steps:

1. Design sprites for different products and a store backdrop.
2. Use lists to store product names, prices, and quantities.
3. Implement "ask" blocks to let users select products and input quantities.
4. Use variables to track the total cost and available stock.

5. Provide feedback on successful purchases and update inventory accordingly.

4. Create a Data Visualization Project

Task: Design a project that visualizes data (e.g., a bar graph) based on user input.

Steps:

1. Use sprites to represent different data categories (e.g., different colors for bars).
2. Implement "ask" blocks to gather data points from the user.
3. Use a list variable to store the data values.
4. Create a function to draw the bars based on the input values using loops.
5. Display the total and average of the data inputted at the end.

5. Build a Quiz with User Feedback

Task: Design a quiz that gives immediate feedback on answers and tracks user performance.

Steps:

1. Use "ask" blocks for questions and multiple-choice answers.
2. Create variables to track the score and total questions.
3. Implement "if" statements to check answers and provide feedback (correct/incorrect).
4. Use "broadcast" messages to move to the next question.
5. At the end, display the user's score and offer options to retake the quiz or view explanations for answers.