**DRAWING POKEMON CHARACTER PIKACHU IN PYTHON**

TABLE OF CONTENTS

[Introduction 3](#_Toc148195910)

[Discussion 3](#_Toc148195911)

[FlowChart 4](#_Toc148195912)

[Pseudocode 11](#_Toc148195913)

[Code 11](#_Toc148195914)

[Outcome 16](#_Toc148195915)

[Conclusion 16](#_Toc148195916)

[Reference List 17](#_Toc148195917)

[Appendices 18](#_Toc148195918)

# Introduction

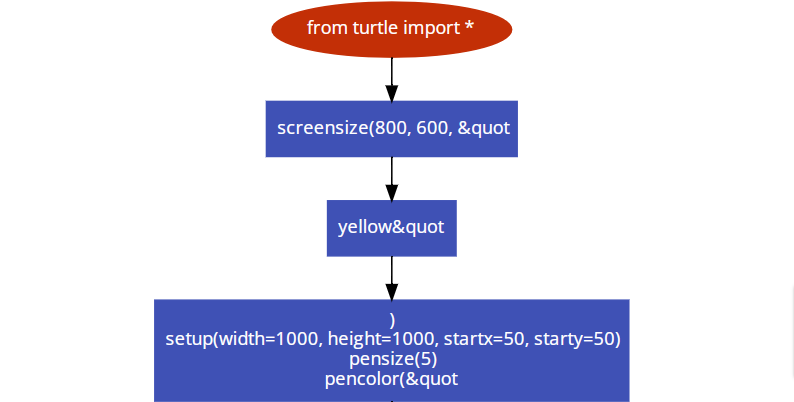
This project is based on the visualization and creation of a Pokemon by Python. The python’s feature of art turtle is used here to create this visualization. To go through the process, one conventional way is to apply pseudocode and flowcharts to define the steps from the initial stage until the end of creating a Pokemon character. The pseudocode and flowcharts generally helped to visualize for the user the inner structure of the code, used inbuild methods with the preferable attributes of the particular character that include the unique identifier of the character, the age of the character, the presence of the character, and the capabilities of the character. These inputs by the user are generally stored in variables for future use. Then the pseudocode and flowcharts may represent the character's backstory or the cause of creating the character. Subsequently, the character's capabilities and attributes are appointed and modified according to the preferable designations and game mechanics. The character's capabilities and attributes of the character increased their values by including the power of the character, alertness of the character, and considering the character's special features. With the help of the pseudocode and flowcharts, the creation process of the character can be definitely detailed and generic.

# Discussion

This project is dedicated to creating a Pokemon character, named Pikachu. This Python creation has used Python IDE IDLE as a Python display tool. Turtle refers to a built-in library called Turtle Graphics (Hopkins III *et al.* 2022). Turtle is a very useful and very efficient tool to create art and animation. By controlling the virtual arrow in the screen this allows the user to create anything artistic that they want to create to implement Python as the instruction setter. This Pikachu character is based on the Pokemon cartoon franchise. The body description of this Pokemon is very simple and iconic according to the views of this franchise. Pikachu is a very famous character known for his pointy ear, thunderbolt tail, and bright yellow color. In this project, the main agenda is to build the body parts of Pikachu separately as functions and then call them in chronological order(Solin *et al.*2023). This approach has a very definite passcode and flowchart. Pseudocode and flowchart are the plain language explanation of an algorithm. Pseudocode is not a programming language, but Pseudocode has the ability to contour the steps of a process in a detailed but compact and condensed way.

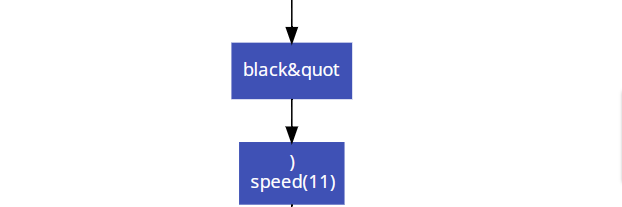
## FlowChart

In this case study, the Pseudocode to create an anime character (Pikachu) is:

****

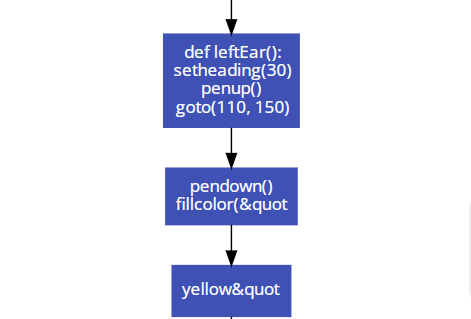
**Figure 1: Flowchart of the code**

(Source: Salf-Created Drawio)

****

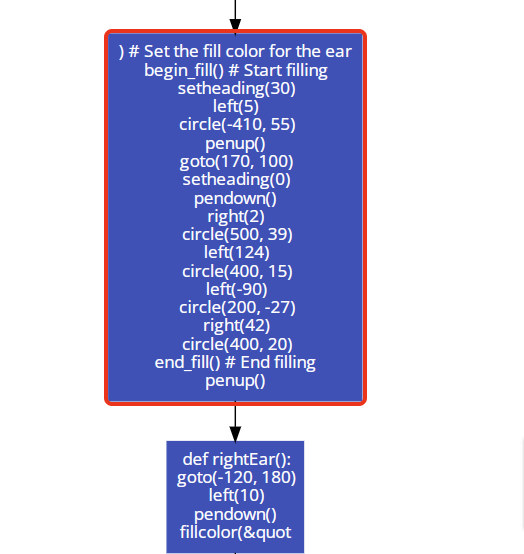
**Figure 2: Flowchart of the code**

(Source: Salf-Created Drawio)

****

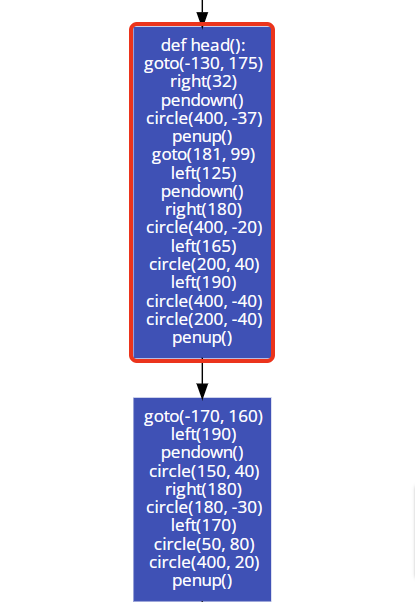
**Figure 3: Flowchart of the code**

(Source: Salf-Created Drawio)

****

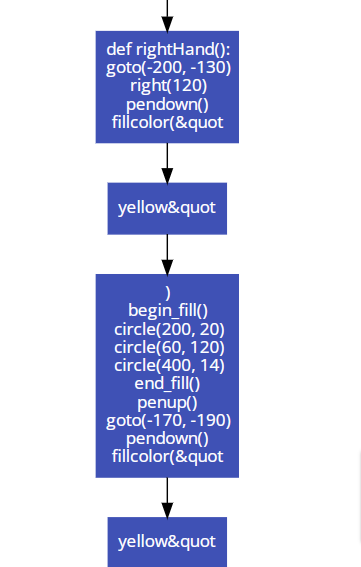
**Figure 4: Flowchart of the code**

(Source: Salf-Created Drawio)

****

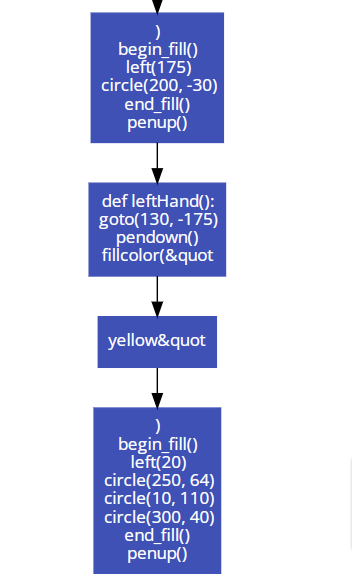
**Figure 5: Flowchart of the code**

(Source: Salf-Created Drawio)

****

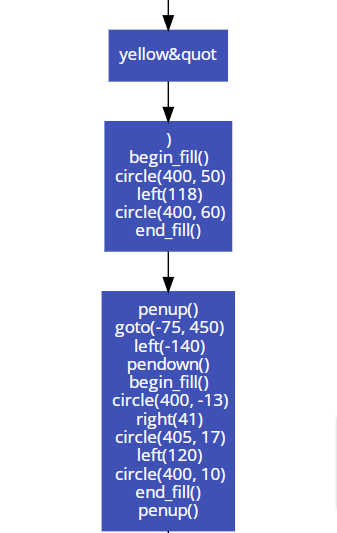
**Figure 6: Flowchart of the code**

(Source: Salf-Created Drawio)

****

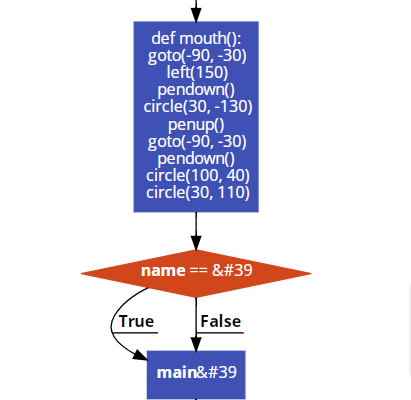
**Figure 7: Flowchart of the code**

(Source: Salf-Created Drawio)

****

**Figure 8: Flowchart of the code**

(Source: Salf-Created Drawio)

****

**Figure 9: Flowchart of the code**

(Source: Salf-Created Drawio)

****

**Figure 10: Flowchart of the code**

(Source: Salf-Created Drawio)

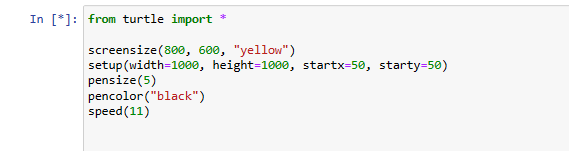
## Pseudocode

This code has several functional architecture.Functions. Create the separate function of the figure and then by calling then create the whole scenario.

**Figure 11: Flow of the program**

(Source: Salf-Created)

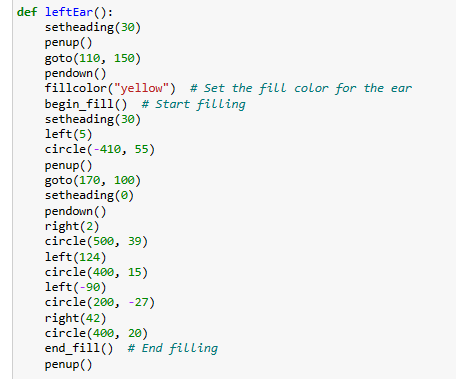
## Code



**Figure 12: Importing the essential libraries to perform the program**

(Source: Salf-Created Python code)

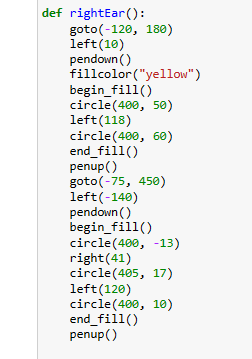
Importing the turtle library in the Python interpreter platform to use the predefined functions of the turtle(Ezeamuzie, 2023). The functions of turtle help to move the controller by using Python commands like foreword( ), penup(), pendown (), circle(), etc.



**Figure 13: Developing the left ear function**

(Source: Salf-Created Python code)

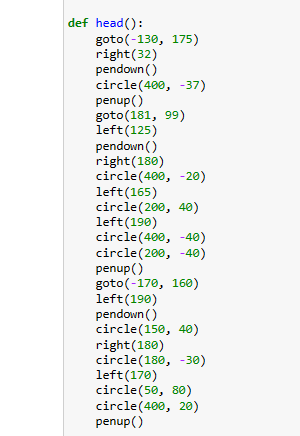
In this case study, this particular segment of code def leftEar(): identifies a function called by main(). The main feature of the function is it takes no arguments and returns nothing. The function starts with setting the fill color to yellow(#F8FF95) (Georgieva-Trifonova, 2023). The function used to call begin\_fill() method. The segmentations of lines are drawn by applying the right(), left(), and forward() methods (Bachmann 2023). The function turns left again and again to draw another line segment forward. When all the lines are drawn then the function calls the end\_fill() method.



**Figure 14: Developing the right ear function**

(Source: Salf-Created Python code)

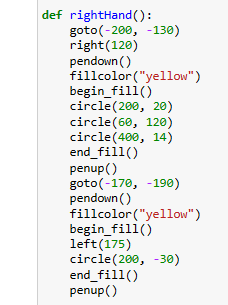
This particular segment of code def rightEar(): identifies a function called by main(). The main feature of the function is it takes no arguments and returns nothing. With each part of the code, the main code will draw the main part of the art.



**Figure 15: Developing the head of the character function**

(Source: Salf-Created Python code)

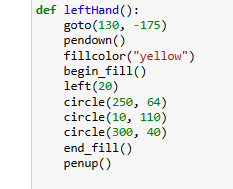
This particular segment of code def head(): identifies a function called by main(). The main feature of the function is it takes no arguments and returns nothing. With each part of the code, the main code will draw the main part of the art.



**Figure 16: Developing the right-hand function**

(Source: Salf-Created Python code)

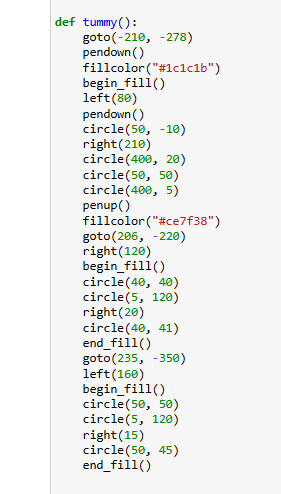
This particular segment of code def rightHand(): identifies a function called by main(). The main feature of the function is it takes no arguments and returns nothing. With each part of the code, the main code will draw the main part of the art.



**Figure 17: Developing the left-hand function**

(Source: Salf-Created Python code)

This particular segment of code def leftHand(): identifies a function called by main(). The main feature of the function is it takes no arguments and returns nothing. With each part of the code, the main code will draw the main part of the art.

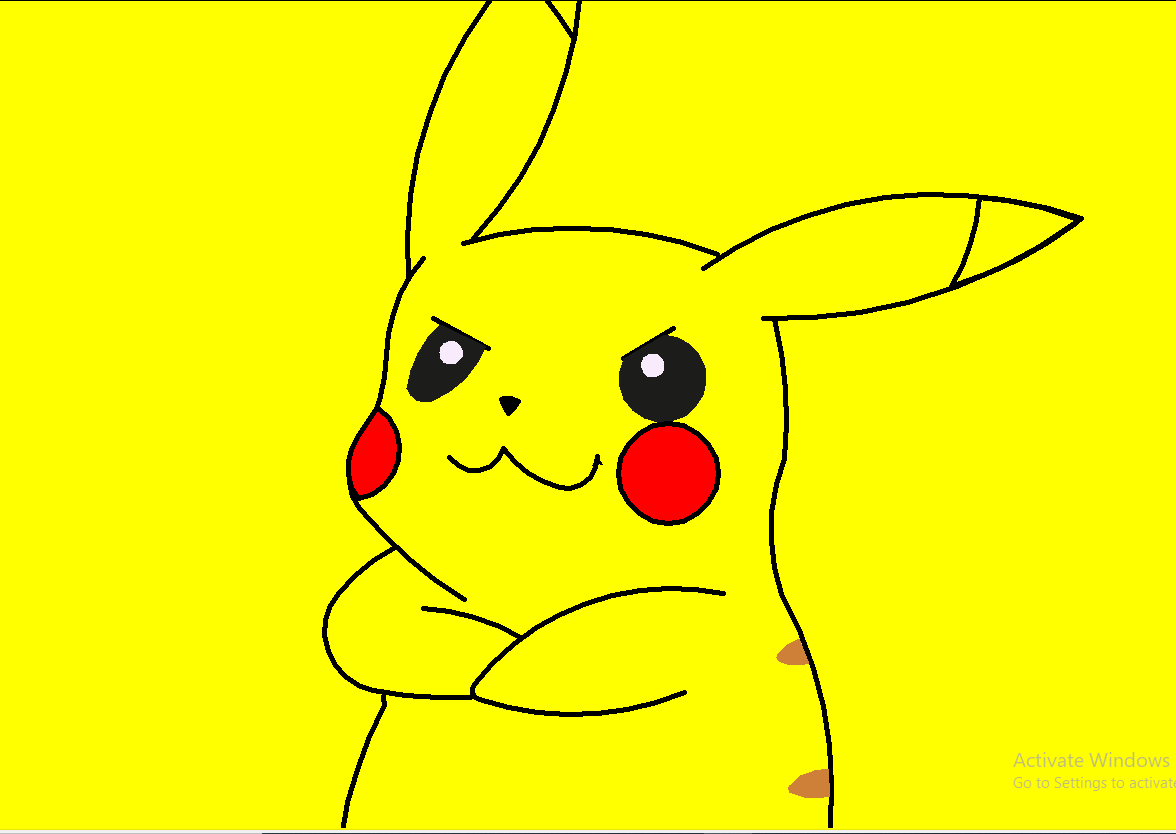


**Figure 18: Developing the tummy function**

(Source: Salf-Created Python code)

This particular segment of code def tummy(): identifies a function called by main(). The main feature of the function is it takes no arguments and returns nothing(Bearss *et al.*2022). With each part of the code, the main code will draw the main part of the art.

## Outcome



**Figure 29: Final outcome of the code**

(Source: Salf-Created Python code)

This is the final outcome of this code.

# Conclusion

This visualization of Pokemon is under the influence of the python turtle. This program is split into several parts in terms of the Python function. This code consists of the pseudocode and the flowchart. The output is the visualization of the several body part collections.

# Reference List

Bearss, E.M., Cantrell, W.A., Hall, C.W., Pinckard, J.E. and Petty, M.D., 2022, April. Wolf sheep predation: reimplementing a predator-prey ecosystem model as an instructional exercise in agent-based modeling. In *Proceedings of the 2022 ACM Southeast Conference* (pp. 38-43).

Georgieva-Trifonova, T., 2023. Establishing Cross-Curricular Connections Through Python Programming in Computer Modeling Education. *TEM Journal*, *12*(3), p.1786.

Bachmann, C., 2023. *WebTigerJython 3: A Web-Based Python IDE Supporting Educational Robotics* (Master's thesis, ETH Zurich).

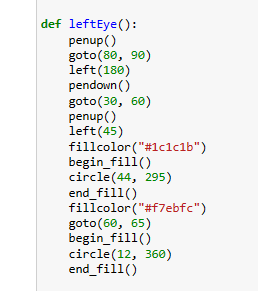
Solin, P. and Freyer, A., 2023. Self-Paced, Instructor-Assisted Approach to Teaching Python Programming. *Mathematics in Computer Science*, *17*(2), pp.1-16.

Ezeamuzie, N.O., 2023. Project-first approach to programming in K–12: Tracking the development of novice programmers in technology-deprived environments. *Education and Information Technologies*, *28*(1), pp.407-437.

Hopkins III, J.B., Frederick, C.A., Yorks, D., Pollock, E. and Chatfield, M.W., 2022. Forensic application of stable isotopes to distinguish between wild and captive turtles. *Biology*, *11*(12), p.1728.

# Appendices

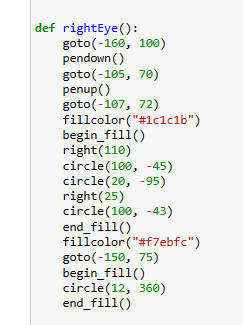
**Appendix 1: Developing the left eye function**



**Developing the left eye function**

(Source: Salf-Created Python code)

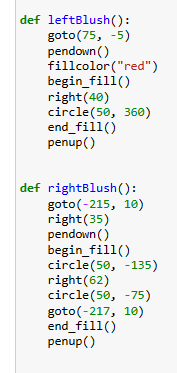
**Appendix 2: Developing the right eye function**



**Developing the right eye function**

(Source: Salf-Created Python code)

**Appendix 3: Developing the extra functions for the character**



**Developing the extra functions for the character**

(Source: Salf-Created Python code)

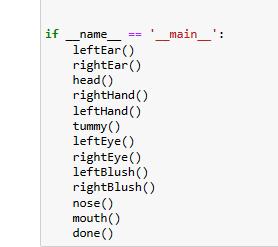
**Appendix 4: Developing the extra functions for the character**



**Developing the extra functions for the character**

(Source: Salf-Created Python code)

**Appendix 6: Main function**



**Main function**

(Source: Salf-Created Python code)