

Practice exercises: Control flow

The FOSSEE Group

Department of Aerospace Engineering
IIT Bombay

Mumbai, India

Note: Python 2.x and 3.x

If you are using Python 2.x

- Use `raw_input` instead of `input`
- Use the following for `print`

```
from __future__ import print_function
```

Exercise: simple `if`

- 1 Ask the user to enter an integer
- 2 If the number is positive, print "Positive"
- 3 If the number is negative, say -10 print:
"Negative of 10"
- 4 If the number is zero, print "Zero"
- 5 Finally, print "Done"

Solution

```
x = int(input())
if x < 0:
    print("Negative of", abs(x))
elif x > 0:
    print("Positive")
else:
    print("Zero")
print("Done")
```

Exercise: simple `if` 2

- 1 Ask the user to enter two integers, one on each line say **`x`** and **`y`**
- 2 If **`x`** is larger than **`y`** print, "first", store the value of **`x`** in a variable **`highest`**
- 3 If **`y`** is larger than **`x`** print, "second", store the value of **`y`** in a variable **`highest`**
- 4 If they are equal, print "same", store either as the **`highest`**
- 5 Finally, print the square of the **`highest`**

Solution

```
x = int(input())
y = int(input())
if x > y:
    print("first")
    highest = x
elif y > x:
    print("second")
    highest = y
else:
    print("same")
    highest = x
print(highest*highest)
```

Simple `while` loop 1

- 1 Use a while loop to print the numbers from 1 to 10

Solution

```
x = 1
while x < 11:
    print(x)
    x += 1
```


Simple `while` loop 2

- 1 Use a while loop to print the odd numbers from 1 to 10

Solution

```
x = 1
while x < 11:
    print(x)
    x += 2
```

Simple `while` loop 2

- 1 Use a while loop to generate the numbers from 1 to 10
- 2 If a number divisible by 3 is found, print `***`
- 3 If a number divisible by 5 is found, print `*****`
- 4 Otherwise just print the number

Solution

```
x = 1
while x < 11:
    if x%3 == 0:
        print('***')
    elif x%5 == 0:
        print('*****')
    else:
        print(x)
    x += 1
```

Simple `while` loop 3

- 1 Use a while loop to generate the numbers from 10 to 1
- 2 For each number, print that many `*` characters
- 3 That is, if the number is 1 print `*`, if the number is 3, print `***` etc.

Solution

```
x = 10
while x > 0:
    print(' *' * x)
    x -= 1
```

Simple `while` loop 4

- 1 Ask the user for an integer, **`n`** (≥ 2)
- 2 Use a `while` loop to print **`n`** numbers uniformly spaced between 0 and 1
- 3 Include both 0 and 1

For example if the user entered 3, print

0.0

0.5

1.0

Solution

```
n = int(input())
dx = 1.0/(n-1)
x = 0.0
while x < (1.0-dx/2):
    print(x)
    x += dx
print(1.0)
```


Simple `for` loop 1

- 1 Ask the user for an integer
- 2 Use a `for` loop to generate the multiplication table for the number (up to 10)

If the user entered 5, print the following

5 x 1 = 5

5 x 2 = 10

5 x 3 = 15

. . .

Solution

```
n = int(input())  
for i in range(1, 11):  
    print(n, 'x', i, '=', n*i)
```

for loop Fibonacci

- 1 Ask the user for an integer, **n** (≥ 1)
- 2 Print the first **n** numbers of the Fibonacci sequence

Solution

```
n = int(input())
a, b = 0, 1
print(a)
for i in range(n-1):
    print(b)
    a, b = b, a+b
```

for loop over list 1

- 1 Ask the user for an integer, **n**
- 2 Use the range to create numbers from 1 to **n**
- 3 Print each value in this in a separate line

For example let us say user enters **3**, print

1
2
3

Solution

```
n = int(input())  
for x in range(1, n+1):  
    print(x)
```

for loop over array

- 1 Ask the user for an integer, **n**
- 2 Use numpy's **linspace** to create n points from 1 to 2
- 3 Print each value in this in a separate line

For example let us say user enters **3**, print

1.0

1.5

2.0

Solution

```
from numpy import linspace
n = int(input())
data = linspace(1, 2, n)
for x in data:
    print(x)
```


for loop over list 2

- 1 Ask the user for an list of fruits separated by a comma
- 2 Print each fruit in a separate line

For example let us say user enters **apple, pear**,
print

```
apple  
pear
```

Solution

```
fruits = input()
fruits = fruits.split(',')
for fruit in fruits:
    print(fruit)
```

for loop over string

- 1 Ask the user for a string
- 2 Print each character of the string using a for loop

For example let us say user enters **box**, print

b

o

x

Solution

```
s = input()  
for char in s:  
    print(char)
```

Nested `for` loops

- 1 Let us say the user supplies an integer, `n` (empty prompt)
- 2 Print an `n x n` matrix where each entry is the sum of the row + column

For example let us say user enters 3, print

```
0 1 2
1 2 3
2 3 4
```

Solution

```
n = int(input())  
for i in range(n):  
    for j in range(n):  
        print(i+j, end='  ')  
    print()
```

Exercise: Fibonacci divisible by 4

- 1 Find the first number in the Fibonacci sequence divisible by 4 but > 8 and less than 500.

Solution

```
a, b = 0, 1
while b < 500:
    if b % 4 == 0 and b > 8:
        print(b)
        break
    a, b = b, a+b
```