**Application Requirements:**

1. Has a label displaying the current sprite image

2. Has a slider that controls the frames per second playing of the animation in a range from 1 to 60 frames per second (FPS)

1. Research how to set the minimum and maximum value of the slider

2. Research how to get some tick marks displayed on the side of the slider

3. Changing the slider does not make the animation start - the Start button does that.

3. Has a label that displays the text "Frames per second"

4. Has another label next to the previous one that displays the current FPS and updates when

the slider changes.

5. Has a Start/Stop button.

1. When "Start" is displayed on the button and the button is pushed, the sprite should

animate at the current FPS. The button text should change to "Stop."

2. When "Stop" is displayed and the button is pushed, the sprite should stop animating and

the button text should change to "Start".

6. Has a menu with 2 options

1. Pause. When the Pause menu item is selected, the animation should stop and the

Start/Stop button should display "Start".

2. Exit. When the Exit menu item is selected, the program should quit.

**To run the application:**

* + Put the code in a folder.
  + Ensure that the sprite images are in the same folder as the code.
  + Ensure that the spiteimages folder is named spriteImages
  + Ensure to have pyqt5 installed.

Start the terminal in the same folder as the code. and type the following on the terminal:

**python SpritePreviewer.py**

**Testing the Application**

That should start the application and you can following the given video to play around with the application. The application should fulfill the following requirements as per the video.

* Pressing Start and seeing the sprite animate and the button text update
* Changing the slider to see the playback speed change and the FPS label update
* pressing Stop and seeing the animation stop
* Pressing Start and then using the File pause option to see the animation stop and the

button text update

* Pressing Start after the File pause to see the animation resume
* Additionally, pressing the Menu > Exit should close the application.

**Assignment Discussions.**

In the codebase comments have been added for the instance variables, the calculation for the frames per second.

As per for the building of the app. It was easier to add all the UI components first and the add signals to the buttons, sliders and the menu one by one and testing. This is as opposed to having all the signals set in the beginning and testing all of them at once.

There is a counter that keeps track of index of the currently displayed image and then fetches the next image based on that index until all sprites are animated. When it gets to the end, the counter resets again.