Avoid global variables

- You should avoid creating one global variable, than then using it in a bunch of other places
  - What happens when you change the name of the global?

- Instead, primarily use local variables
  - When you need those local values in other functions, use parameter variables and arguments

Constants

- A variable name with all CAPITAL LETTERS indicates that the value should be treated as a **constant**
- A **constant** is a value that can be **used** but not **changed**

Instead of

```
last = input('Enter last name: ')
if not last.isalpha() or len(last) >= 30:
    print('Invalid last name.')
    exit(0)

print('Last name OK')
```

You can write

```
MAX_NAME_LENGTH = 30

last = input('Enter last name: ')
if not last.isalpha() or len(last) >= MAX_NAME_LENGTH:
    print('Invalid last name.')
    exit(0)

print('Last name OK')
```

```
gui = graphics(500, 500, 'Example')
color_string = 'blue'
i = 0
while True:
    if i % 60 == 0:
        blue = random.randint(0, 255)
        color_string = gui.get_color_string(0, 0, blue)
    gui.clear()
    gui.rectangle(gui.mouse_x - 50, gui.mouse_y - 50, \
                  100, 100, color_string)
    gui.update_frame(60)
    i += 1
```

What could be a constant?

```
gui = graphics(500, 500, 'Example')
color_string = 'blue'
i = 0
while True:
    if i % 60 == 0:
        blue = random.randint(0, 255)
        color_string = gui.get_color_string(0, 0, blue)
    gui.clear()
    gui.rectangle(gui.mouse_x - 50, gui.mouse_y - 50, \
                  100, 100, color_string)
    gui.update_frame(60)
    i += 1
```

What could be a constant?

```
CANVAS_WIDTH = 500
CANVAS_TITLE = 'Example'
FRAME_RATE = 60
SQUARE_WIDTH = 100

gui = graphics(CANVAS_WIDTH, CANVAS_WIDTH, CANVAS_TITLE)
color_string = 'blue'
i = 0
while True:
    if i % FRAME_RATE == 0:
        blue = random.randint(0, 255)
        color_string = gui.get_color_string(0, 0, blue)
    gui.clear()
    gui.rectangle(gui.mouse_x - SQUARE_WIDTH//2, gui.mouse_y - SQUARE_WIDTH//2, \
                  SQUARE_WIDTH, SQUARE_WIDTH, color_string)
    gui.update_frame(FRAME_RATE)
    i += 1
```

Magic Values

- When a number is used in a program and it is not immediately clear WHY it is what it is, these are sometimes referred to as a **magic number**
- Instead of using **magic numbers (or values)**, use constants
- That way
  - If they need to be changed, can be changed in one place
  - Self-documentation

Write the code

- Write a full program that. . .
- Opens a file names **amounts.txt**
- Writes a file named **dollars.txt**, with only the dollar amounts from each line of amounts.txt

amounts.txt:	dollars.txt:
$100.00	100
$1.57	1
$0.76	0
$.09	
$5712.21	5712
$450.10	450
$10.50	10

Write the code

```
amounts = open('amounts.txt', 'r')
dollars = open('dollars.txt', 'w')
for line in amounts:
    just_dollar = line[1:].split('.')[0]
    dollars.write(just_dollar + '\n')
dollars.close()
```

amounts.txt:	dollars.txt:
$100.00	100
$1.57	1
$0.76	0
$.09	
$5712.21	5712
$450.10	450

Swap the numbers

```
names = ['joe', 'sam', 'alex', 'rome', 'alice']
# Swap 'sam' with 'rome'
print(names)
```

Swap the numbers

```
names = ['joe', 'sam', 'alex', 'rome', 'alice']
temp = names[1]
names[1] = names[3]
names[3] = temp
print(names)
```

Swap the numbers

```
names = ['joe', 'sam', 'alex', 'rome', 'alice']
names[1], names[3] = names[3], names[1]
print(names)
```

Reverse the values in the list

```
names = ['a', 'b', 'c', 'd', 'e', 'f']
# ???
print(names)
# should print ['f', 'e', 'd', 'c', 'b', 'a']
```

Reverse the values in the list

```
names = ['a', 'b', 'c', 'd', 'e', 'f']
for i in range(0, int(len(names)/2)):
    i2 = len(names) - i - 1
    temp = names[i]
    names[i] = names[i2]
    names[i2] = temp
print(names)
```

Reverse the values in the list

```
names = ['a', 'b', 'c', 'd', 'e', 'f']
for i in range(0, int(len(names)/2)):
    i2 = len(names) - i - 1
    names[i], names[i2] = names[i2], names[i]
print(names)
```