Junior Knights Assignments: Practice Game Jam

Task A: Pong

In this task, you will recreate the classic game of pong. If you haven't played before, it's simply two paddles on either side of the screen that move up and down and a ball that gets hit back and forth.

Specifications:

- Two players in the game. One on each side of the screen (left/right)
    - Players should only be allowed to move up and down.
    - You can make this a two player game or try making a simple AI for the other player to play against.
- A ball that starts in the middle of the screen
    - Make it so that the ball randomly picks a side to move towards at the start of the round.
    - Give the ball movement in the y direction as well so there's a reason to move up and down to hit it.
    - When the ball hits the top/bottom of the screen, it should bounce off it.
    - When the ball hits the left/right side of the screen, the opposite player should get a point and the ball will reset.
- Keep track of the score for each player.
    - Put the left players' score on the top left and the right players score on the top right.
    - Set up a win condition like best of 5 or first to 5.
- Display a game over message stating which player won.

Extras:

- If you haven't done so already, clean your code up and place repeated tasks into functions that you can call. (practicing good coding habits)
- Increase the speed of the ball the longer the round goes (this way someone loses the round eventually)
- If you didn't try implementing a simple AI for the second player, try doing so.

Task B: Snake

In this task, you will recreate the classic game of snake. If you haven't played before, it's a game in which you start off controlling a single block snake and every time you eat an apple you grow one block longer. The point of the game is to grow the snake to as big a size as possible.

Specifications:

- Start the snake off in the middle of the screen.
    - Can be in a random location if you would like
- Place the apple on the screen.
    - This should be in a function that will randomly assign the apple a location, then draw it on the screen.
- Whenever the snake eats an apple, it should grow in size by one block.
- The snake should move automatically in the direction of the last pressed directional key.
    - Ex. if you last hit left, the snake keeps moving left until another direction key is pressed or they hit the side of the screen and lose.
- Lose conditions:
    - If the snakes head hits his body, the player loses
    - If the snakes head tries to go out of bounds (off screen) the player loses.
- Keep track of the score (how many apples have been eaten) on the screen.
- Display a game over message when the player loses and display their score.

Extras:

- If you haven't done so already, clean your code up and place repeated tasks into functions that you can call. (practicing good coding habits)
- Try making a snake class that stores the location of the head and the rest of its body parts.
    - You can make a method that moves the body of the snake.
    - You can make a method that draws the snake onto the screen.
- Try making an apple class that stores the location of the current apple and the number of apples eaten.
    - You can make a method that can be called to randomly assign a new location for the apple.
    - You can make a method that draws the apple onto the screen.
    - You can make a method that draws the score onto the screen.