Vocabulary

Variable	Hold a value and can be change
String	A list of character such as number, better and symbols
Integer number	Whole number/ counting number
Float number	The umber in decimal
Syntax	Grammar/Structure of language
Modulo	Find the remainder
Boolean	True/False

Addition

string+string	combines the strings together
string+number	crash
number-+number	math(addition)

Multiplication and Exponents

string* string	crash
string* number	combines the string multiple times
number* number	math(multiply)
string** number	crash
number** number	exponent(math)

Multiplication and Exponents (cont)

string** string	crash

Example

```
mylist3 = [1, 'hello', 2.3]
print (mylist)
print (mylist2)
print (mylist3)
#how to make a list with all
number from 0-99
mynumbers = range(5)
for num in my numbers:
        print (num)
```

Example: Create function

```
def calc (num1,num2, operation)
:
       if operation == " sum ":
                  return sum(nu -
m1, num2)
       elif operation == " pro -
duc t" :
                  return produc -
t(n um1 ,num2)
       elif operation == " dif -
f" :
                  return diff(n -
um1 ,num2)
       elif operation == " -
div " :
                  return div(nu -
m1, num2)

def sum (a, b) :
        return a + b

def product (a, b) :
        return a *b
def diff (a, b) :
        return a - b
def div (a, b) :
       if b != 0:
```

Example: Create function (cont)

```
>         return a /b
   else:
      print("Error")

print (calc(10, 0, "div"))
print (calc(1,2,"sum"))
print (calc(4,2,"diff"))
print (calc(9,3,"div"))
print (calc(2,12,"product"))
```

Maximum

```
write a function that returns
the largest of two values
# name: max2
# arguments: num1, num2
# return: the largest value
def max2 (num1, num2):
         max value = num1

        if num2 > maxvalue:
                  max value = num2
         return maxvalue
print (max2( 3,4))
# write a function that returns
the largest of three values
# name: max3
# arguments: num1, num2, num3
# return: the largest value
def max3 (num1, num 2,n um3):
         max value = num1

        if num2 > maxvalue:
                  max value = num2

        if num3 > maxvalue:
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Maximum (cont)

```
>       maxvalue = num3

    return maxvalue
print (max3(3,4,8))
# write a function that returns the largest
number in a list
# name: maxlist
# argument: list
# returns the largest value in the list
def maxlist (list):
    maxvalue = list[0]
    for item in list:
        if item > maxvalue:
            maxvalue = item

    return maxvalue
list = [1,2,3,6,19,50,2,4,5]
print (maxlist(list))
```

Range()

```
#creates a list of numbers from
0 to the specified
number
numberlist = range(5)
# is the same as creating the
following list
number list2 = [0, 1, 2, 3, 4]
for num in range( 100):
        print (num) # prints all
numbers from 0 - 99
for num in range(5, 50):
        pri nt(num) #prints all
numbers from 5 - 49
```

From Worksheet

```
Write a program that repeatedly
receives positive integers from
the user. When the user enters a
negative integer, exit the loop
an print how many of the number
entered were odd and even.
evencount = 0
oddcount = 0
while True:
          use r_input = int(in -
put ("Enter the number: "))
         if user_n umber < 0:
                pri nt( " Even:
" ,ev enc ount)
                pri nt( "Odd:
" ,od dcount)
                break
         elif user_input > 0:
               if user_input % 2
==0:#even
                     eve ncount
= evencount + 1
                else:
                     odd count
= oddcount + 1
```

Even number from -100 to -1(While loop)

```
num = 0
while num > -100
      number = num -2
      print (num)
```

From worksheet

```
Determines that number is
negative, positive or zero
Ex; 4 is positive / 0 is Zero /
-8 is negative
user_num = input ("Enter the
number ")
user_num = int(us er_num)
If user_num == 0
```

From worksheet (cont)

```
>      print (user_num,"iszero")
elif user_num < 0:
     print (user_num," is negative")
elif user_num > 0:
     print( user_num, "is positive")
```

Write a function(multiplication table)

```
def multipicationTable():
      use r_input = int(in put -
("Enter a number: ")
     num = 1
      while num <= 10 :
              print (user_ -
input, "", " num ", " =", " -
use r_inputnum")
             num = num + 1
```

Functions

print()	Show information that you want on the screen
input()	Gain info from the user
int()	Change number to be number integer
float()	Change number to be decimal number
str()	All ;ist of number, letter and symbols
len()	The length of the string
#	Comment, no effect

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Math

==	equal to
!=	no equal to
<	less than
>	more than
<=	less than or equal to
>=	more than or equal to
%	Modulo, Find the remainder

Reverse Word

```
while True:
word = input( " Please enter
word")
index = 0
reverse = ' '
while int(in dex )<l en( word)
              reverse = word[i -
ndex] + (reverse)
              index = int(index)
+ 1
print( " Rev ers e:", reverse)
```

Import random

```
import random
intlist = [1,2,3 ,4,5]
random_int = random.ch oic e(i -
ntlist)
print (intlist, random _int)
fplist = [1.5,2.5, -
3.5 ,4.5 ,5.5]
random_fp = random.ch oic e(f -
plist)
print (fplist, random_fp)
strlist = ['1', '2', '3', '4',
'5']
random_str = random.ch oic e(s -
trlist)
print (strlist, random _str)
mylist = [1, 2, 3, 4, 5, 1.5,
2.5, 3.5, 4.5, 5.5, '1', '2',
'3', '4', '5']
```

Import random (cont)

```
> random_item = random.choice(mylist)
print (mylist, random_item)
myvar1 = 1
myvar2 = 2
myvar3 = 3
varlist = [myvar1, myvar2, myvar3]
random_var = random.choice(varlist)
print (varlist, random_var)
```

Function call

```
def printDefinitions (word):
         if word == " var iab -
le":
               # variable
                print ("""
               a variable is are
nothing but reserved memory
locations to store values or
something that can be change.
                " " ")
       elif word == " fun cti -
on":
               # function
                print ("""
               a function is the
thing that define and put the
code in there to reuse it again.
                " " ")
       elif word == " fun ction
call":
                #fu nction call
                print ("""
               a function call
is function that already have
code and can use it.
                " " ")
       elif word == " par ame -
ter ":
```

Function call (cont)

```
>       #parameter
       print ("""
       a parameter is something that we put
in function to define variable
       """)
   elif word == "argument":
       #argument
       print ("""
       a argument is the parameter.
       """)
   elif word == "string":
       #string
       print ("""
       a string is a letter.
       """)
   else:
        print ("Unknown word")
while True:
    user_input = input("Enter a word to
define: ")
    printDefinitions(user_input) # function
call
```

Forever While Loop

```
while True: # forever
            use r_input =
input( 'Enter a number: ')
            number = int(us -
er_ input)
            print ('The number
squared is', number ** 2)
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Decision Making/Conditional Statements:

```
if 3 < 2: #if statement must
compare two Booleans
        print ('3 is less than
2')
elif 4 < 2: #can have 0 or more
elif statements
        print ('4 is less than
2')
elif 5 < 2:
      print ('5 is less than 2')
else: #can have 0 or 1 else
statement at the end
      print ('none of the above
are True')
```

Lists:

```
mylist = [2,3,4,5] # create a
list
#select an item from a list
print (mylis t[0]) #selects
first item and displays 2
# len() determines the length of
the list
print (len(m ylist)) # displays
4
mylist.ap pend(5) # adds an item
to the end of the list
```

Exercise

```
'''1. Write a program that uses
a for loop to print out all the
items from a list
      called theList'''
theList = [1,2,3,4]
for item in theList:
        pri nt( item)
'''2. Write a program that uses
a while loop to print out all
the items from a list
      called whileL ist'''
whileL ist =[' bacon', 'cokez -
ero', 'pepsi']
num = 0
```

Exercise (cont)

```
> while num < len(whileList):
    print(whileList[num])
    num = num + 1
'''3. Write a program that repeatedly accepts
user input, prints out the length of
  whatever they type in and quits when the
user enters the word 'exit' '''
while True:
    user_input = input('Enter a word: ')
    if user_input == 'exit':
        break
    else:
        print (len(user_input))
'''4. Create a function called theFunction,
that has no parameters
and returns nothing. This function should
repeatedly accept user input until
the user enters the word 'stop'. Call this
function and run it.'''
def theFunction():
    while True:
        user_input = input('Enter a word: ')
        if user_input == 'stop':
            break
    return
#call the function(function call)
theFunction()
'''5. Create a function called computeThis,
that takes two parameters, a1 and b2.
```

Exercise (cont)

```
> The function should return the product of
both parameters. Call this function
and print the result.'''
def computeThis(a1, b2):
    return a1 * b2
print ("Product of 2 and 3 = ",computeThis-
(2,3))
'''6. Create a function called finalFunction,
that has 1 argument called string.
The function should print the argument
surrounded by "*" and return nothing.
Call this function. '''
def finalFunction(string):
    print ('*' + string + *')
finalFunction('hello')
```

Naming Convention

```
Rule for giving name
- letter
- numbers
- unders core_
                                    -


Valid name
- _myStr
- my3
- Hello_ there
Invalid name
- 3my="hi " --cannot start with
number
- first name="hi'
- first-name
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Convert to binary

```
user_number = ' '
while user_n umber != ' 0 ' :
            use r_n umber =
input( " Enter a number to
convert to binary ")
            number = int(us -
er_ number)
            bin ary _string = '
'
while( number > 0):
              rem ainder =
number%2
              bin ary _string =
str(re mai nder)+ binary _string
              number = number/
/2
print( " Binary string is",
binary _st ring)
number = number/ / 2
print( " Binary string is",
binary _st ring)
```

Countdown Machine

```
user_number = input(:What number
do you want to countdown?")
number = int(us er_ number)
countd own _string = ' '
while number > 0:
            cou ntd own _number
= countd own _string + str(nu -
mber) + " "
            number = number - 1
#print (nu mber)
print (count dow n_s tring)
```

Example

Print (2) - integer
Print (2.5) - floating point
Print ("Hello") - string
Print (mystr) - variable

Example (cont)

Print (mystr. "Hi",2,1.0) -- commas
mystr = "Hi"
mystr ◂☐ name
"Hi" ◂☐ value can change
mtstr (int(int)) ➢ 1
print (int("2")) ➢ 2
print (float(1)) ➢ 1.0 anything to a float

Example

```
def myprint(text) : #mtvar is an
argument (parameter) to he
function
        print" #" str(text) +
""")
        return #return exists
the function
myprint(1)
myprin t(2.5)
myprin t("h ell o")
def myprin tne w(text, decora -
tion) : #text and decoration are
arguments to the function
        print (decor ation +
text + decora tion)
        return
myprin tne w("h ell o", " -
+++ ")
myprin tne w("h ell o", " -=- =-
= -=- =")
myprin tne w("h ell o", " >>> -
>>> ")
def double It( number) :
        return number * 2
#return value
print (times Two(2))
myvar + timesT wo( tim esT -
wo(3)) #same as timesT wo(6)
bacuse timesT wo(3) == 6
print (myvar) # it will display
12
def areaOf Cir cle(r):
       if r <= 0:
```

Example (cont)

```
>       return "Error: invalid radius"

   pi = 3.1415
   area = pi r* 2
   return area
user_radius = float(input("Enter the radius:
")
print("The area of the circle is", areaOfCir-
cle(user_radius))
```

Conditional While Loop

```
count = 0 # start at zero
while count < 10: # loop while
count is less than 10
                pri nt( count)
#will print numbers 0 - 9
                count = count +
1 #must increase count
```

Printing values:

```
print("hello", "there")
#displays hello there
print( " hel lo" + " the re")
#displays hellothere
```

For-Loop with List:

```
forlist = [3, 4, 5, 2, 1]
for item in forlist:
      pri nt( item)
```

While Loop with List:

```
thelist = [4, 3, 2, 1, 0]
index = 0 # start at the first
item
while index < len(th elist):
            print (theli st[ -
index]) #prints each item
            index = index + 1
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Comments

```
# hashtag - everything after #
is a comment not code
" " "
Double quote - Multi-line
comment, everything in
between three double quotes is a
comments
" " "
''' Single quote - Multi-line
comment, everything in
between three single quotes is a
comments '''
```

From worksheet

```
0
01
012
0123
01234
mystring = " "
count = 0
while count < 5:
        mys tring = mystring +
str(count0
        print (mystring)
        count = count + 1
```

From worksheet

```
Use a for loop to print the
following:
0
012
0123
01234
mystring = " "
for num in range(5)
```

From worksheet (cont)

```
>     mystring = mystring + str(num)
    print (mystring)
```

From worksheet

```
Create a program to receive a
number from the user and
determine if that number is
divisible by3.
Example:
- 9 is divisible by 3.
- 7 is not divisible by 3.
user_num = int(in put ("Enter
the number: "))
if user_num%3 == 0:
      pri nt( use r_num, "is
divisible by 3")
else:
      pri nt( use r_num, " is
not divisible by 3")
```

Print all items in the list

```
Give a list called mylist, print
all items in the list using a
loop.
mylist = [1,2,3,4]
for item in mylist:
        print (mylist)
Another method
num = 0
while num < len(my list)
          pri nt( myl ist [num])
         num = num + 1
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com