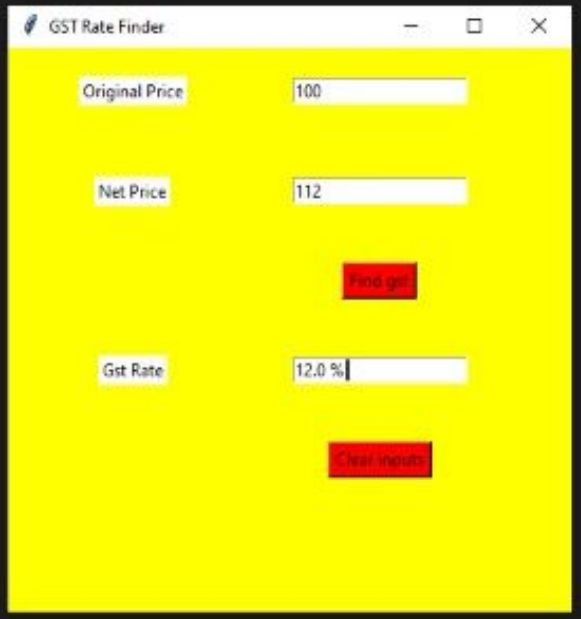
#Q1 SID-21107052 Untitled-1 | #Q2 SID-21107052 Untitled-2 | #Q3 SID-21107052 Untitled-3 | def partition(l, r, nums): Untitled-4 | 21107052_assignment 7.ipynb | Untitled-1.ipynb

```python
#Q1         SID-21107052
from tkinter import *

def findGst() :

    org_cost= int(org_priceField.get())
    N_price = int(net_priceField.get())
    gst_rate = ((N_price - org_cost) * 100) / org_cost;
    gst_rateField.insert(10, str(gst_rate) + " % ")

def clearAll():
    org_priceField.delete(0, END)
    net_priceField.delete(0, END)
    gst_rateField.delete(0, END)

if __name__ == "__main__" :
    gui = Tk()
    gui.configure(background = "yellow")
    gui.title("GST Rate Finder")
    gui.geometry("400x400")
    org_price = Label(gui, text = "Original Price",
                      bg = "white")
    net_price = Label(gui, text = "Net Price",
                      bg = "white")
    find = Button(gui, text = "Find gst", fg = "Black",
                  bg = "Red",
                  command = findGst)
    gst_rate = Label(gui, text = "Gst Rate", bg = "white")
    clear = Button(gui, text = "Clear inputs", fg = "Black",
                  bg = "Red",
                  command = clearAll)
    org_price.grid(row = 1, column = 1,padx = 50,pady = 20)
    net_price.grid(row = 2, column = 1, padx = 50, pady = 30)
    find.grid(row = 3, column = 2,padx = 50,pady = 10)
    gst_rate.grid(row = 4, column = 1,padx = 50, pady = 30)
    clear.grid(row = 5, column = 2, padx = 50, pady = 10)
    org_priceField = Entry(gui)
    net_priceField = Entry(gui)
    gst_rateField = Entry(gui)
    org_priceField.grid(row = 1, column = 2 ,padx = 10,pady = 10)
    net_priceField.grid(row = 2, column = 2, padx = 10,pady = 10)
    gst_rateField.grid(row = 4, column = 2, padx = 10,pady = 10)

    gui.mainloop()
```



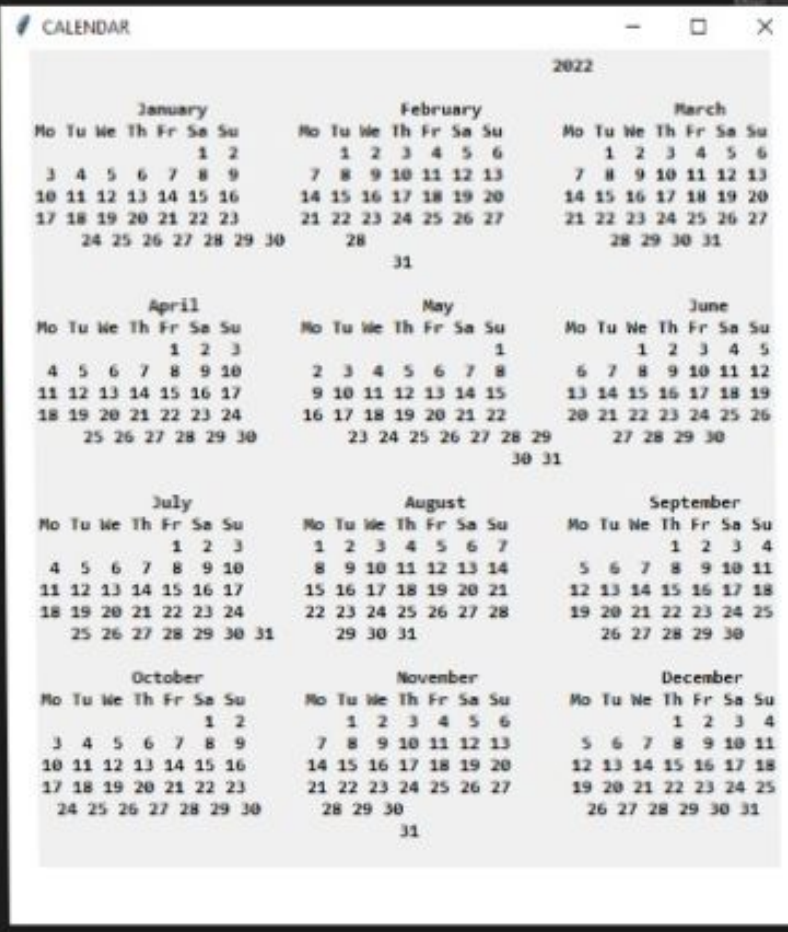
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Code

```
[Done] exited with code=0 in 0.058 seconds

[Running] python -u "c:\Users\gurki\Downloads\assignments\21107052_assignment 7.ipynb"
```

#Q1 SID-21107052 Untitled-1 ● #Q2 SID-21107052 Untitled-2 ● #Q3 SID-21107052 Untitled-3 ● def partition(l, r, nums): Untitled-4 ● 21107052_assignment 7.ipynb ● Untitled-1.ipynb ●

```python
#Q2              SID-21107052
from tkinter import *

import calendar

def showCal() :
    new_gui = Tk()
    new_gui.config(background = "white")
    new_gui.title("CALENDAR")
    new_gui.geometry("550x600")
    fetch_year = int(year_field.get())
    cal_content = calendar.calendar(fetch_year)
    cal_year = Label(new_gui, text = cal_content, font = "Consolas 10 bold")
    cal_year.grid(row = 5, column = 1, padx = 20)
    new_gui.mainloop()

if __name__ == "__main__" :
    gui = Tk()
    gui.config(background = "light green")
    gui.title("CALENDAR")
    gui.geometry("250x250")
    cal = Label(gui, text = "CALENDAR", bg = "yellow",
                        font = ("times", 18, 'bold'))
    year = Label(gui, text = "Enter Year", bg="light green",font=('bold'))
    year_field = Entry(gui)
    Show = Button(gui, text = "Show Calendar", fg = "Black",
                        bg = "Red", command = showCal)
    Exit = Button(gui, text = "Exit", fg = "Black", bg = "Red", command = exit)
    cal.grid(row = 1, column = 1,padx=40,pady=10)
    year.grid(row = 2, column = 1,padx=40,pady=5)
    year_field.grid(row = 3, column = 1,padx=40,pady=5)
    Show.grid(row = 4, column = 1,padx=40,pady=10)
    Exit.grid(row = 6, column = 1,padx=40)

    gui.mainloop()
```



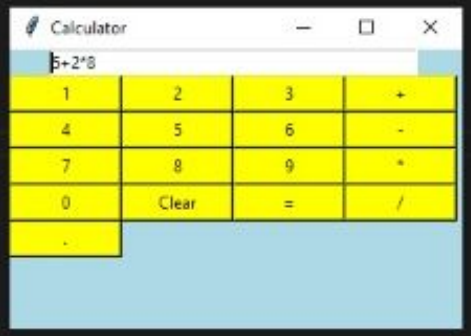
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Code

```
[Running] python -u "c:\Users\gurki\Downloads\assignments\tempCodeRunnerFile.python"
```

#Q1 SID-21107052 Untitled-1 ● #Q2 SID-21107052 Untitled-2 ● #Q3 SID-21107052 Untitled-3 ● def partition(l, r, nums): Untitled-4 ● 21107052_assignment 7.ipynb ● Untitled-1.ipynb ●

```python
#Q3            SID-21107052
from tkinter import *
expression = ""
def press(num):
    global expression
    expression = expression + str(num)
    equation.set(expression)

def equalpress():
    try:
        global expression
        total = str(eval(expression))
        equation.set(total)
        expression = ""

    except:
        equation.set(" error ")
        expression = ""

def clear():
    global expression
    expression = ""
    equation.set("")

if __name__ == "__main__":
    gui = Tk()
    gui.configure(background="light blue")
    gui.title("Calculator")
    gui.geometry("300x200")
    equation = StringVar()
    expression_field = Entry(gui, textvariable=equation)
    expression_field.grid(columnspan=4, ipadx=70)
    button1 = Button(gui, text=' 1 ', fg='black', bg='yellow',
                     command=lambda: press(1), height=1, width=10)
    button1.grid(row=2, column=0)
    button2 = Button(gui, text=' 2 ', fg='black', bg='yellow',
                     command=lambda: press(2), height=1, width=10)
    button2.grid(row=2, column=1)
    button3 = Button(gui, text=' 3 ', fg='black', bg='yellow',
                     command=lambda: press(3), height=1, width=10)
    button3.grid(row=2, column=2)
    button4 = Button(gui, text=' 4 ', fg='black', bg='yellow',
                     command=lambda: press(4), height=1, width=10)
    button4.grid(row=3, column=0)
    button5 = Button(gui, text=' 5 ', fg='black', bg='yellow',
                     command=lambda: press(5), height=1, width=10)
```



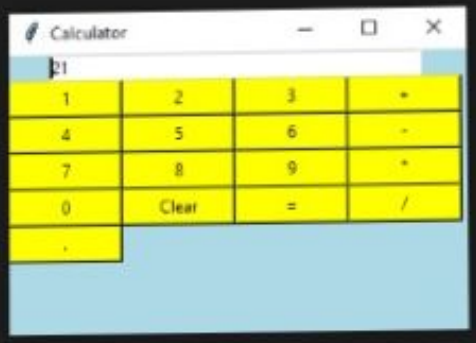
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Code

```
[Running] python -u "c:\Users\gurki\Downloads\assignments\tempCodeRunnerFile.python"
```

#Q1 SID-21107052 Untitled-1 ● #Q2 SID-21107052 Untitled-2 ● #Q3 SID-21107052 Untitled-3 ● def partition(l, num): Untitled-4 ●

```
    button3.grid(row=2, column=2)
    button4 = Button(gui, text=' 4 ', fg='black', bg='yellow',
                     command=lambda: press(4), height=1, width=10)
    button4.grid(row=3, column=0)
    button5 = Button(gui, text=' 5 ', fg='black', bg='yellow',
                     command=lambda: press(5), height=1, width=10)
    button5.grid(row=3, column=1)
    button6 = Button(gui, text=' 6 ', fg='black', bg='yellow',
                     command=lambda: press(6), height=1, width=10)
    button6.grid(row=3, column=2)
    button7 = Button(gui, text=' 7 ', fg='black', bg='yellow',
                     command=lambda: press(7), height=1, width=10)
    button7.grid(row=4, column=0)
    button8 = Button(gui, text=' 8 ', fg='black', bg='yellow',
                     command=lambda: press(8), height=1, width=10)
    button8.grid(row=4, column=1)
    button9 = Button(gui, text=' 9 ', fg='black', bg='yellow',
                     command=lambda: press(9), height=1, width=10)
    button9.grid(row=4, column=2)
    button0 = Button(gui, text=' 0 ', fg='black', bg='yellow',
                     command=lambda: press(0), height=1, width=10)
    button0.grid(row=5, column=0)
    plus = Button(gui, text=' + ', fg='black', bg='yellow',
                  command=lambda: press("+"), height=1, width=10)
    plus.grid(row=2, column=3)
    minus = Button(gui, text=' - ', fg='black', bg='yellow',
                   command=lambda: press("-"), height=1, width=10)
    minus.grid(row=3, column=3)
    multiply = Button(gui, text=' * ', fg='black', bg='yellow',
                      command=lambda: press("*"), height=1, width=10)
    multiply.grid(row=4, column=3)
    divide = Button(gui, text=' / ', fg='black', bg='yellow',
                    command=lambda: press("/"), height=1, width=10)
    divide.grid(row=5, column=3)
    equal = Button(gui, text=' = ', fg='black', bg='yellow',
                   command=equalpress, height=1, width=10)
    equal.grid(row=5, column=2)
    clear = Button(gui, text='Clear', fg='black', bg='yellow',
                   command=clear, height=1, width=10)
    clear.grid(row=5, column='1')
    Decimal= Button(gui, text='.', fg='black', bg='yellow',
                    command=lambda: press('.'), height=1, width=10)
    Decimal.grid(row=6, column=0)

    gui.mainloop()
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Code

[Running] python -u "c:\Users\gurki\Downloads\assignments\tempCodeRunnerFile.python"

+ Code + Markdown | Run All | Clear Outputs of All Cells | Restart | Interrupt | Variables | Outline ···

```python
#Q4            SID-21107052
def partition(l, r, nums):

    pivot, ptr = nums[r], l
    for i in range(l, r):
        if nums[i] <= pivot:

            nums[i], nums[ptr] = nums[ptr], nums[i]
            ptr += 1

    nums[ptr], nums[r] = nums[r], nums[ptr]
    return ptr


def quicksort(l, r, nums):
    if len(nums) == 1:
        return nums
    if l < r:
        pi = partition(l, r, nums)
        quicksort(l, pi-1, nums)
        quicksort(pi+1, r, nums)
    return nums


example = [4, 5, 1, 2, 3]
result = [1, 2, 3, 4, 5]
print(quicksort(0, len(example)-1, example))

example = [2, 5, 6, 1, 4, 6, 2, 4, 7, 8]
result = [1, 2, 2, 4, 4, 5, 6, 6, 7, 8]
print(quicksort(0, len(example)-1, example))
```

[1] ✓ 0.4s

```
[1, 2, 3, 4, 5]
[1, 2, 2, 4, 4, 5, 6, 6, 7, 8]
```

```python
#Q5              SID-21107052
def heapify(nums, heap_size, root_index):

    largest = root_index
    left_child = (2 * root_index) + 1
    right_child = (2 * root_index) + 2
    if left_child < heap_size and nums[left_child] > nums[largest]:
        largest = left_child
    if right_child < heap_size and nums[right_child] > nums[largest]:
        largest = right_child
    if largest != root_index:
        nums[root_index], nums[largest] = nums[largest], nums[root_index]
        heapify(nums, heap_size, largest)

def heap_sort(nums):
    n = len(nums)
    for i in range(n, -1, -1):
        heapify(nums, n, i)
    for i in range(n - 1, 0, -1):
        nums[i], nums[0] = nums[0], nums[i]
        heapify(nums, i, 0)
random_list_of_nums = [35, 12, 43, 8, 51]
heap_sort(random_list_of_nums)
print(random_list_of_nums)
```

[2] ✓ 0.5s

```
[8, 12, 35, 43, 51]
```

+ Code + Markdown | Run All Clear Outputs of All Cells Restart

```python
#Q6                  SID-21107052
def Remove(duplicate):
    final_list = []
    for num in duplicate:
        if num not in final_list:
            final_list.append(num)
    return final_list

duplicate = [2, 4, 10, 20, 5, 2, 20, 4]
print(Remove(duplicate))
```

[4] ✓ 0.4s

```
[2, 4, 10, 20, 5]
```