

PRACTICE MATERIAL FOR SPELLATHON

(GRADE 12)

1. Create a Physics Simulation

Task: Design a simulation that demonstrates fundamental physics concepts like motion, gravity, and friction.

Steps:

1. Choose a backdrop and sprites for objects (e.g., a ball, a slope).
2. Use variables to represent position, velocity, and acceleration.
3. Implement motion algorithms using equations of motion (e.g., $s=ut+\frac{1}{2}at^2$).
4. Use "if" statements to check for collisions and apply forces (e.g., bouncing off walls).
5. Allow users to adjust parameters like mass and friction to see the effects in real time.

2. Build a Social Media Platform Simulation

Task: Create a basic simulation of a social media platform where users can post, like, and comment.

Steps:

1. Design sprites for user avatars, posts, and interaction buttons.
2. Use lists to store posts and user interactions (likes, comments).
3. Implement "ask" blocks for users to create new posts and add comments.
4. Use variables to track the number of likes and display them on the posts.
5. Include functionalities for users to see the most liked posts or trending topics.

3. Develop an Escape Room Game with Puzzles

Task: Create an interactive escape room game that requires players to solve multiple puzzles to "escape."

Steps:

1. Design multiple backdrops representing different rooms.
2. Create interactive objects (e.g., keys, locks, puzzles).
3. Use "ask" blocks to gather user input for solving puzzles.

4. Implement "if" statements to check if puzzles are solved and transition to the next room.
5. Track time taken to escape and display the result at the end.

4. Make a Data Analysis Tool

Task: Develop a tool that allows users to input data and visualize it (e.g., bar charts, pie charts).

Steps:

1. Use "ask" blocks to gather numerical data from users.
2. Store the data in a list variable.
3. Create a function to draw graphs based on the input data using loops.
4. Implement options for different types of graphs (e.g., bar, line).
5. Allow users to compare datasets by switching between graphs.

5. Design a Simple AI Chatbot

Task: Create an advanced chatbot that can have more dynamic conversations and learn from user input.

Steps:

1. Choose a sprite for the chatbot and set up a friendly backdrop.
2. Use "when green flag clicked" to initialize the conversation.
3. Implement an array or list to store common questions and responses.
4. Use "if/else" statements to match user input with keywords for relevant responses.
5. Allow the bot to learn new responses by adding user-inputted answers to its list.