***Extra Code Challenges***

All programs must be commented and have a test plan for each possible output, showing the results for run tests.

**If**

|  |  |
| --- | --- |
| **Input** | **Output required** |
| 0% to 31% | Grade U – you must work harder |
| 32% to 39% | Grade E – room for improvement |
| 40% to 48% | Grade D – reasonable work |
| 49% to 56% | Grade C – good work |
| 57% to 65% | Grade B – very good work |
| 66% to 100% | Grade A – excellent work |

Write a program using If, else if and else to get an input for percentage score for a student and then output the following according to the input:

inpt = True

while inpt == True:  #allowing input to reject values not 0-100

    inpt = False

    grade\_percentage = float(input("Enter your grade percentage: ")) # user input

    if grade\_percentage > 100 or grade\_percentage < 0:

        print("Your percentage was not vaild")

        inpt = True

    elif grade\_percentage >= 0 and grade\_percentage <= 31:

        print("Grade U – you must work harder")

    elif grade\_percentage >= 32 and grade\_percentage <= 39:

        print("Grade E – room for improvement")

    elif grade\_percentage >= 40 and grade\_percentage <= 48:

        print("Grade D – reasonable work")

    elif grade\_percentage >= 49 and grade\_percentage <= 56:

        print("Grade C – good work")

    elif grade\_percentage >= 57 and grade\_percentage <= 65:

        print("Grade B – very good work")

    else:

        print("Grade A – excellent work")

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Test Case** | **Expected Result** | **Actual Result** |
| Float | 44.4 | Grade D | ‘’ |
| Int | 60 | Grade B | ‘’ |
| Extreme values | 100, 0 | Grade A, Grade U | ‘’ |
| Incorrect values | 120, -41 | Asks user to re-enter % | ‘’ |

**For loop**

Produce a for loop which adds up numbers 1 to 10

e.g. 1+2+3+4 to 10

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Test Case** | **Expected Result** | **Actual Result** |
| Pre challenge (10) | 10 | 55 | ‘’ |
| Int | 60, (7,31) | 1830, 475 | ‘’ |
| Extreme values | 2000, (-2,99) | 2001000, 4947 | ‘’ |
| Float | 44.4, (10.5, 20.7) | “Please enter integers only” | ‘’ |

# Task 1

total = 0

for number in range(11):    #range produes a list 0-10 to iterate through

   total += number

print(total)

#Challenge

def sum\_up\_to\_my\_number():

    total = 0

    try :

        my\_number1 = int(input("Please enter your start number: "))  #user inputs

        my\_number2 = int(input("Please enter your end number: "))

        for number in range(my\_number1, my\_number2+1):

            total += number

        return total

    except ValueError:    # prevents entry of a float

         return "Please enter integers only try again!"

print(sum\_up\_to\_my\_number())

Extra challenges:

1. Improve this program so that it allows the user to input the end value instead of 10
2. Further improve the program so that the user can also input the start value instead of 1

**While loop**

Produce a while loop which adds up numbers 1 to 10

e.g. 1+2+3+4 to 10

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Test Case** | **Expected Result** | **Actual Result** |
| Pre challenge (10) | 10 | 55 | ‘’ |
| Int | 60, (7,31) | 1830, 475 | ‘’ |
| Extreme values | 2000, (-2,99) | 2001000, 4947 | ‘’ |
| Float | 44.4, (10.5, 20.7) | “Please enter integers only” | ‘’ |

#2

count = 0

total\_new = 0

while count <11:

    total\_new = total\_new + count

    count +=1

print(total\_new)

# 2 - Challenge

def add\_up():

    try:

        total\_new = 0

        count = int(input("Enter a starting value: "))

        end = (int(input("Enter a starting value: ")) + 1)

        while count < end:

            total\_new = total\_new + count

            count +=1

        return total\_new

    except ValueError:

        return "Please enter integers only"

print(add\_up())

Extra challenges:

1. Improve this program so that it allows the user to input the end value instead of 10
2. Further improve the program so that the user can also input the start value instead of 1

**Do While loop**

#3

def add\_up\_do():

    try:

        total\_new = 0

        count = int(input("Enter a starting value: "))

        end = (int(input("Enter a starting value: ")) + 1)

        if count > end:

            return "Error"

        while True:

            total\_new = total\_new + count

            count +=1

            if count == end:

                break

        return total\_new

    except ValueError:

        return "Please enter integers only"

print(add\_up\_do())

You can only produce a do while using break in python as they don’t use them.

Produce a do while loop which adds up numbers 1 to 10

e.g. 1+2+3+4 to 10

Extra challenges:

1. Improve this program so that it allows the user to input the end value instead of 10
2. Further improve the program so that the user can also input the start value instead of 1

**Arrays**

Produce a simple array with 10 elements. Use a for loop to add the following data to your array;

2,4,6,8,10,12,14,16,18,20

Then output the data in the array using another for loop.

array = [2,4,6,8,10,12,14,16,18,20]

new\_array = []

for number in array:

    new\_array.append(number)

for number in new\_array:

    print(number)

**2 D array**

Produce a 2D array and input the following data to make a times table:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | | |  | |  | |  | |
| **Showing the 2D array with stored data:** | | | | | | | | | | |
|  | *0* | | *1* | | | | *2* | | *3* | | *4* |
| *0* | 1 | | | 2 | | 3 | | 4 | | 5 | |
| *1* | 2 | | | 4 | | 6 | | 8 | | 10 | |
| *2* | 3 | | | 6 | | 9 | | 12 | | 15 | |
| *3* | 4 | | | 8 | | 12 | | 16 | | 20 | |
| *4* | 5 | | | 10 | | 15 | | 20 | | 25 | |

Then produce the following output from your 2D array:

**Showing output:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 2 | 4 | 6 | 8 | 10 |
| 3 | 6 | 9 | 12 | 15 |
| 4 | 8 | 12 | 16 | 20 |
| 5 | 10 | 15 | 20 | 25 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Test Case** | **Expected Result** | **Actual Result** |
| Call number | Array\_2d[5] | [5, 10, 15, 20, 25] | ‘’ |
|  | Array\_2d[2][0] | 3 | ‘’ |
|  | Array\_2d[4][4] | 25 | ‘’ |

#2D Array

cols,rows = (5,5) #columns and rows for 2d array defined

array\_2d = []           #declarin empty variable to append to

for i in range(1,rows+1):

    row = []

    for j in range(1,cols+1):

        row.append(j\*i)

    array\_2d.append(row)

for row in array\_2d:

    print(row)

**Functions**

Write your own function for allowing two numbers to be input into the function and then the two numbers added together.

Add to the above function to make it into a calculator, so that 3 numbers can be input as follows:

First and second number – numbers as above

Third number – 1 for +, 2 for -, 3 for x, 4 for /

#function task

def calcu():

    inpt = True

    while inpt == True:

        try:

            inpt = False

            num1 = float(input("Enter your first number: "))

            num2 = float(input("Enter your second number: "))

            num3 = int(input("Chose operation: 1 for +, 2 for -, 3 for x, 4 for /: "))

            if num3 not in range(1,5):

                print("Only enter 1, 2, 3 or 4!")

                inpt = True

            elif num3 == 1:

                return num1 + num2

            elif num3 == 2:

                return num1 - num2

            elif num3 == 3:

                return num1 \* num2

            else:

                return num1 / num2

        except ValueError:

            print("Only enter 1, 2, 3 or 4!")

            inpt = True

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Test Case** | **Expected Result** | **Actual Result** |
| Int | (7,31,3) | 217.0 | ‘’ |
| Extreme values | (20,000,2.8,4) | 7142.86… | ‘’ |
| Float | (0.9,0.5,1) | 1.4 | ‘’ |
| Error values | (10,15,7) | “Only enter 1, 2, 3 or 4!”  Reruns input prompt | ‘’ |