

# loops\_hw\_soloution

September 2, 2021

## 1 While Loops

**Task 1:** Using a while loop, display numbers in range of 1 to 10 (including 1 and 10).

```
[ ]: i = 1
      while i <= 10:
          print(i, end=" ")
          i += 1
```

**Task 2:** Using a while loop, code two different programs to display even numbers in range of 0 to 10 (including 0 and 10).

*Hint:* you can use *break*.

```
[ ]: i = 0
      while i<=10:
          if i %2==0:
              print(i)
          i += 1
```

```
[ ]: i = 0
      while i<=10:
          print(i)
          i += 2
```

### Simple Iteration

**Task 3:** Using a while loop, display all the elements in the given list in separate lines.

*Hint:* you can use *len()* funtion.

```
[ ]: my_list = [{"Name" : "Jake"} , {"Age" : "16"} , "I Love Python!!" , [1,2,3] ,
               ↪ (1,2) , 1.1]
```

```
[ ]: i = 0
      while i < len(my_list):
          print(my_list[i])
          i += 1
```

### Run-Time Error

Jake is trying to display his classmates grades in separate lines. He has made the list and coded his program but there are some errors. Help him debug the code and then fix the code.

**Task 4:** Explain the code below and what happens when you run it and fix the bugs.

*Hint:* If you run a program that enters an infinite loop, you can force Python to quit by pressing Ctrl+C. To stop the program in Jupyter Notebook, go to Kernel -> Restart & Clear Output.

```
[ ]: #Task4
Grades = [{"Name" : "Jake" , "Grade" : "B+"} , {"Name" : "Leo" , "Grade": "A-"}
        ↪,
        {"Name": "Olivia", "Grade" : "A+"} , {"Name" : "Ruby", "Grade": "D+"}]
i = 0
while i <= len(Grades):
    Grade = Grades[i]
    print ("{} Got the Grade of ".format(Grade['Name'])) , end = ""
    print ("{}".format(Grade['Grade']))

# The indices of items in Grades range from 0 to i-1,
#but this loop attempts to access the Grades[i]
#also, the value of i never changes, so the loop never stops!
```

```
[ ]: Grades = [{"Name" : "Jake" , "Grade" : "B+"} , {"Name" : "Leo" , "Grade": "A-"}
        ↪,
        {"Name": "Olivia", "Grade" : "A+"} , {"Name" : "Ruby", "Grade": "D+"}]
i = 0
#Fixed
while i < len(Grades):
    Grade = Grades[i]
    print ("{} Got the Grade of ".format(Grade['Name'])) , end = ""
    print ("{}".format(Grade['Grade']))
    #added this line:
    i += 1
```

**Task 5:** Using a while loop, code a program in which you can get unlimited integers and for each integer, - if the number is odd, display “The number is odd!”. - if the number is even, display “The number is even!”.

*Bonus:* Change the program so that the loop ends when the input is ‘e’.

```
[ ]: while True:
    n = int(input())
    if n %2 == 0:
        print("The number is even!")
    else:
        print("The number is odd!")
```

**Task 6:** Using a while loop, code a program in which you can get unlimited numbers each representing a Celsius degree, and for every number, display the converted degree to Fahrenheit in each

line.

```
[ ]: #Celsius to Fahrenheit conversion formula is: F = (c*9.5) + 32

while True:
    c = float(input())
    f = (c*9.5) + 32
    print(f)
```

## 2 For Loops - Iteration

**Task 7:** Print all the squares of the numbers of the given list and all the square root the given tuple in separate lines.

```
[ ]: my_list = [1.2 , 0 , 4 , 22 , -2 , 3]
my_tuple = (-3 , 1 , 2 , 3 , 0)
```

```
[ ]: for item in my_list:
    print(item**2)

for item in my_tuple:
    print(item**(1/2))
```

**Task 8:** Using a for loop and iteration, determine whether or not there's one or more 'h' in the string below. - if this letter could be found, display "The letter was found" otherwise display "The letter not found"

```
[ ]: Greeting = "Welcome Welcome to our Python Course!"

found = False
for i in Greeting:
    if i == 'h':
        print("The letter was found")
        found = True

if found == False:
    print("The letter was not found")
```

Jake has entered his classmate's names and grades to a dictionary in Python (names are the keys and grades are the values). Now he wants to write a program that gets a name and determines whether or not the name has been found in the dictionary. if the name is found, then the grade belonging to the name must be displayed. Otherwise another message appears that declares this name hasnt been entered yet!

**Task 9:** Help Jake code this program, using a for loop and iteration, - if the name is not found, then display "No grade has been registered for this name." - if the name is found, then display "{Entered\_name} has got the grade of {Grade}."

**Input Format:** - the first line contains the entered name (string).

Please note that the person's name starts with a capital letter and other letters are small.

```
[ ]: grades = {"Amy" : "B" , "Brian" : "C+" , "Charles" : "B-" , "Celia" : "C-" ,  
             ↪ "Jake" : "B+" , "Justin" : "D" , "Olivia" : "A" , "Riley" : "D-" , "Sam" :  
             ↪ "B+" , "Tara" : "A" , "Umar" : "C+" , "Zoe" : "B"}  
  
std = input()  
name_found = False  
for name, grade in grades.items():  
    if name == std:  
        print("{} has got the grade of {}".format(name, grade))  
        name_found = True  
        break  
  
if not name_found:  
    print("No grade has been registered for this name.")
```

### 3 Nested Loops

**Task 10:** In every tuple in the given list below, there's a character 'f' or 'c' which determines the number in that tuple is whether in Fahrenheit or Celsius respectively. using nested loops, if the temperature is declared in Celsius, convert it to Fahrenheit and display it and if the temperature is in Fahrenheit, convert it to Celsius and display it.

```
[ ]: my_list = [('c' , 12) , ('c' , 70) ,  
               ('c' , -1) , ('c' , 13) ,  
               ('f' , 100) , ('f' , 70) ,  
               ('f' , 10) , ('f' , 32)]
```

```
[ ]: for item in my_list:  
    if item[0] == 'f':  
        c = (item[1]-32)/(9.5)  
        print(c)  
    else:  
        f = (item[1]*9.5) + 32  
        print(f)
```

### 4 For Loops - range

**Task 11:** Using range function and a for loop, code two different programs to display odd numbers in range of 0 to 11 (including 0 and 11).

```
[ ]: for i in range(0,12):  
    if i%2!= 0:  
        print(i)
```

```
[ ]: for i in range(1,12,2):  
      print(i)
```

**Task 12:** Using range function and a for loop, display every character in the given name in separate lines.

```
[ ]: name = "Jacob"  
  
for letter in name:  
    print(letter)
```

## 5 Nested Loops

**Task 13:** Using range function and nested for loops, display the sequence of the numbers below to  $n = 10$ .

0

1 0

2 1 0

3 2 1 0

4 3 2 1 0

5 4 3 2 1 0

6 5 4 3 2 1 0

in this example we have gone far till  $n = 6$ .

*Hint:* To print “1 0” in one line, you can also do this

```
print(1 , end = " ")
```

```
print(0 , end = " ")
```

```
[ ]: n = 10  
for i in range(n+1):  
    for j in range(i, -1,-1):  
        print(j, end=" ")  
    print()
```

```
[ ]:
```