

CS 115, Autumn 2021
Programming Project #8: Paint (20 points)
Due Friday, December 3rd, 11:30 PM

Special thanks to Marty Stepp for parts of this specification.

This assignment focuses on creating GUIs and event driven programming. Turn in files named `paint.py`, `stamps.py` and `backgrounds.py`. You will need `ECGUI.py`. Make sure you download a new copy of it! Save all of these files in the same folder.

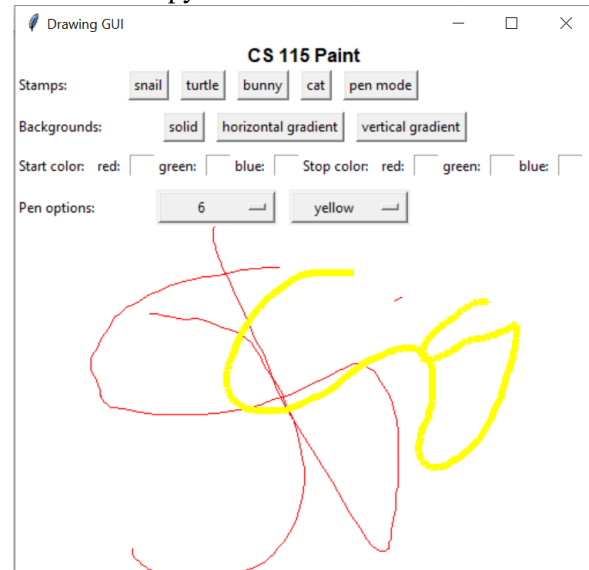
Program Description:

In this project you will be creating a simple painting program. This painting program must allow the user to select options in a graphical interface and alter the appearance of a drawing canvas by clicking on it.

GUI Appearance:

Your GUI must contain at least the following:

- A title at the top telling the user what this program is.
- Controls to select each of your stamp animals from project 3
- Controls to select a solid background color, horizontal background gradient or vertical background gradient
- Controls to allow the user to specify red, green and blue values for both the start and the stop of the gradient.
- A control to allow the user to select pen stroke width.
- A control to allow the user to select pen color.



All input boxes should be labeled so that the user can tell what they are expected to enter in them. Each group of similar controls should be labeled as well.

You are welcome to select different controls than the ones pictured in the screenshot above. For example, you could choose to use radio buttons for the stamp or pen selected. However, make sure you select a control type that makes sense for what you want the user to do. Don't display text in entry boxes as the user can alter the text in those. Also don't use an entry box when the user has a specific set of options (the background type, for example).

You are welcome to space your controls however you like as long as your layout meets the following criteria:

- Groups of related controls should appear together. For example, all stamp options should appear near each other on the interface.
- Your interface must have at least three separate rows that contain multiple inputs. You may not position everything in one row or one column.
- Your interface must have three different input types. You cannot use entry boxes or buttons for everything.
- A canvas must take up at least 50% of your window.

Implementation Guidelines:

Although this project may look pretty large at first, you have already written most of the code.

Use the code you wrote for project 3 for the stamps. Place each animal in its own function. Store these functions in `stamps.py`. This file will not have a `main`. Instead, you can access and call its functions in your `paint.py` file by writing:

```
from stamps.py import *
```

at the top of `paint.py` and then calling your animal functions the way you would call any other function.

Use the code you wrote for the project 6 gradients for the background gradient options. Store your gradient code in `backgrounds.py`. Import and use the functions in this file just as described for `stamps.py` above.

Start your pen drawing code with the code we wrote together in class.

This means that the contents of your complex functions are almost all already written. What you need to do is connect them up with the GUI.

Development Strategy

There is a lot of code for this project so it can feel a bit daunting at first. Following the steps outlined below will help keep you on track. Run your code after each step to make sure you have accomplished it successfully.

1. Copy your animal code into stamps.py and put each animal in its own function.
2. Copy your gradient code into backgrounds.py and put it in one or two functions.
3. Create a GUI that just opens a window.
4. Add a title and a drawing canvas to the GUI.
5. Add a button for each of your stamps.
6. Add a mouse pressed reaction to the canvas so that it draws the snail at the x, y location of your mouse whenever you click the canvas.
7. Add a variable to control the mode. This variable should be set to a different value when each of the stamp buttons is pressed.
8. Add some logic to your mouse pressed reaction so that it checks the mode and draws the animal of the most recently selected mode instead of always drawing the snail.
9. Add a pen mode and incorporate the pen drawing code we wrote together in class.
10. Add inputs for pen weight (line thickness) and pen color. Use the values the user selects from these to alter the thickness and color of the line the user draws.
11. Add background selection inputs and inputs where the user can specify starting and stopping red, green and blue values.
12. The background should change immediately when any of the background buttons are pressed. To make this happen, you should draw on the canvas immediately when they are clicked. You shouldn't wait for the user to click on the canvas. Start with the solid color. Then use the values the user types into the start and stop color inputs to call your gradient functions for the others.
13. Arrange your controls into rows, columns or other groups.
14. Add any appearance adjustments you would like to your GUI.

Style Guidelines:

For full credit on this project, every user interaction must happen through the GUI. You should definitely print to the console to help yourself debug but the finished product should have no console output.

We will check carefully for redundancy on this project. If you have a very similar piece of code that is repeated several times in your program, eliminate the redundancy such as by creating a function, by using loops and/or by factoring `if/else` code as described earlier this quarter.

Give meaningful names to functions, variables, and parameters. Follow Python's naming standards as specified in lecture. Limit the lengths of your lines to fewer than 80 characters. Include meaningful comment headers at the top of your program and at the start of each function.

You are welcome to use more advanced Python that you have learned elsewhere on this project. The only constraints are that your program must meet the requirements listed above, it cannot depend on having anything installed besides default Python and `ECGUI.py` and it must be your own work.