**Python Program to** **Find the Factorial of a Number**

**Purpose of program:**

This program can help people to:

1. Simplify the process of calculate the factorial of a number, get the factorial quickly with program’s assist.

**Target users:**

The students can use this program to reduce the work in calculation*.* Moreover, this program can help teachers in teaching. They have this program, it will double their working efficiency.

**How will they use the program?**

When user run this program, it will ask user to input a number(integer) to find factorial. When user input wrong type of variable such as "@" and "abc", they’ll get an error message: "Please Enter Appropriate Values". After typing the number and clicked "Factorial" button, computer will output the result and a message to users "Did you get the right answer? ". Hence, if users click the "Yes" button, computer will output a message randomly from "You did great!", "Keep your precision!", "Wonderful!". In opposite, if users click the “No” button, computer will output a message to users "Sorry, try again.", "You should do it again.", "Wrong answer."

**Which is the programming language?**

This program using python 3.5.0 to create. IDLE 3.5 to edit.

**Variables Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Scope** | **Description** |
| num | Integer | factorial | Get number from users’ input |
| factorial\_num | Integer | factorial | Set "factorial\_num" to 1 as default |
| i | Integer | factorial | Use "i" to loop and multiply |
| c\_word | Integer | correct | Random number chose from the list 0, 1, 2 |
| w\_word | Integer | wrong | Random number chose from the list 0, 1, 2 |

**Class Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | **Type** | **Scope** | **Description** |
| Application (Frame) | class | global | Application Class for the complete program |

**List Table**

|  |  |  |
| --- | --- | --- |
| **Type** | **Scope** | **Description** |
| c\_wordlist | correct | Storing three list values in c\_wordlist and displays when users get right result |
| w\_wordlist | wrong | Storing three list values in w\_wordlist and displays when users get wrong result |

**Testing Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** | **Test Result** |
| num=3 | Factorial: 6 | 6 | Passed |
| num=5 | Factorial: 120 | 120 | Passed |
| num=7 | Factorial: 5040 | 5040 | Passed |

**Subroutine Table**

|  |  |
| --- | --- |
| **Subroutine Name** | **Description** |
| factorial | Creates a find factorial number subroutine |
| correct | Creates a subroutine that displays comments when users get the right answer |
| wrong | Creates a subroutine that displays comments when users get the wrong answer |

**Program Sketch**

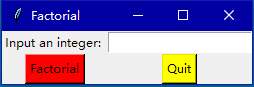
**Factorial**

**Solve**

**Quit**

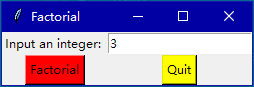
**Input an integer:**

**Program Code Screenshot – Explanation:**

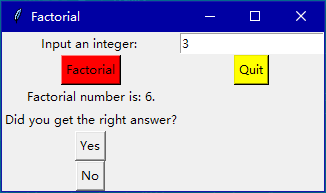


This is main GUI for the program. The progress of the program goes like the following.

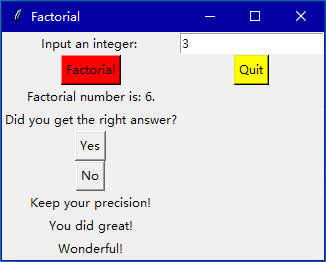
1. First, user input an integer.



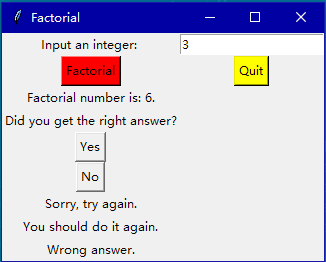
1. Click “Find”, then the program will calculate the result of the factorial number and display down below.



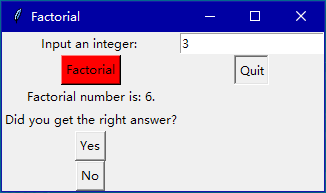
1. Programs among will give you a question "Did you get the right answer?" If you click button "Yes", it will show the sentences randomly like below ("You did great!", "Keep your precision!", "Wonderful!").



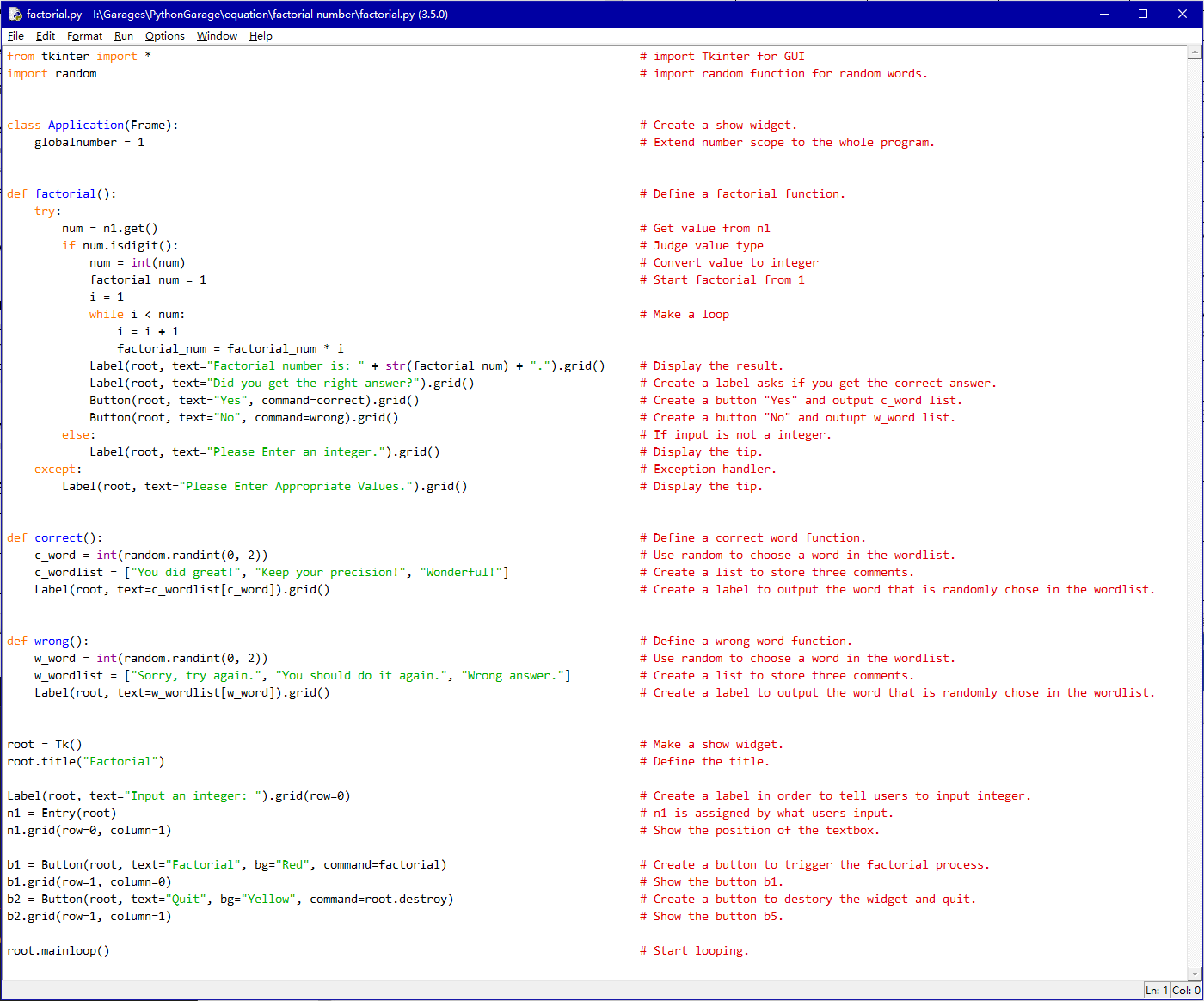
If users click button "No", it will show the sentences randomly like below ("Sorry, try again.", "You should do it again.", "Wrong answer.").



1. When users want to finish the program, they can click the button “Quit” to stop it



**Program Code – Screenshot:**



**Program Code:**

from tkinter import \* # import Tkinter for GUI

import random # import random function for random words.

class Application(Frame): # Create a show widget.

globalnumber = 1 # Extend number scope to the whole program.

def factorial(): # Define a factorial function.

try:

num = n1.get() # Get value from n1

if num.isdigit(): # Judge value type

num = int(num) # Convert value to integer

factorial\_num = 1 # Start factorial from 1

i = 1

while i < num: # Make a loop

i = i + 1

factorial\_num = factorial\_num \* i

Label(root, text="Factorial number is: " + str(factorial\_num) + ".").grid() # Display the result.

Label(root, text="Did you get the right answer?").grid() # Create a label asks if you get the correct answer.

Button(root, text="Yes", command=correct).grid() # Create a button "Yes" and output c\_word list.

Button(root, text="No", command=wrong).grid() # Create a button "No" and outupt w\_word list.

else: # If input is not a integer.

Label(root, text="Please Enter an integer.").grid() # Display the tip.

except: # Exception handler.

Label(root, text="Please Enter Appropriate Values.").grid() # Display the tip.

def correct(): # Define a correct word function.

c\_word = int(random.randint(0, 2)) # Use random to choose a word in the wordlist.

c\_wordlist = ["You did great!", "Keep your precision!", "Wonderful!"] # Create a list to store three comments.

Label(root, text=c\_wordlist[c\_word]).grid() # Create a label to output the word that is randomly chose in the wordlist.

def wrong(): # Define a wrong word function.

w\_word = int(random.randint(0, 2)) # Use random to choose a word in the wordlist.

w\_wordlist = ["Sorry, try again.", "You should do it again.", "Wrong answer."] # Create a list to store three comments.

Label(root, text=w\_wordlist[w\_word]).grid() # Create a label to output the word that is randomly chose in the wordlist.

root = Tk() # Make a show widget.

root.title("Factorial") # Define the title.

Label(root, text="Input an integer: ").grid(row=0) # Create a label in order to tell users to input integer.

n1 = Entry(root) # n1 is assigned by what users input.

n1.grid(row=0, column=1) # Show the position of the textbox.

b1 = Button(root, text="Factorial", bg="Red", command=factorial) # Create a button to trigger the factorial process.

b1.grid(row=1, column=0) # Show the button b1.

b2 = Button(root, text="Quit", bg="Yellow", command=root.destroy) # Create a button to destory the widget and quit.

b2.grid(row=1, column=1) # Show the button b5.

root.mainloop() # Start looping.

**Flow Chart:**

Start

Run the “Main” subroutine

Import the GUI

Create the Application class

Make a labels called enter integer

Wrong

Correct

Find the Factorial of a Number

Name GUI window as “Root”

Main

Start loop until something happens

Find the Factorial of a Number

Get num from input as integer number

Create a label display:” Did you get the right answer?”

No

Create a button “No”

Create a button “Yes”

Return

While i < num

i = i + 1

factorial\_num = factorial\_num \* i

Yes

Create a label to display result

Quit

End

Create a wordlist

Import random function

Wrong

Correct

Import random function

Create a wordlist

Create a label display c\_wordlist[c\_word]

Create a label display w\_wordlist[w\_word]

Return

Return