Readme!

Please feel free to edit this template by adding / editing and removing slides. I’ve tried to cover most of what you might want to include but there are many ways of providing evidence so don’t be afraid to delete slides that you don’t need.

Slides with a green background should be used to show evidence of planning / project management. The slide with a purple background should be duplicated and used to show ‘relevant implications’ evidence.

The instructions in the grey boxes should be deleted once they have been read. Copies of the instructions are underneath the slides in the speaker notes (just in case they are needed at a later stage).

91906 & 91907 Complex Programming

Please add links to your work on this slide. Then remove this instruction. Note that most of the instructions have been supplied as speaker notes for later reference :) **Ensure your Github repositories, Trello board, and videos are set to public**.

Trello board / Project Management: Trello **Make PUBLIC**

The project on Github: <https://github.com/ethanleevers/91906_ethan> **Make PUBLIC**

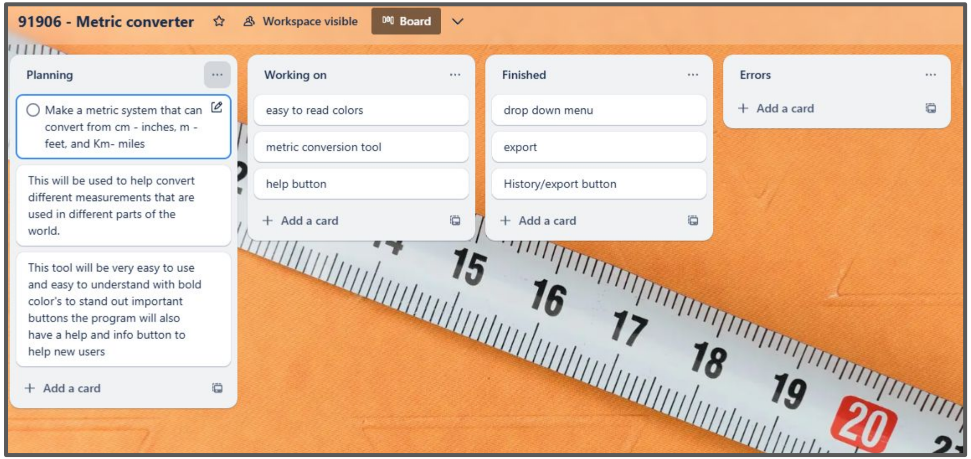
Final Project Filename: <ie: **metric_converter_v4** >

Final Testing Video: <insert link here> **Make PUBLIC**

Project Management

I am using the Agile methodology because it helps me work in small, manageable steps and adapt quickly to any changes or feedback. This approach allows me to continuously improve the project throughout the development process, rather than waiting until the end. By reviewing and adjusting regularly, I can ensure the final product better meets the needs of the users and stays aligned with the project goals. Agile also supports better time management and keeps the project flexible and responsive this methodology works perfectly with trello seen on next slide

I chose to use Trello as my project management tool because it is simple, clear, and easy to use. It allows me to break down my project into smaller tasks and organize them using lists and cards, which helps me stay on track and avoid feeling overwhelmed. The drag and drop system makes it easy to update progress by moving tasks between columns like 'working on' 'errors' and 'Finished' This visual layout makes it easier to see what needs to be done next and helps with prioritising tasks. Trello has also improved my time management by allowing me track my progress in one place. Overall it helps me stay focused organised, and efficient while working through my project



91906 - Metric converter

Planning

- ○ Make a metric system that can convert from cm - inches, m - feet, and Km- miles
- This will be used to help convert different measurements that are used in different parts of the world.
- This tool will be very easy to use and easy to understand with bold color's to stand out important buttons the program will also have a help and info button to help new users

Working on

- easy to read colors
- metric conversion tool
- help button

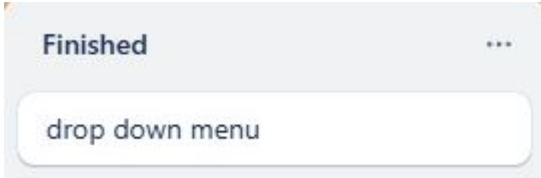
Finished

- drop down menu
- export
- History/export button

Errors

Project Management - Sprint Tracking

Task	Sprint Number	Start Date	End Date
Metric system dropdown	1	28/05/25	2/05/25



Finished

drop down menu

What are you working on?	What did you achieve?	What are the next steps?
A drop down bar so users can select their conversion system easily this also helps prevent errors and human typing errors	Before the drop down bar users did not have a choice on what to convert and if a typing system was imputed there would be chance for human errors and typos	to add a effect to the dropdown menu options. When the user hovers over an option it will be highlighted.

Evaluation - what worked well & what did not?
Adding a single dropdown menu to the converter improved its flexibility and usability by letting users choose the unit they want to convert from. The original version had a fixed starting unit, which limited the user’s options and made the tool less adaptable.

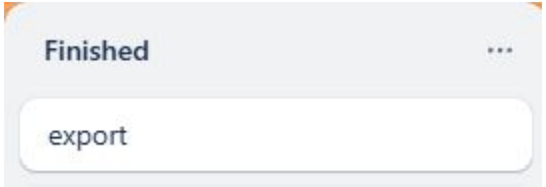
Project Management - Sprint Tracking

Task	Sprint Number	Start Date	End Date
Export data	2	04/05/2025	07/05/2025

What are you working on?	What did you achieve?	What are the next steps?
I'm making a new part of my metric conversion program that lets users save their past conversions into a file. This means they can keep a copy of what they've changed, like from centimeters to inches or kilometers to miles.	I got the file-saving feature to work. Now the program can remember the user's conversions and save them into a file. This means users can look at their past conversions later or share the file if needed.	I plan to improve the Export Data menu by using high-contrast, consistent colors to enhance readability and visual appeal. Key buttons like "Export" will be bold and vibrant, with hover effects for better user feedback.

Evaluation - what worked well & what did not?

The file-saving feature worked effectively, allowing users to store and revisit their previous conversions, which added convenience and functionality. However, the original Export Data menu lacked visual clarity, so I improved it by applying high-contrast colors and clearer formatting to make it easier to read and more user-friendly.

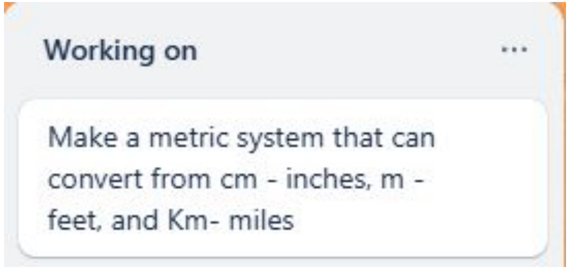


Finished

export

Project Management - Sprint Tracking

Task	Sprint Number	Start Date	End Date
Metric converter	4	28/04/25	6/05/25



Working on

Make a metric system that can convert from cm - inches, m - feet, and Km- miles

What are you working on?	What did you achieve?	What are the next steps?
I'm developing a converter that accurately switches values between units like inches and centimeters, feet and meters, and miles and kilometers, ensuring precise results for each conversion.	I successfully developed a working converter that delivers accurate results when switching between different units. I made sure the conversion constants used were precise to at least two decimal places, ensuring reliability and consistency in the outputs.	The next step is to implement error handling so that if a user enters something other than a numbers such as letters or symbols the program displays a clear error message instead producing incorrect results

Evaluation - what worked well & what did not?
The converter accurately switches between units using precise units, but it lacks error handling. If users enter letters or symbols, it doesn't respond properly. Next, I'll add helpful error messages for invalid input

Addressing Relevant Implications

One key implication I considered in my program was usability, which refers to how easily users can achieve their goals with minimal effort. Usability was important because if the program was difficult or confusing to use, users might not get the results they needed, making it unfit for its purpose. I addressed this by making the program simple and clear. I avoided using complex language and designed an easy-to-follow layout. I also tested the program thoroughly and fixed any errors to improve the user experience. For example, the questions are straightforward and suitable even for primary school students, such as "Centimeters to inches," "Meters to feet," and "Kilometers to miles." The buttons and prompts are short and precise, helping the program run smoothly and ensuring user satisfaction.

Addressing Relevant Implications

One key implication for my program is functionality, meaning it must perform as intended without any errors that could cause it to crash or behave unexpectedly.

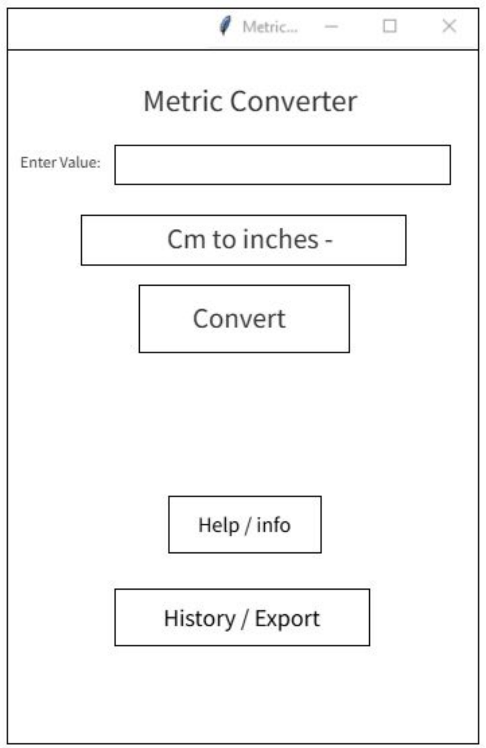
Functionality is crucial because if the program doesn't work properly, users may lose trust in it or stop using it altogether, making it ineffective and unusable. To address this, I tested different types of inputs and fixed any issues to ensure the program responds correctly in all situations.

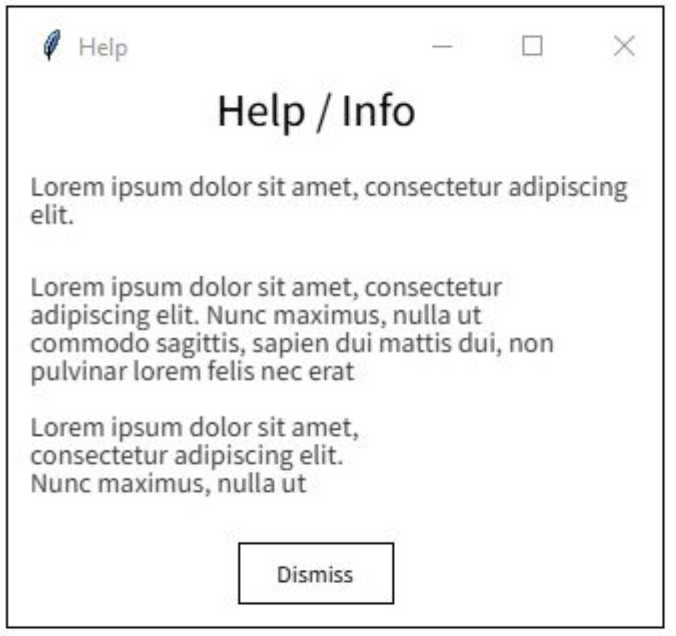
For instance, when a user enters a number, the program gives the correct output for a positive integer (e.g., 100), a negative number (e.g., -100). This helps the program run smoothly and keeps users satisfied.

Addressing Relevant Implications

I applied to my metric converter was aesthetics. Aesthetics refers to how visually appealing, organized, and user-friendly the program looks. This is important because a clean, well-designed interface makes the converter easier to use and helps users feel confident in its accuracy. I addressed this by using bold, centered headings with clear fonts to make the title easy to read. I added large buttons in bright, contrasting colours like green, red and blue so users can quickly find the unit they want to convert. I used padding between buttons to keep them spaced and reduce clutter and kept the background a white - gray so the text and buttons stand out clearly. The layout uses a grid to keep all conversion buttons in one row and places the Help/info button above the History/export button. Centered to give a balanced, symmetrical look. These design choices improve user satisfaction by making the converter feel simple and intuitive without needing lots of instructions.

Graphical User Interface Design...

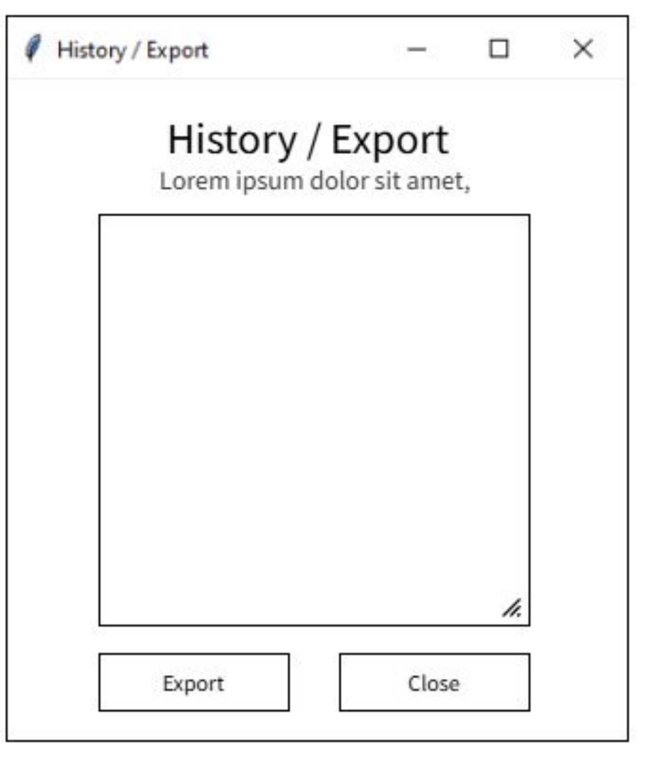




Lorem ipsum dolor sit amet, consectetur adipiscing elit.

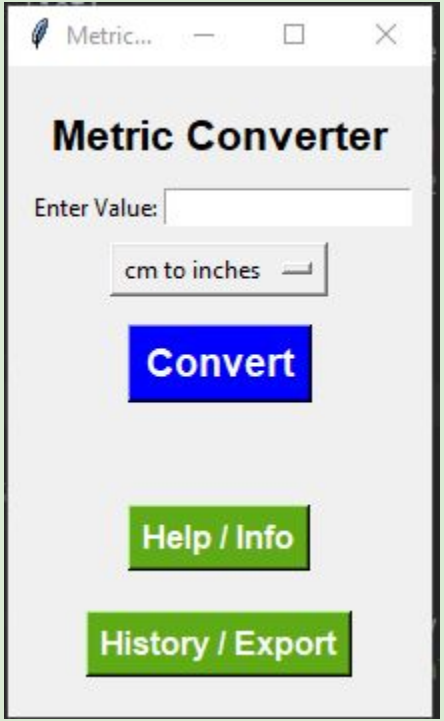
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc maximus, nulla ut commodo sagittis, sapien dui mattis dui, non pulvinar lorem felis nec erat

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.
Nunc maximus, nulla ut



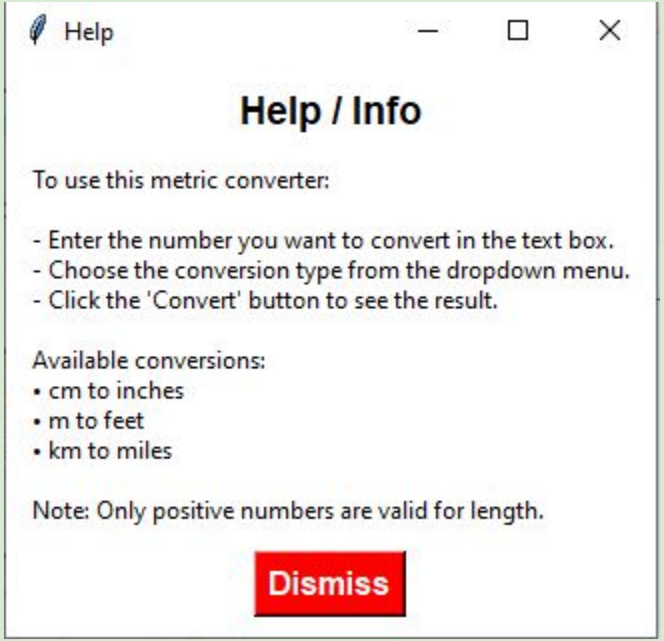
Lorem ipsum dolor sit amet,

Graphical User Interface Design...



Enter Value:

cm to inches



Help / Info

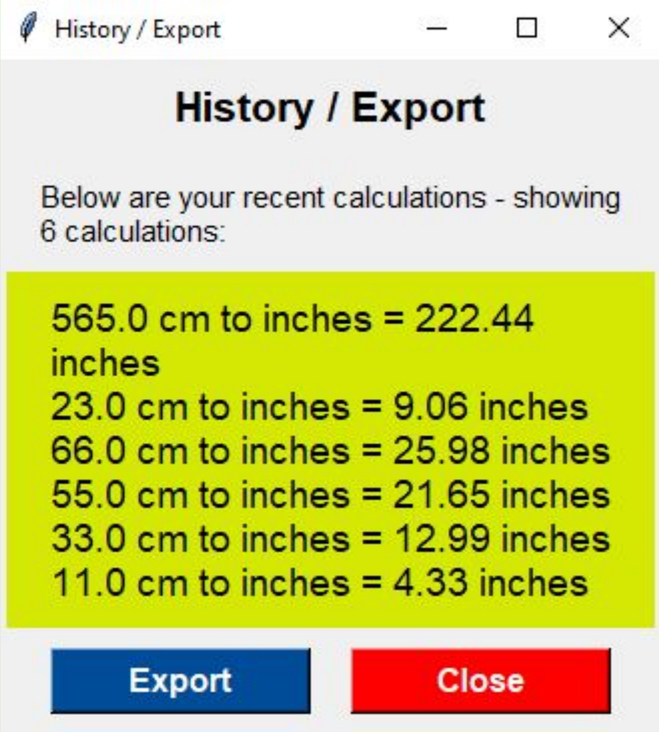
To use this metric converter:

- Enter the number you want to convert in the text box.
- Choose the conversion type from the dropdown menu.
- Click the 'Convert' button to see the result.

Available conversions:

• cm to inches
• m to feet
• km to miles

Note: Only positive numbers are valid for length.

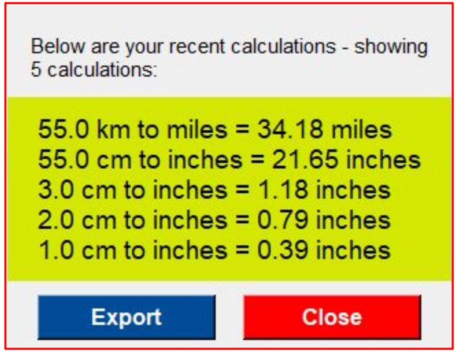


History / Export

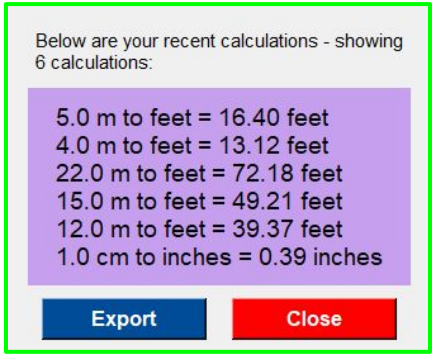
Below are your recent calculations - showing 6 calculations:

```
565.0 cm to inches = 222.44 inches
23.0 cm to inches = 9.06 inches
66.0 cm to inches = 25.98 inches
55.0 cm to inches = 21.65 inches
33.0 cm to inches = 12.99 inches
11.0 cm to inches = 4.33 inches
```

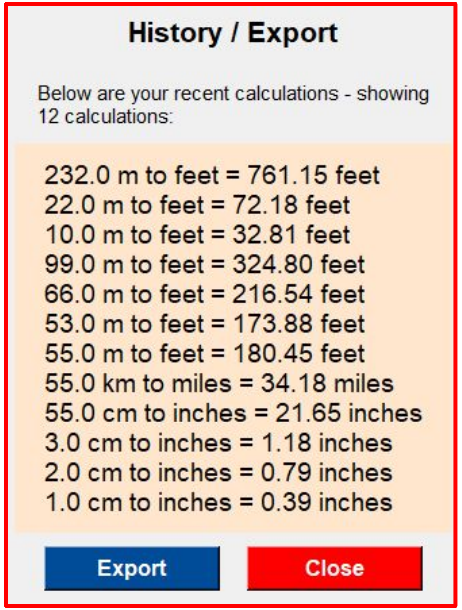
improvements


Below are your recent calculations - showing 5 calculations:

```
55.0 km to miles = 34.18 miles
55.0 cm to inches = 21.65 inches
3.0 cm to inches = 1.18 inches
2.0 cm to inches = 0.79 inches
1.0 cm to inches = 0.39 inches
```

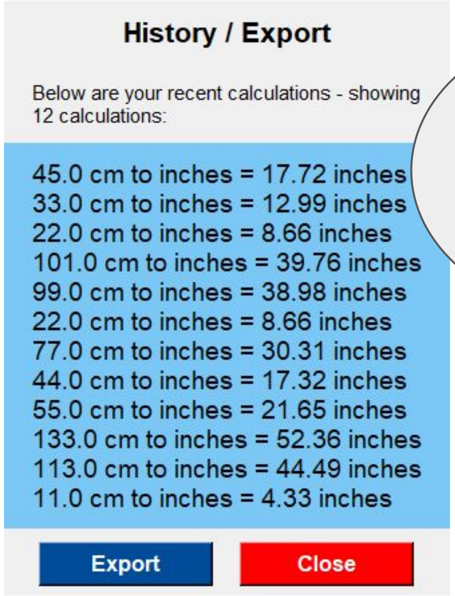

Below are your recent calculations - showing 6 calculations:

```
5.0 m to feet = 16.40 feet
4.0 m to feet = 13.12 feet
22.0 m to feet = 72.18 feet
15.0 m to feet = 49.21 feet
12.0 m to feet = 39.37 feet
1.0 cm to inches = 0.39 inches
```


History / Export

Below are your recent calculations - showing 12 calculations:

```
232.0 m to feet = 761.15 feet
22.0 m to feet = 72.18 feet
10.0 m to feet = 32.81 feet
99.0 m to feet = 324.80 feet
66.0 m to feet = 216.54 feet
53.0 m to feet = 173.88 feet
55.0 m to feet = 180.45 feet
55.0 km to miles = 34.18 miles
55.0 cm to inches = 21.65 inches
3.0 cm to inches = 1.18 inches
2.0 cm to inches = 0.79 inches
1.0 cm to inches = 0.39 inches
```


History / Export

Below are your recent calculations - showing 12 calculations:

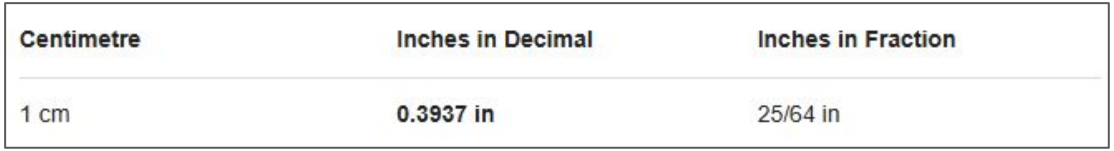
```
45.0 cm to inches = 17.72 inches
33.0 cm to inches = 12.99 inches
22.0 cm to inches = 8.66 inches
101.0 cm to inches = 39.76 inches
99.0 cm to inches = 38.98 inches
22.0 cm to inches = 8.66 inches
77.0 cm to inches = 30.31 inches
44.0 cm to inches = 17.32 inches
55.0 cm to inches = 21.65 inches
133.0 cm to inches = 52.36 inches
113.0 cm to inches = 44.49 inches
11.0 cm to inches = 4.33 inches
```

Changed the export background colour based on the number of calculations light green yellow for fewer than 10 and warm orange-yellow for 10 or more calculations.

Updated the colours to a pleasant magenta for fewer than 10 conversions and a calm blue for 10 or more

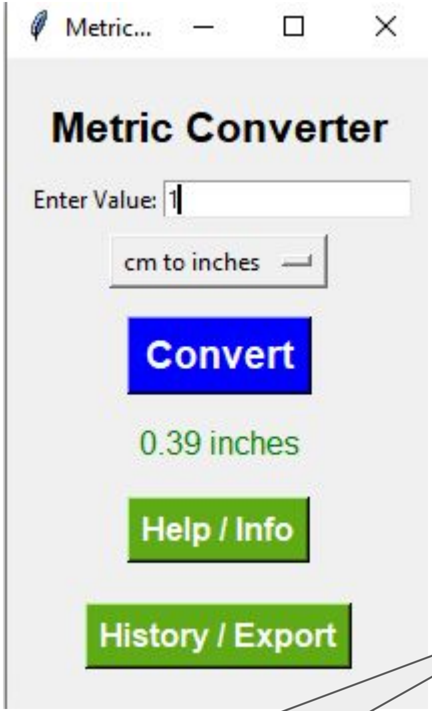
This is to make it more noticeable and easier for the user to understand

Testing



Centimetre	Inches in Decimal	Inches in Fraction
1 cm	**0.3937 in**	25/64 in

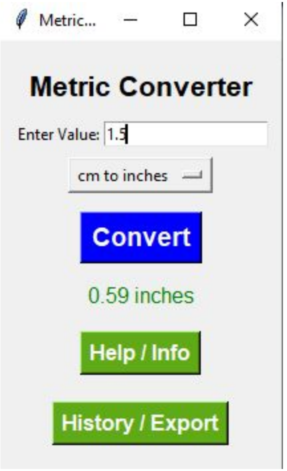
From google conversion



Enter Value: 1

cm to inches

0.39 inches



Enter Value: 1.5

cm to inches

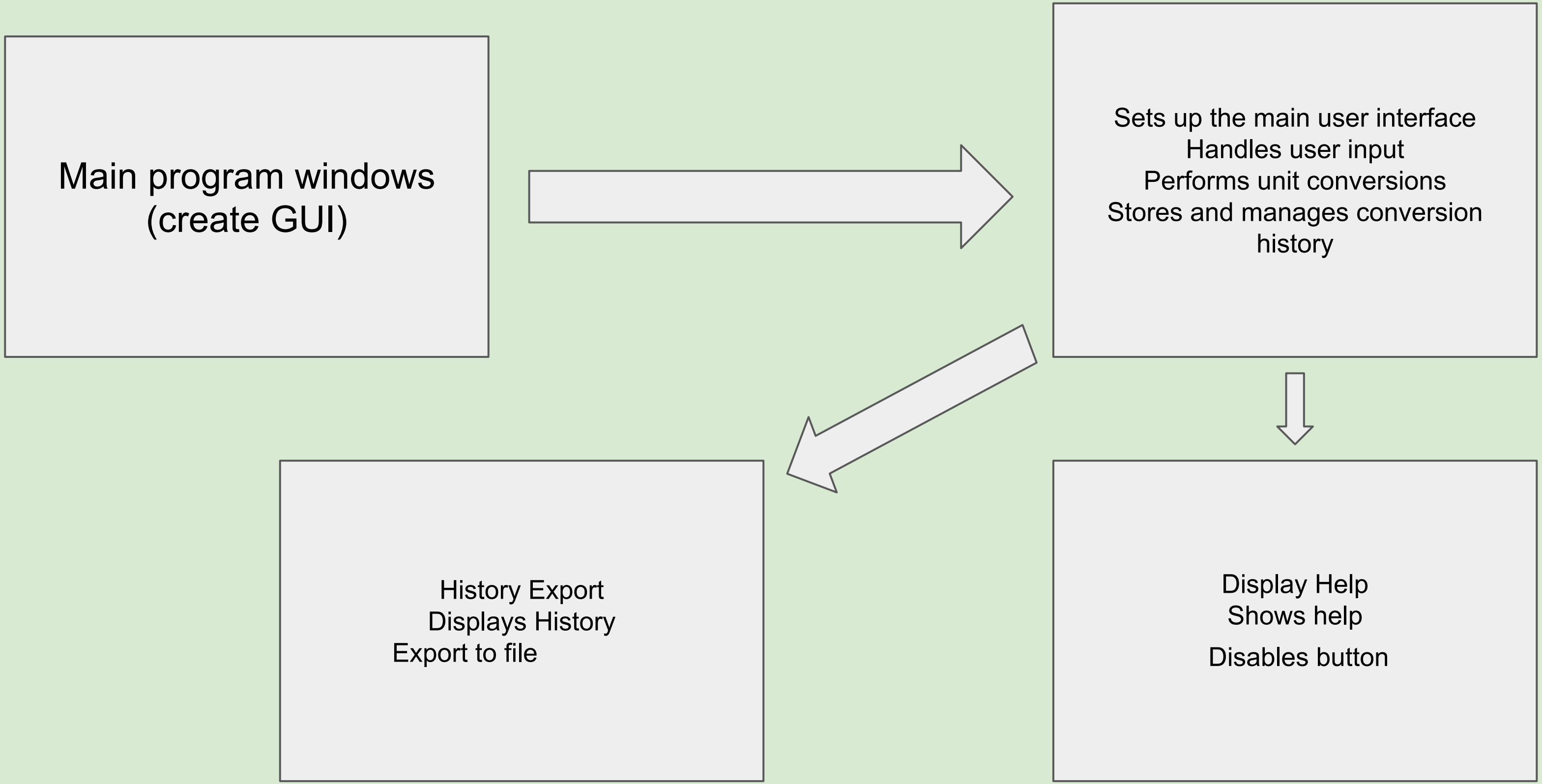
0.59 inches

Made sure decimals could also be used

I used Google to validate my conversion code by searching for similar conversions. This helped me validate my conversion rates and confirm my code worked correctly. Google was a big help in refining and verifying my work.

Test Data	Expected Result
0.1-10000	Program converts
0	Please enter a number 0.1 or greater
-10	Please enter a number 0.1 or higher

Program Structure



Main program windows
(create GUI)

Sets up the main user interface
Handles user input
Performs unit conversions
Stores and manages conversion history

History Export
Displays History
Export to file

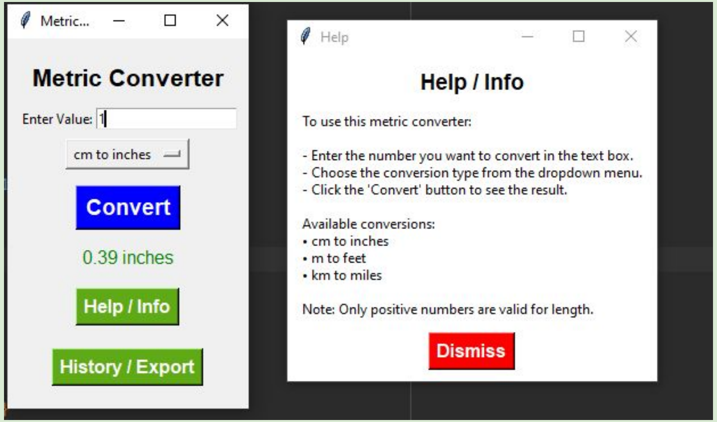
Display Help
Shows help
Disables button

Problem Decomposition

Provide evidence showing that you have decomposed the task. This can be in the form of a trello screenshot or a list of components. If necessary, you may revisit this slide and add to it / edit it as you create your outcome. When you make changes to this slide, please do so in a different colour, date the changes and explain why they were made (this can help provide evidence for M / E grades).

Hint: Use the structure you developed earlier to work out what components you need. For each function, you should have at least one component.

Problem Decomposition



0.39 inches

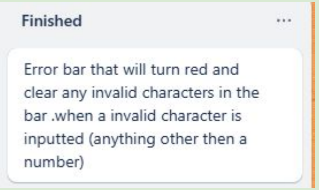
To use this metric converter:

- Enter the number you want to convert in the text box.
- Choose the conversion type from the dropdown menu.
- Click the 'Convert' button to see the result.

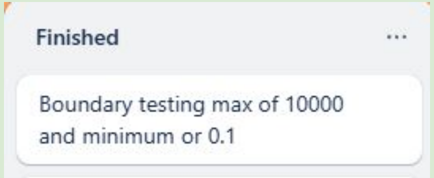
Available conversions:
• cm to inches
• m to feet
• km to miles

Note: Only positive numbers are valid for length.

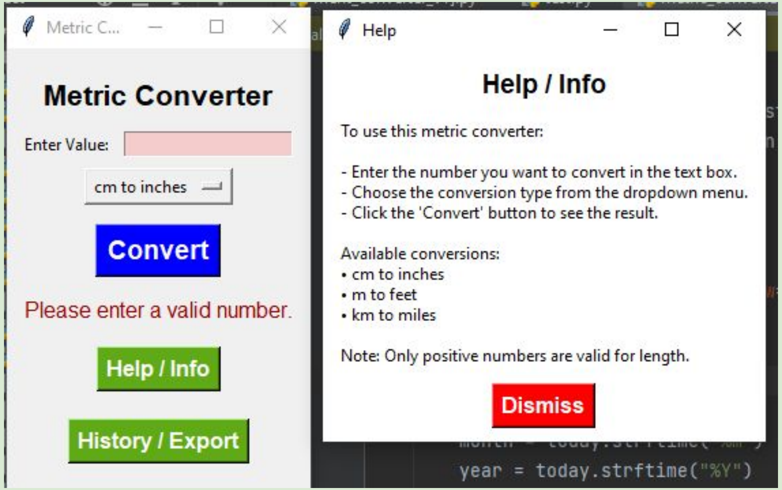
Test Data	Expected Result
abc	Please enter a valid number
!@#%^&	Please enter a valid number
0.0	Please enter a number 0.1 or higher
0.1-10000	Program continues



Error bar that will turn red and clear any invalid characters in the bar .when a invalid character is inputted (anything other then a number)



Boundary testing max of 10000 and minimum or 0.1



Please enter a valid number.

To use this metric converter:

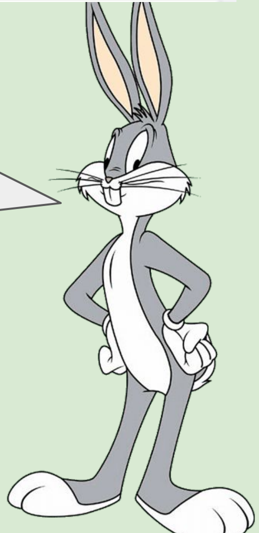
- Enter the number you want to convert in the text box.
- Choose the conversion type from the dropdown menu.
- Click the 'Convert' button to see the result.

Available conversions:
• cm to inches
• m to feet
• km to miles

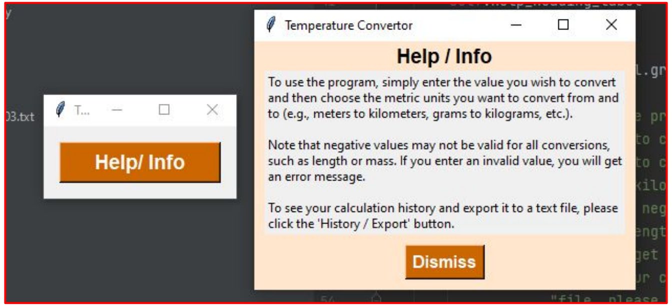
Note: Only positive numbers are valid for length.

```
year = today.strftime("%Y")
```

I updated the input system so that only valid entries are allowed. If an invalid character is entered, a red bar appears, and the input field is cleared to help prevent repeated mistakes.



improvements

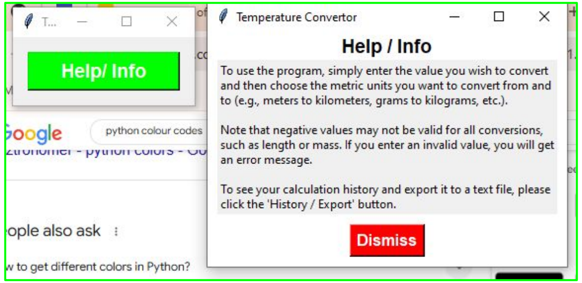


Help / Info

To use the program, simply enter the value you wish to convert and then choose the metric units you want to convert from and to (e.g., meters to kilometers, grams to kilograms, etc.).

Note that negative values may not be valid for all conversions, such as length or mass. If you enter an invalid value, you will get an error message.

To see your calculation history and export it to a text file, please click the 'History / Export' button.

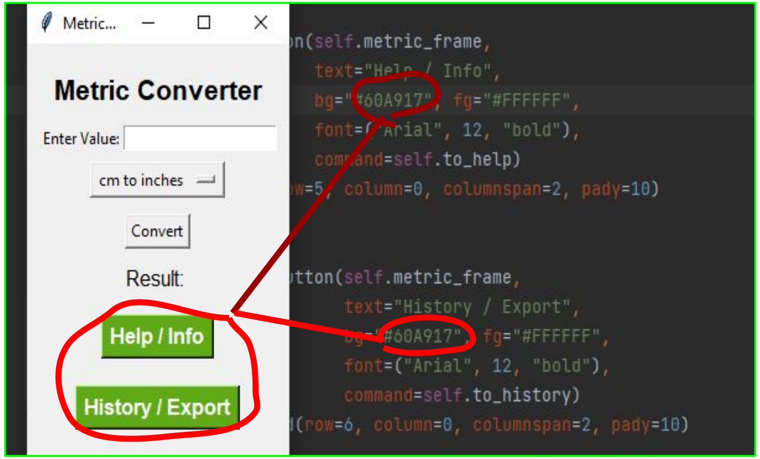


Help / Info

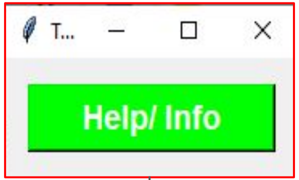
To use the program, simply enter the value you wish to convert and then choose the metric units you want to convert from and to (e.g., meters to kilometers, grams to kilograms, etc.).

Note that negative values may not be valid for all conversions, such as length or mass. If you enter an invalid value, you will get an error message.

To see your calculation history and export it to a text file, please click the 'History / Export' button.



```
n(self.metric_frame,
   text="Help / Info",
   bg="#60A917", fg="#FFFFFF",
   font=("Arial", 12, "bold"),
   command=self.to_help)
w=5, column=0, columnspan=2, pady=10)


tton(self.metric_frame,
     text="History / Export",
     bg="#60A917", fg="#FFFFFF",
     font=("Arial", 12, "bold"),
     command=self.to_history)
d(row=6, column=0, columnspan=2, pady=10)
```

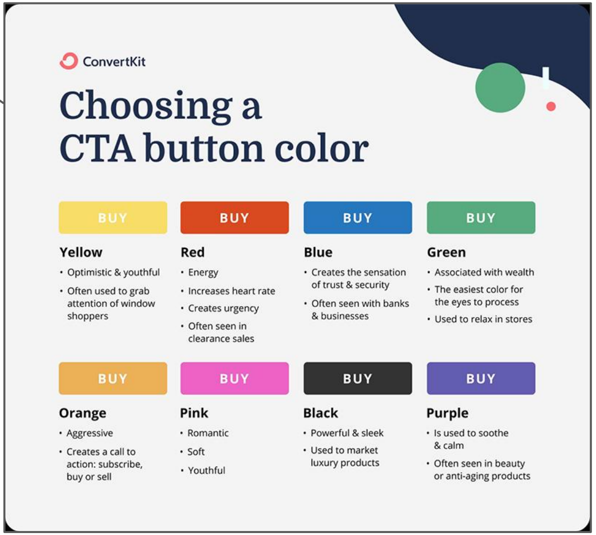


I changed the colors to better fit my project, switching from yellow and orange to green, white, and red, as they align more with the overall color scheme.

Changed colour from a harsh neon green to a darker green (#60A917)

Component Planning...

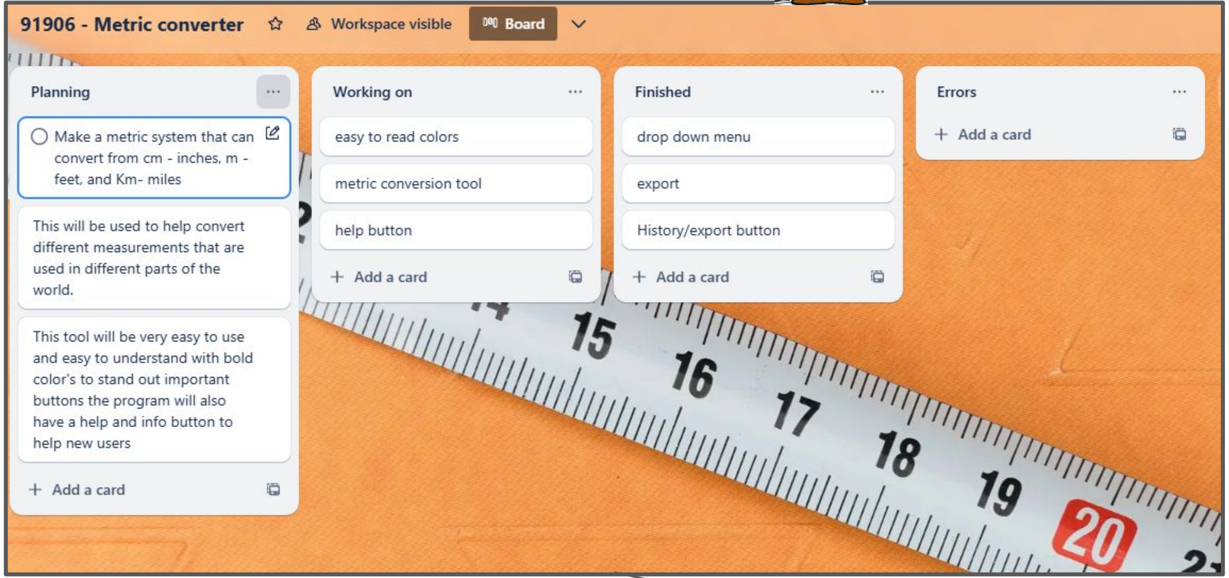
By researching different colors, I was able to choose the best ones for my buttons



ConvertKit

Choosing a CTA button color

BUY	BUY	BUY	BUY
Yellow • Optimistic & youthful • Often used to grab attention of window shoppers	**Red** • Energy • Increases heart rate • Creates urgency • Often seen in clearance sales	**Blue** • Creates the sensation of trust & security • Often seen with banks & businesses	**Green** • Associated with wealth • The easiest color for the eyes to process • Used to relax in stores
BUY	BUY	BUY	BUY
Orange • Aggressive • Creates a call to action: subscribe, buy or sell	**Pink** • Romantic • Soft • Youthful	**Black** • Powerful & sleek • Used to market luxury products	**Purple** • Is used to soothe & calm • Often seen in beauty or anti-aging products



91906 - Metric converter ☆ Workspace visible Board

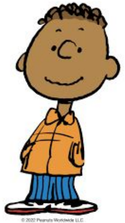
Planning
- ○ Make a metric system that can convert from cm - inches, m - feet, and Km- miles
- This will be used to help convert different measurements that are used in different parts of the world.
- This tool will be very easy to use and easy to understand with bold color's to stand out important buttons the program will also have a help and info button to help new users
- + Add a card

Working on
- easy to read colors
- metric conversion tool
- help button
- + Add a card

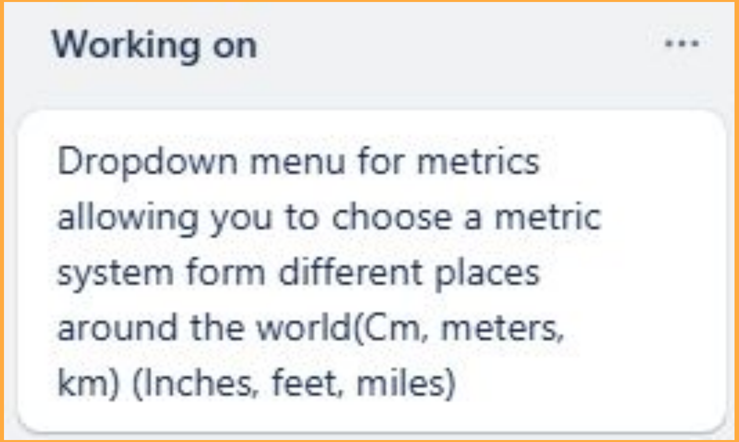
Finished
- drop down menu
- export
- History/export button
- + Add a card

Errors
- + Add a card

Using this system in Trello helps me keep track of what I need to work on and what has been completed. It also outlines a clear plan to ensure I don't get lost or confused. The system is constantly being updated to include new plans, ongoing tasks, or current issues that need to be resolved.



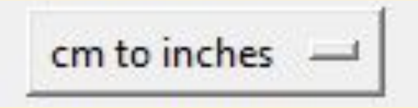
Component Trialling - Dropdown

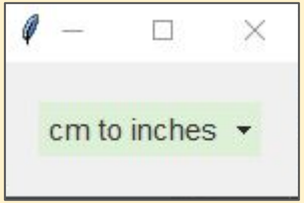


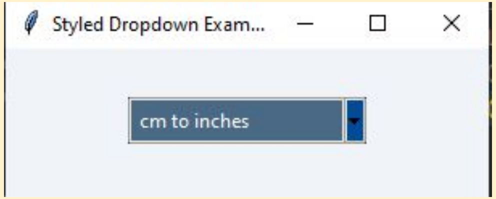
Working on

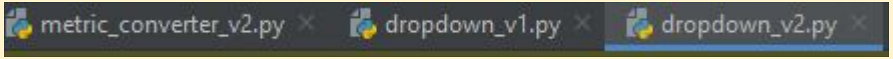
Dropdown menu for metrics allowing you to choose a metric system form different places around the world(Cm, meters, km) (Inches, feet, miles)

Component Trialling - Dropdown

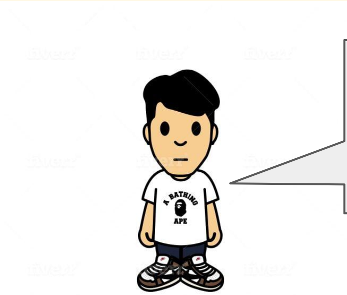






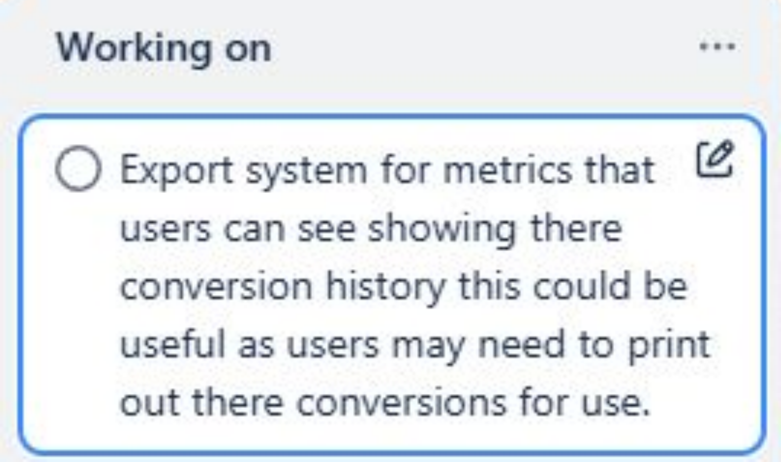


I decided to go with version 3 because it looked the best and worked smoothly. Creating three versions also helped me stay organized and clearly see the strengths and weaknesses of each design before making my final choice.



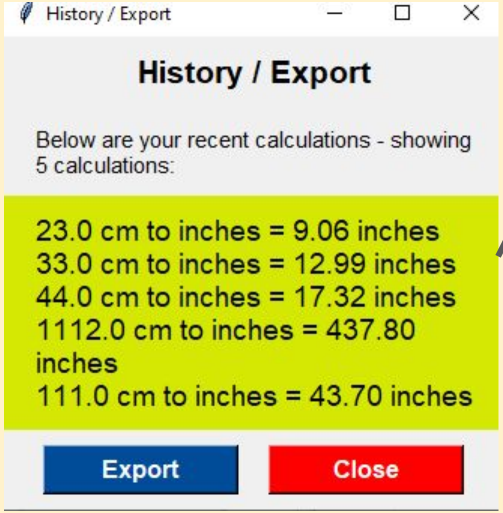
I created three different versions of the dropdown bar, each with a unique design. This allowed me to test which layout looked the best and felt most user-friendly.

Component Trialling - Export



○ Export system for metrics that users can see showing there conversion history this could be useful as users may need to print out there conversions for use.

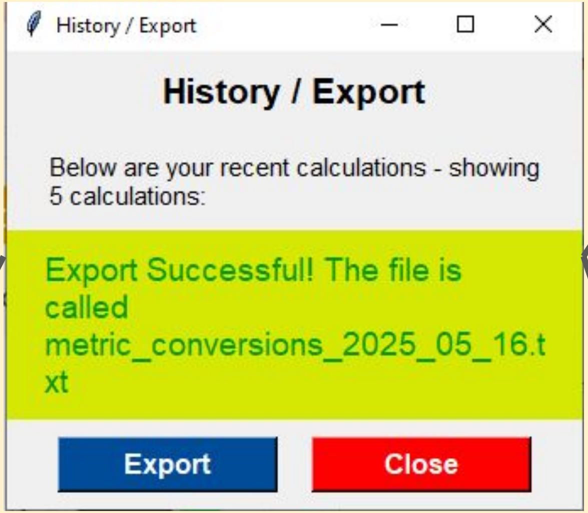
Component Trialling - Export



History / Export

Below are your recent calculations - showing 5 calculations:

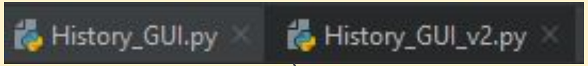
23.0 cm to inches = 9.06 inches
33.0 cm to inches = 12.99 inches
44.0 cm to inches = 17.32 inches
1112.0 cm to inches = 437.80 inches
111.0 cm to inches = 43.70 inches

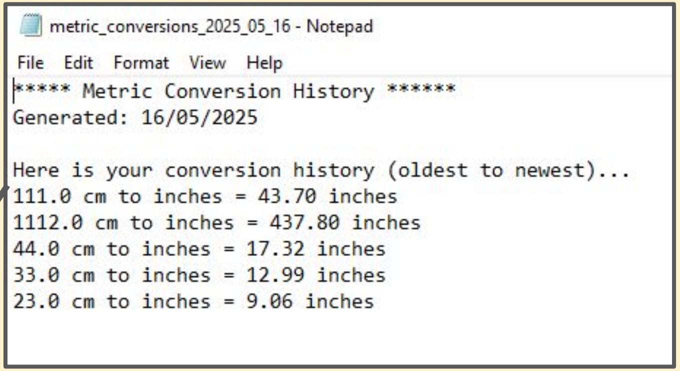


History / Export

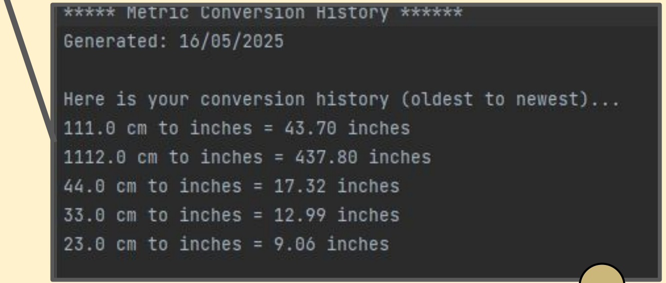
Below are your recent calculations - showing 5 calculations:

Export Successful! The file is called metric_conversions_2025_05_16.txt





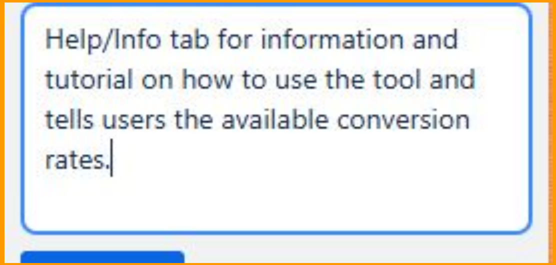
```
***** Metric Conversion History ******
Generated: 16/05/2025

Here is your conversion history (oldest to newest)...
111.0 cm to inches = 43.70 inches
1112.0 cm to inches = 437.80 inches
44.0 cm to inches = 17.32 inches
33.0 cm to inches = 12.99 inches
23.0 cm to inches = 9.06 inches
```



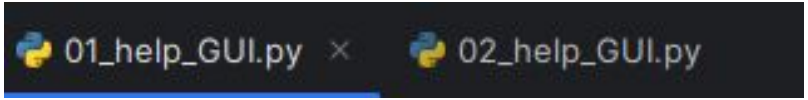
```
***** Metric Conversion History ******
Generated: 16/05/2025

Here is your conversion history (oldest to newest)...
111.0 cm to inches = 43.70 inches
1112.0 cm to inches = 437.80 inches
44.0 cm to inches = 17.32 inches
33.0 cm to inches = 12.99 inches
23.0 cm to inches = 9.06 inches
```

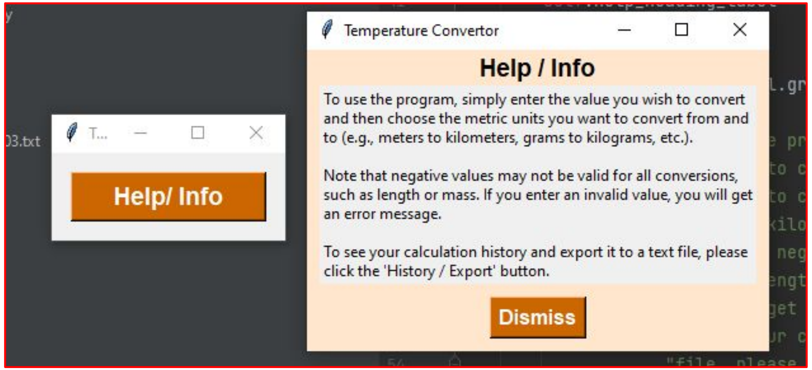
As part of trialling my data exporter, I tested whether the export button accurately saved the calculation history to a text file. I verified that the file contained the correct data in the right order and that the filename included the current date. I also modified the export button colour and layout to ensure it was noticeable without being too distracting.

Component Trialling - help / info



Help/Info tab for information and tutorial on how to use the tool and tells users the available conversion rates.



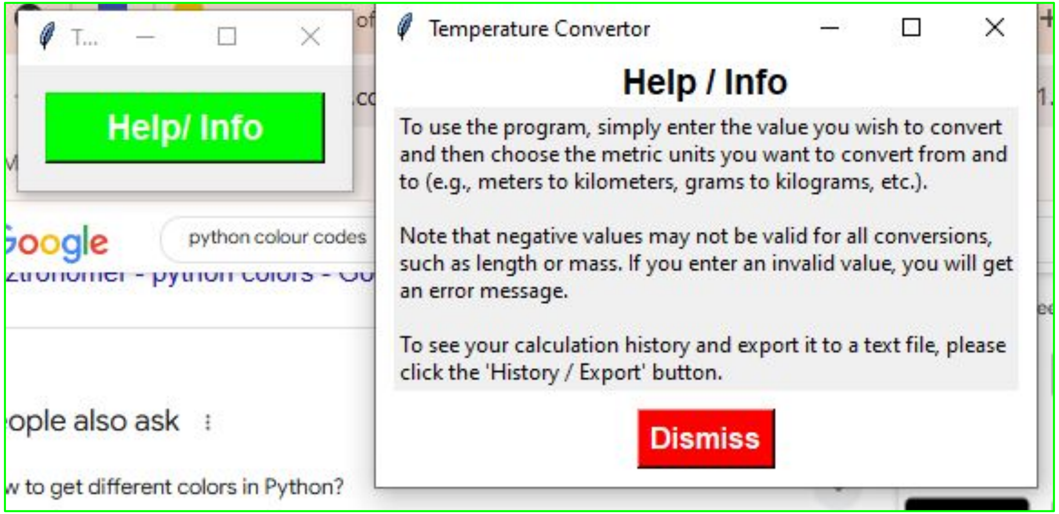


Help / Info

To use the program, simply enter the value you wish to convert and then choose the metric units you want to convert from and to (e.g., meters to kilometers, grams to kilograms, etc.).

Note that negative values may not be valid for all conversions, such as length or mass. If you enter an invalid value, you will get an error message.

To see your calculation history and export it to a text file, please click the 'History / Export' button.



Help / Info

To use the program, simply enter the value you wish to convert and then choose the metric units you want to convert from and to (e.g., meters to kilometers, grams to kilograms, etc.).

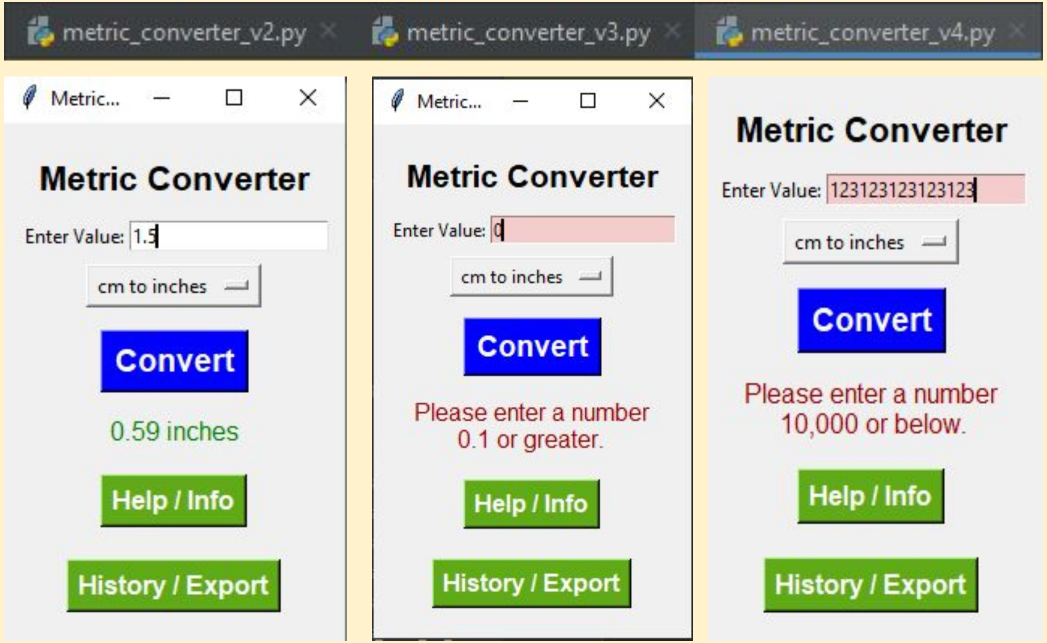
Note that negative values may not be valid for all conversions, such as length or mass. If you enter an invalid value, you will get an error message.

To see your calculation history and export it to a text file, please click the 'History / Export' button.

When trialling my Help/Info button. I initially used a bright orange background with black text, but found it too dull. This experience showed me how important color balance is for a consistent interface. I then adjusted it to a white theme with green and red buttons, as those colours stand out without overwhelming the design.

Use slides with this background colour to show your trialling evidence.

Component Trialling



Metric Converter

Enter Value: 1.5

0.59 inches

Metric Converter

Enter Value: 0

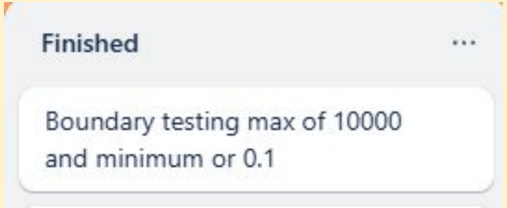
Please enter a number 0.1 or greater.

Metric Converter

Enter Value: 123123123123123

Please enter a number 10,000 or below.

Test Data	Expected Result
0	Please enter a number 0.1 or greater
1.5	Program converts
10000	Program converts
10001	Please enter a number 10,000 or below
xyz	Program converts
-10	Please enter a number 0.1 or greater

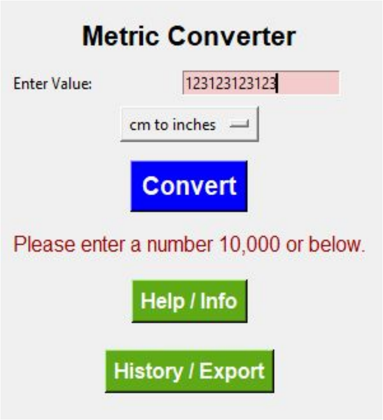


Finished

Boundary testing max of 10000 and minimum or 0.1

Debugging Evidence <Edit Me>

During component trialling, I found a bug where users could enter very long numbers in the input field, which messed up the layout of the GUI. To fix this, I added a limit in the function to allow a maximum of 10000. If the input goes over this limit, the program shows an error message and stops the conversion. This fix makes the program more reliable and keeps the interface clean and easy to read.

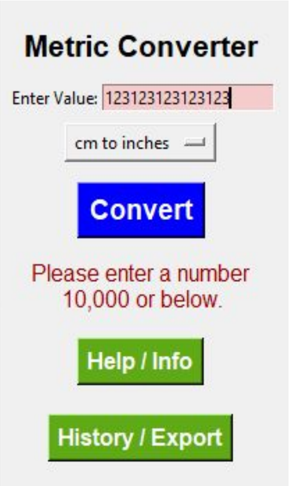


Metric Converter

Enter Value: 123123123123

cm to inches

Please enter a number 10,000 or below.



Metric Converter

Enter Value: 123123123123123

cm to inches

Please enter a number 10,000 or below.

Debugging Evidence <Edit Me>

I identified a significant issue when clicking the **Export** button it failed to save the conversion history to a file as expected.

Upon investigating the export functionality, I discovered that the problem was from how the file was being opened and written to specifically, the program wasn't properly handling the file.

To resolve this, I modified the `export_data()` function to ensure it correctly opens a text file with the current date in the filename and writes each conversion entry from the list into the file.

I validated the fix by exporting varying numbers of conversions and verifying the contents of the saved file. This enhancement ensures that users can reliably save their data, thereby improving usability and aligning with the program's intended functionality.

Component Test Plan <Edit Me>

Create a test plan for this component BEFORE you start coding. Your plan should allow you to test all logical pathways for this component. It should also include test cases for relevant boundary and unexpected values.

Data input	Expected output
xyz	Please input a number.
0	= 0 Inches/Feet/Miles
-100	= -x Inches/Feet/Miles
100	= x Inches/Feet/Miles
5.5	= x.x Inches/Feet/Miles

Complete Program <test plan>

Use	cm to inches	100	Shows: "100cm = 39.37 inches"
	m to feet	2	Shows: "2m = 6.56 feet"
	km to miles	5	Shows: "5 km = 3.11 miles"
Boundary	Zero value	0	Shows: “please enter a number 0.1 or greater”
	Very small number	0.0001	Please enter a number 0.1 or grater
	Number max	10000	Works, shows result
Invalid	Letters only	abc	Error message shown
	Symbols	!@%^	Error message shown
	Empty box	(nothing)	Error message shown
UI/Buttons	Click Help	Click	Help window pops up
	View History after converting	Click	Last results shown
	Export conversions	Click	File saved, confirmation shown

Complete Program...

Assemble your components into a working program. On the slides that follow, please provide a test plan and evidence that your program works as expected.

If you are going for M / E, you also need to create extra slides showing how you have used your testing to improve the functionality of your program. This could mean having multiple test plans (and screenshots) showing several iterations of the assembled program.

Complete Program...

While putting all parts of my program together, I found a bug during testing. Users could enter very long numbers, which made the layout messy and hard to read.

To fix this, I changed the program to allow a maximum number of 10,000. If a user tries to enter a number bigger than that, the program shows an error message and stops the conversion. I tested this by entering different types of inputs—short numbers, long numbers, letters, and symbols to make sure the program works correctly.

This change improved the program's appearance and reliability. It also shows how testing helped make the program better by finding and fixing problems, making it easier for users. Now, I have added a test for maximum number input in my test plan.

Complete Program <testing evidence>

https://drive.google.com/file/d/1RS8MCUvN1D14xwJghOaoV1f-zGkYVC4Z/view?usp=sharing

Complex Processes - Discussion

Discuss how you used (and combined) information gained from the planning, testing and trialling of your components to improve the quality of your program. Note that **synthesising information** from planning, testing and trialling of components and then discussing how this lead to a high quality outcome are needed for an E grade.

To synthesise means to “bring together” to create something new.

To synthesise information, you need to look at the testing/trialling information from different components, or different stages of development, or from different kinds of testing, and bring together this information.

- You might need to compare contradictory information and work out the best solution.
- You could show how different tests all led to one unified conclusion, or led to new ideas
- You might need to come up with whole new ideas not originally tested

Complex Processes - Discussion

When developing my metrics converter, I relied on the planning, testing, and trialling stages to improve the overall quality of the program. During the planning and wireframe creation, I focused on keeping the layout clean, uncluttered, and user-friendly. I included clear and simple buttons, and implemented a working drop down menu to allow users to choose between the three different types of conversions. These decisions helped make the program more intuitive and easy to use.

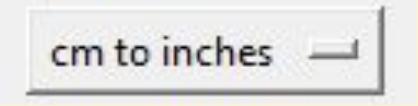
While I was testing, I saw that the program was printing out error messages too long causing the GUI to distort the program to fix this major issue I fixed the error printing message function and tested it to make sure the issue was fixed. I also tested the input bar and found out that users could enter numbers as large or as small as they wanted 0>∞ so to fix this i added a limit of 10000 this helps make sure the program not lag and even crash. With this new feature it makes the program much cleaner and also helps with error prevention

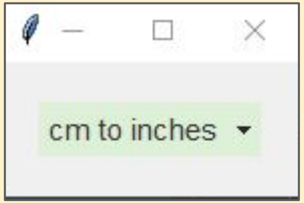
applying what I learned from each stage, I was able to improve both the functionality and design of the program. These improvements helped me develop a final product that runs reliably and provides users with a smooth, intuitive experience.

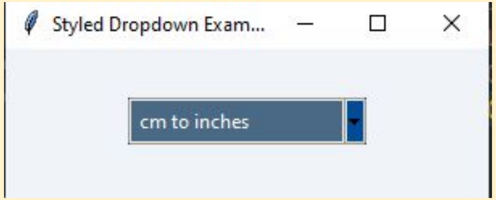
Complex Processes - Discussion - trello

While developing my program, I tested and tried different ideas to make it better. I looked at two planning tools that people online recommended. One was a basic checklist app with few features, and the other was Trello, which uses boards and cards to organize tasks. After trying both, I found that Trello made it much easier to organize tasks, track progress, and make changes as I worked. I chose Trello because it was more flexible and fit better with how I wanted to plan my program. This choice helped everything run smoothly from start to finish. At first, I planned to show all the conversion options on screen as buttons that could be pressed. But during testing, I saw that this made the program look crowded and overstimulating . So, I did some research and tried using a dropdown menu, where options appear only when clicked. This made the layout cleaner and easier to use, so I kept that change.

I also tested different designs for the dropdown, like changing the font size and layout. After trying some styles, I picked the one that was clearest and easiest for users to understand. This testing helped me make better choices and made the final program work well and look professional.







Complex Processes - Discussion - feedback

Feedback was a very important in improving the overall quality and usability of my program. During the development, I always wanted to take advice from both my teacher and classmates, and I used their good ideas to make positive changes that improved both the design and functionality.

Mr. Tuahine pointed out that entering "0" as an input value was ineffective and also suggested removing negative values to prevent unhelpful conversions. In response, I updated the input validation to set a minimum value of 0.1, ensuring the program remained useful and produced relevant outputs.

My classmates feedback was also very important. One tester (Stefan) told me that typing in long numbers made my program look messy and distorted. I took this feedback in and I added a 10000 number input limit to keep the program clean without affecting the program’s usability. Another tester (Ryotaro) mentioned that my programs buttons and there colours were making my program look messy and unorganized so to fix this i changed the buttons shape and size as well as changing their colour doing this further made my program more seamless and more organized.

By taking in the feedback and making these adjustments, I significantly improved the user experience and successfully completed the final version of the program. Looking ahead, I plan to expand its versatility by adding more conversion options and features, such as unit switching and a contact button.