

```
# welcome to the RC Python workshop.
```

```
# to get started run your first code.
```

```
print("Hello, World")
```

```
# use "#%" to divide your code into cells
```

```
# use "#" to make comments to your code
```

```
#####
```

```
# 1. Variables (Named storage for data) and data types
```

```
'''
```

```
data types:
```

```
int: Integer -> eg. 10
```

```
Float: eg. 2.5
```

```
complex: complex numbers -> eg. 3 + 4j
```

```
str: string values -> eg. "Hello, World!"
```

```
bool: booleans -> eg. True/False
```

```
list: lists -> eg. [1, 2, 3, 4, "stable"]
```

```
tuple: tuples -> eg. (5, 6, 7, "stable")
```

```
range: ranges -> eg. range(0, 10)
```

```
dict: dictionaries -> eg. {"name": "Caivil", "age": 25}
```

```
set: sets -> eg. {1, 2, 3, 4}
```

```
'''
```

```
a = 10
```

```
b = 15
```

```
print(a,b)
```

```
print("a")
```

```
print("a",a,b)
```

```
call = 'good day sir'
```

```
print(call)
```

```
print(Call) # name 'Call' is not defined? why?
```

```
a1=20 #valid variable can start with a letter, followed by letters, numbers, or underscores.
```

```
1a=20 # SyntaxError. Invalid because variable names cannot start with a digit.
```

```
# Syntax is the grammar of a programming language.
```

```
# To check for the kind of data type use "is.instance(variable, data type suspected)
```

```
greeting = "hello everyone"
```

```
type(greeting)# str
```

```
##%#####
```

```
#2. F String
```

```
name = "Cyril"
```

```
surname = "Ramaphosa"
```

```
print(name + " " + surname) # output 'Cyril Ramaphosa'
```

```
print(name,surname)
```

```
c = 20
```

```
name + c # TypeError (operation on a value of the wrong data type)
```

```
print(f"The name of the president of South Africa is {name} and his second name is {surname}")  
# takes str
```

```
print(f'i have {c} amount of apples") #takes int
```

```
print('my lucky number is:', c)
```

```
print(c,'my lucky number!')
```

```
##%#####  
####
```

#3. Comparison operator (==, !=, >, <, >=, <=) #output is boolean (True/False)

```
a = 8
```

```
b = 10
```

```
a == b # is a the SAME as b -> output: False
```

```
a != b # is a NOT SAME as b -> output: True
```

```
a > b # a GREATER THAN b -> output: False
```

```
a < b # a LESS THAN b -> output: true
```

```
a >= b # a GREATER THAN OR EQUAL TO b -> output: False
```

```
a <= b # a LESS THAN OR EQUAL TO b -> output: True
```

```
##%#####  
####
```

4. logical operator (and or not)

```
a=5
```

```
b= 10
```

```
a<8 and a>6 # If one premise is false, the whole statement is false.
```

```
a<8 or a>10 # If one premise is true, the statement is true.
```

```
a!=b or a>b and a<=b # and has higher precedence than or
```

```
not a!=b or a>b and a<=b
```

```
###  
#
```

5. List

```
X = [10,"Apple", "Orange", 30]
```

```
type(X)# what data type is it
```

```
'''
```

```
x = ["Paul Mashatile","John Steenhuisen","Caivil Ndobela"]
```

```
for student in x:
```

```
    print(f"Dear Dr.{student} we hope this email finds you well ...")
```

```
'''
```

```
X.append(1) # adds 11 to the back of the list
```

```
print(X)
```

```
X.remove(11)# removes specified item
```

```
X[2]='Grapes' # replaces items, remember python counts from 0
```

```
X.insert(2, 'black') # list.insert(position, new item)
```

```
###  
####
```

6. Dictionary

```
Siblings={'Caivil':27, 'Pepe': 24, 'Julie': 21}
```

```
type(Siblings)
```

```
print(Siblings['Caivil'])
```

```
print(Siblings)
```

```
##%#####
```

```
# 7. Input function
```

```
a=input('Enter the value:')
```

```
print(a)
```

```
b = int(input('Enter first the value:'))
```

```
c = int(input('Enter second the value:'))
```

```
print(b+c)
```

```
##%#####  
####
```

```
# 8. IF statement
```

```
'''
```

```
if(conditional statement):
```

```
(statement to be executed)
```

```
'''
```

```
# special math % -> only give the remainder after dividing eg. 10%3 = 1
```

```
z=100
```

```
if z%10 == 0:
```

```
    print("10 is factor of z")
```

```
 #(if, else)
```

```
a = 5
```

```
b = 2
```

```
if a + b == 7:
    print('The correct answer is 7')
else:
    print('you are bad at math')
```

```
 #(if,else,elif)
fruit_colour = "red"
```

```
if fruit_colour == "yellow":
    print ("its a banana!")
elif fruit_colour == "green":
    print("its an apple!")
elif fruit_colour == "red":
    print("its a strewberry!")
elif fruit_colour == "blue":
    print("its a blueberry!")
else:
    print("i give up, just tell me?")
```

```
# Making a basic calculator!
A=int(input("Enter first val ="))
B=int(input("Enter second val ="))
opr=input("Enter the operator:")
if opr=="+":
    print(A+B)
elif opr=="-":
    print(A-B)
elif opr=="*":
```

```
    print(A*B)
elif opr=="/":
    print(A/B)
else:
    print("invalid operation")
```

```
#range(start,stop)# does not include the last number
value = list(range(35,60))
print(value)
```

```
##### GOLD PREDICTOR ACTIVITY #####
```

```
A = int(input("mass of metal:"))
if A in range(35,60):
    print(f'we found Gold with a mass of {A}g bosssss!')
elif A < 35:
    print(f'{A}g is too small to be gold")
else:
    print(f'{A}g is too big to be gold")
```

```
%%#####
#####
```

```
# 9. Defining a function
```

```
#A function is a block of code
```

```
#that does something — like a small machine you can reuse
```

```
def greeting(): # defining simple function
```

```
    print("Hi my name is John")
```

```
greeting()
```

#using our calculator

```
def calculator():  
    A=int(input("Enter first val ="))  
    B=int(input("Enter second val ="))  
    opr=input("Enter the operator:")  
    if opr=="+":  
        print(A+B)  
    elif opr=="-":  
        print(A-B)  
    elif opr=="*":  
        print(A*B)  
    elif opr=="/":  
        print(A/B)  
    else:  
        print("invalid operation")
```

calculator()

#alter

```
def calculator(A, B, opr):  
    if opr == "+":  
        print(A + B)  
    elif opr == "-":  
        print(A - B)  
    elif opr == "*":  
        print(A * B)  
    elif opr == "/":
```



```
    print(A / B)
else:
    print("Invalid operation")
```

```
calculator(5,6,"*")
```

```
#####
```

```
#return vs print
```

```
'''
```

```
# Difference between print() and return
```

```
- print():
```

- * Shows the result on the screen immediately.
- * Used to display output to the user.
- * Does NOT send the result back for further use.

```
- return:
```

- * Sends the result back to where the function was called.
- * Allows you to save or reuse the result later in your program.
- * Does NOT automatically display anything unless you print it.

```
'''
```

```
def add(a, b):
```

```
    c = a + b
```

```
    print(a+b)
```

```
add(7, 8)
```

```
print(add(7,8)*5)
```

```
###with return###
```

```
def add(a, b):
```

```
    c = a + b
```

```
    return(c)
```

```
add(7, 8)
```

```
print(add(7,8)*5)
```

```
##Alter cal
```

```
def calculator(A, B, opr):
```

```
    if opr == "+":
```

```
        return(A + B)
```

```
    elif opr == "-":
```

```
        return(A - B)
```

```
    elif opr == "*":
```

```
        return(A * B)
```

```
    elif opr == "/":
```

```
        return(A / B)
```

```
    else:
```

```
        return("Invalid operation")
```

```
print(f"the answer is:", calculator(5,6,"*"))
```

```
##%#####  
###
```

```
# 10. For Loop ( for repeatation and repeating a process)
```

to repeat a block of code for each item in a sequence

```
for i in range(10):  
    print(i) # takes the in between number
```

```
for i in range(10):  
    if i == 5:  
        break  
    print(i)
```

```
for letter in "hello ":  
    print(letter)
```

#####

```
def print_even_numbers(n):  
    even_numbers = []  
    for num in range(1, n + 1):  
        if num % 2 == 0:  
            even_numbers.append(num)  
    return(f"Even numbers found in a range of {n} are: {even_numbers}")
```

Call the function

```
results = print_even_numbers(8)  
print(results)
```


#####

11. While loop

i=1

```
while i<=10:
    print(i, '.Just give me my money!')
    i=i+1
```

```
#####
```

```
def money (i):
    while i<=10:
        print(i, '.Just give me my money!')
        i=i+1
```

```
money(2)
```

```
###%#####
#####
```

```
# 12. Exception handling (#code is executed even when error occurs)
```

```
'''
```

common types of exceptions are:

ZeroDivisionError: Raised when the second argument of a division or modulo operation is zero.

TypeError: Raised when an operation or function is applied to an object of inappropriate type.

ValueError: Raised when a built-in operation or function receives an argument that has the right type but an inappropriate value.

IndexError: Raised when a sequence subscript is out of range.

KeyError: Raised when a dictionary key is not found.

FileNotFoundError: Raised when a file or directory is requested but doesn't exist.

IOError: Raised when an I/O operation (such as a print statement, the built-in open() function or a method of a file object) fails for an I/O-related reason.

ImportError: Raised when an import statement fails to find the module definition or when a from

... import fails to find a name that is to be imported.

MemoryError: Raised when an operation runs out of memory.

OverflowError: Raised when the result of an arithmetic operation is too large to be expressed by the normal number format.

AttributeError: Raised when an attribute reference or assignment fails.

SyntaxError: Raised when the parser encounters a syntax error.

IndentationError: Raised when there is incorrect indentation.

NameError: Raised when a local or global name is not found

'''

the flow of the code

'''

try:

 # risky code

except SomeError:

 # what to do if error happens

'''

#enter letter

try:

 number = int(input("Enter a number: "))

 print(f"You entered: {number}")

except ValueError:

 print("That's not a valid number!")

#####

def calculator(A, B, opr):

```
if opr == "+":
    return(A + B)
elif opr == "-":
    return(A - B)
elif opr == "*":
    return(A * B)
elif opr == "/":
    return(A / B)
else:
    return("Invalid operation")
```

```
print(f"the answer is:", calculator(5,T,"*"))
```

```
#new calculator
```

```
# add letter
```

```
#5/0
```

```
def calculator():
```

```
    try:
```

```
        A=int(input("Enter first val ="))
```

```
        B=int(input("Enter second val ="))
```

```
        opr=input("Enter the operator:")
```

```
        if opr=="+":
```

```
            return(A+B)
```

```
        elif opr=="-":
```

```
            return(A-B)
```

```
        elif opr=="*":
```

```
        return(A*B)
    elif opr=="/":
        return(A/B)
    else:
        return("invalid operation")
except ValueError:
    return("Thats not a number silly!")
except ZeroDivisionError:
    return("You can not divide by zero!")
```

```
print (calculator())
```

```
##### WE ARE
DONE!!!#####
```

```
##### WELL DONE! THE STUDENT IS ONLY AS GOOD AS THE MASTER
#####
```

```
###%#####
#####
```