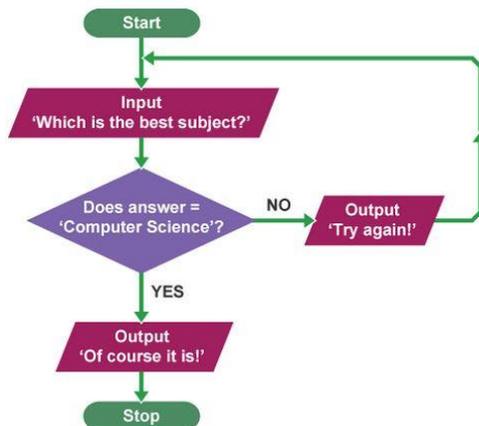


What is Programming?



Let's take a moment to remind ourselves about what we are learning in this course and why.

If you can remember to the beginning of the course, basically we stated that this course is designed to show you how you can use computers **solve problems** and **make cool stuff**. We also said that *all* computer programming languages have a few key aspects:



1. **Input**
2. **Output**
3. **Decisions**
4. **Calculations**
5. **Repetition**

You can do
ALL of these!



So far on our journey we have actually covered the *basics* of *most* of the stuff above! Good for you! Look at the words above. Can you do those things with python? You sure can!

We will now continue on our journey, adding to our tool kit, and strengthen your ability to **solve problems** and create **cool stuff** with computers!

More Repetition (Looping)

While loops

So far in the course we have mostly worked with **For** loop. For loops are great and can do most of what we want, but we will take a moment to explore a second way to repeat stuff in computer programming (**While Loops**).

While loops

Sometimes the number of times the loop is run *isn't specified in advance*. Rather, the designated block is executed repeatedly as long as some condition is met.

```
for x in range(6):
    print(x)

fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)
```

In both cases above, the repetition will only occur for a **specific amount** of times: 6 for the first block of code and 3 for the second block of code.

Sometimes we want a loop to continue for an **unknown amount of times** depending on **certain conditions**. We can do this with **while loops**. But first let's look at some simple examples of while loops:

```
i = 1
while i < 6:
    print(i)
    i = i + 1
```

```
i = 1
while i <= 10:
    print(i ** 2)
    i += 1      #important - {i+=1 is shorthand in coding for i=i+1}
```

Exercise#1

Run the code blocks above in trinket and **save them as exercise #1**. Then use a **while loops** to do the following:

1. Create a program using a **while loop** that will print out numbers 1,3,7,9,11,13,15
2. Use a while loop to print a count down from 10 to 0 and then print "blast off!!!"
Use the `sleep()` method to pause one second between printing each number of the count down.
3. Use a while loop to allow a user to enter different *names* of video games into a **list** until the length of list is 5 *names* long.

In most of the examples in exercise#1 we could have used a **for** loop instead of a while loop. This is often the case.

But there are some instances where it's much easier to use a **while** loop:
When we want to repeating things for an **unknown amount of times** depending on **certain conditions**....then a **while** loop is our best choice. Let's have a look:

Exercise#2

Type the code below into trinket and notice that each game *could* go on forever!!!! This is very different to the classic **for** loops we have been using...**save your work in exercise#2**

```
import random
while roll_again == "yes" or roll_again == "y":
    print("Rolling the dice...")
    time.sleep(1)
    dice = random.randint(1, 6)
    print("Dice: ", dice)
    roll_again = input("Do you want to roll again? (yes/y)")
print("You ended the game!")
```



```
import random
power = 15
while power > 0:
    answer=input('you meet a enemy. Do you want to battle?')
    if answer=="yes":
        battle_points=random.randint(-10,2)
        power=power+battle_points
        print("Your power is %d." % power)

print("\nOh no, your power dropped to 0! Game Over.")
```



Now create your own **coin flipping game** using *at least* one while loop to end the game.
Your game should do the following:

1. Introduce the game with some instructions.
2. Allow the user to flip a coin using `random.choice` of a "coin-side" list:
`coin_sides=['heads', 'tails']`
3. Each time they flip a coin, count the number of times **in a row** either heads or tails has been selected (hint for how to do this on next page).
4. End the game if the person gets either 3 head or 3 tails *in a row*



```

flip=random.choice(coin_sides)
    if flip=='heads':
        print ("heads")
        heads_counter+=1
        tails_counter=0    # each time a head is counted your reset the tails.

    else:
        print ("tails")
        tails_counter+=1
        heads_counter=0    # each time a tails is counted your reset the heads.

```

Infinite loops with while loops

Another thing that is tricky to do with **for** loops is make a procedure repeat **forever**. We can do this easily with a **while loop**

Exercise#3

Enter the following into trinket and make sure you **understand how each works**.
*Make sure you save your work as **exercise#3**.*

```

while True:
    num = int(input("Enter an integer: "))
    print("The double of",num,"is",2 * num)

```

```

vowels = "aeiouAEIOU"

# infinite loop
while True:
    v = input("Enter a vowel: ")
    # condition in the middle of loop
    if v in vowels:
        break # break will end the infinite loop
    print("That is not a vowel. Try again!")

print("YES! This is a vowel. Thank you!")

```

Exercise#3 Continued...

```
while True:
    i = input('Please type the following sentence exactly \'The man and his
brown fox.\': ')
    if i.strip() == 'The man and his brown fox.':
        break # break will end the infinite loop
    else:
        print('Nope! Please try again')

print('Well done you did it!')
```

Exercise#4

Write a program to keep asking for a number until you enter a negative number. At the end, print the sum of all entered numbers.

Exercise#5

Write a program to keep asking the user for the name of his friends and then adds each name to a list called `friends_list=[]` until the user inputs "no more friends". At this point the program will print out the list of friends in alphabetical order and the list will NOT include the 'no more friends' entry.

Exercise#5

Create a program that prints out the countries in the list below on separate line using a **while loop**. You may NOT use a for loop.

```
clist = ["Canada", "USA", "Mexico", "Russia", "Austria", "Guyana", ]
```

Exercise#6

Write a while loop that adds all the numbers up to 100 (inclusive).

Exercise#7

You want to get rid of some mistakes in a list of friends that you are inviting to a party. Using **while loop** and an **if statement** to write a **function** named `party_guests()` which appends all the elements in a list to a new list unless the element is an empty string: ""

```
people=["Joe", "Sarah", "Mike", "Jess", "", "Matt", "", "Greg"]
#Type your code here.
def party_guests(list):
print(party_guests(people))
```

Exercise#8

Create a **while loop** that will go through a **list** of numbers and count the number of elements that have a value of above 50.

```
lst=[10, 99, 98, 85, 45, 59, 65, 66, 76, 12, 35, 13, 100, 80, 95]
```

Exercise#9

Add the sample code shown to create a program that will tell you how many year it will take to get to your "target" cash amount given a particular interest rate and original balance. You will have to accept: **current balance**, **target**, and **yearly interest rate** from the user.

```
while balance < TARGET :
    year = year + 1
    interest = balance * RATE / 100
    balance = balance + interest

print(year)
```

Exercise#10

Create a **while loop** that accepts numbers from a user (*one at a time*) and puts them into a **list**. Have the loop end **if** the user ever input two numbers in a row that are identical.

```
Number: 7
Number: 2
Number: 9
Number: 9
Number: 9
Sorry that's three of the same numbers in a row!...you list is: 7,2,9,9,9
```