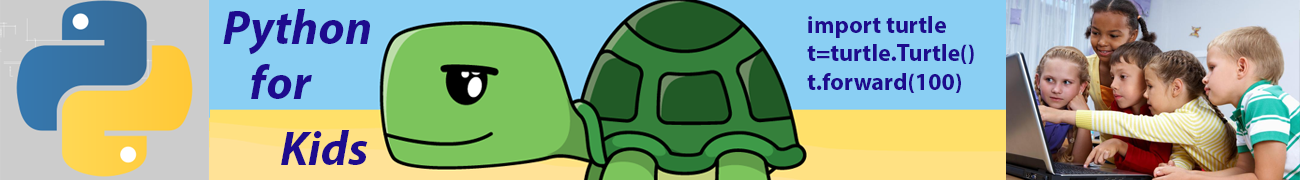
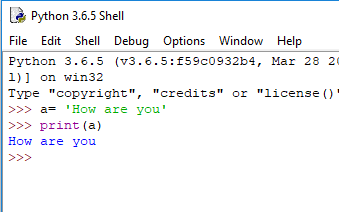
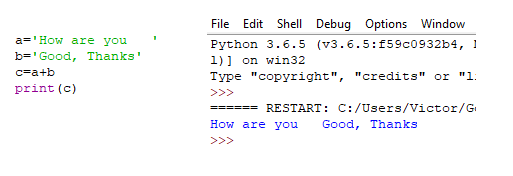
****

**Lesson 9: Strings, Lists**

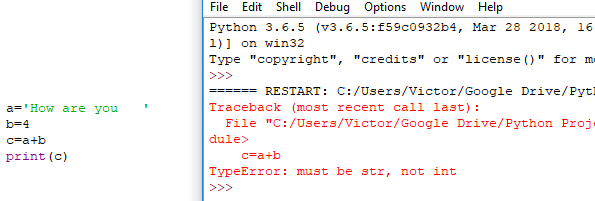
As we discussed before (Lesson #7), Python can remember *values* of several types, including **number values** (like 7, 42, or even 98.6) and **strings.** A word**-**or any collection of letters and symbols- is called a string (letters, symbols, words, sentences). Strings can include letters, numbers, spaces, and symbols such as stops and commas. You define a string by enclosing the characters in single (' ') or double quotes (" "), like so. For example, variable a=' How are you' is a string.



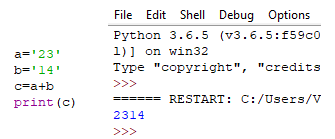
We can add two string variables. Example below shows how to do it. (Code –from left side, result is from right side)



When we add two string variable a to string variable b, we get new string variable, which is a combination of two variables a and b. **Keep in mind that you can’t add string type data and number type data.**



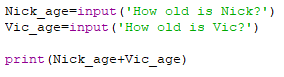
We can add two numbers if both of them are string type, like this:



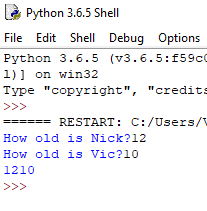
In this example both '23' and '14' are string made of numbers, not number types. So, when you add them together you get a longer string '2314' that is combination of the two strings.

Sometimes you need to operate with numbers and string in one program. How to do it. Let’s discuss the following example (**this code uses function called input**). Don’t forget click button ENTER after input data.

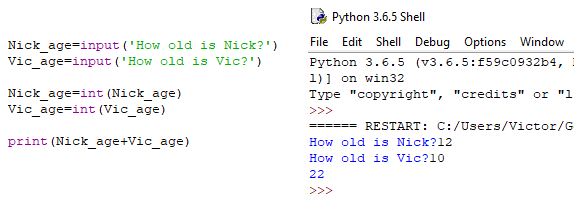
So, this code looks, like this:



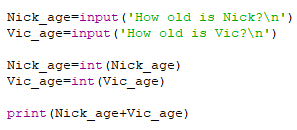
Code requests the age of Nick and Vic and must print the sum of their age. Let’s see the result



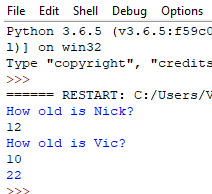
**Result is wrong because both input ages are string variables.** How to fix this problem? Very simple. You need to convert string variable number 12 and string variable number 10 to the integer numbers.



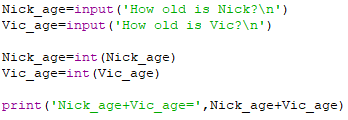
If you want the input to be on its own line, then you could just add symbol **\n** as shown below:



**Result:**

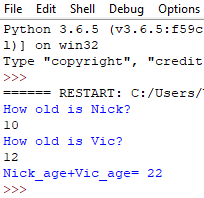


If you want to add comments to the result you need modify code as:

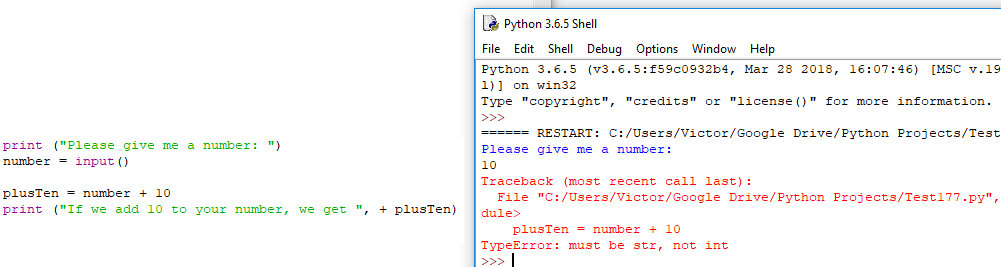


Last line includes string data 'Nick\_age+Vic\_age=' and integer number Nick\_age+Vic\_age.

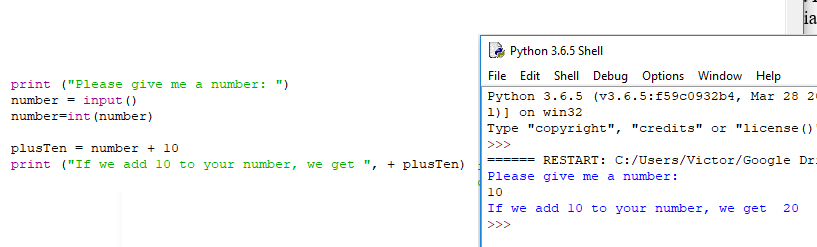
Result:



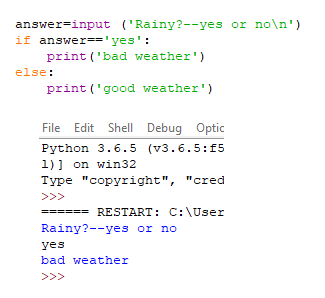
Another example



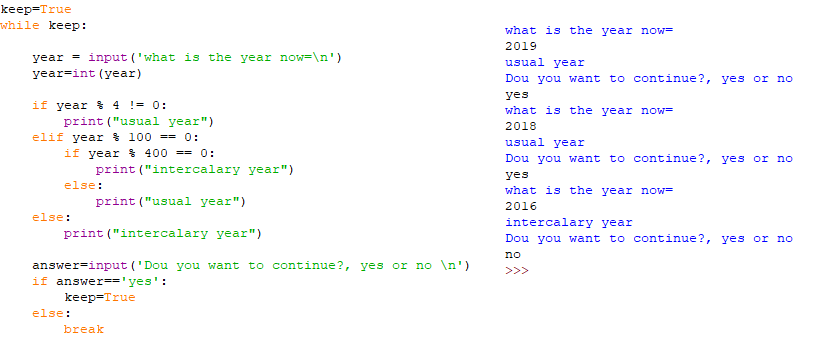
We can’t get correct result because line #3 of the code adds string variable “number” with integer 10. Again, to fix this problem we have to convert string variable “number” to integer number. Below, we show corrected code and accurate result



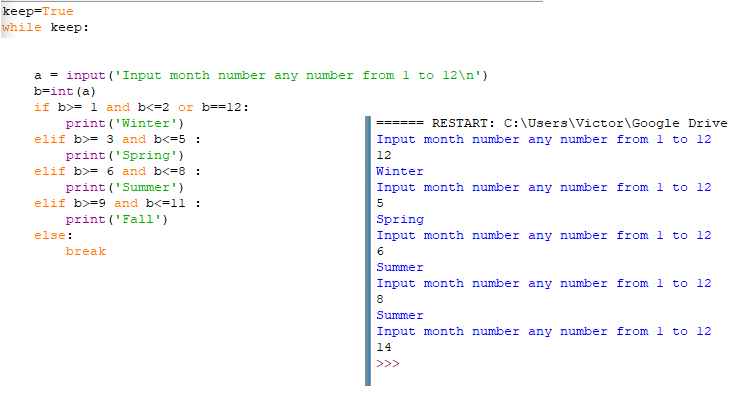
Example: Bad or good weather with Python



Example: what is the year now🡪regular or intercalary(leap)



Example: Winter, Spring, Summer, Fall

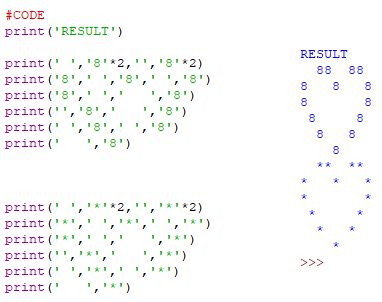


You know that 10 multiplied by 3 is equal to 30, of course. But what’s 10 multiplied by '3' ('3' is a string). Here’s Python’s answer:

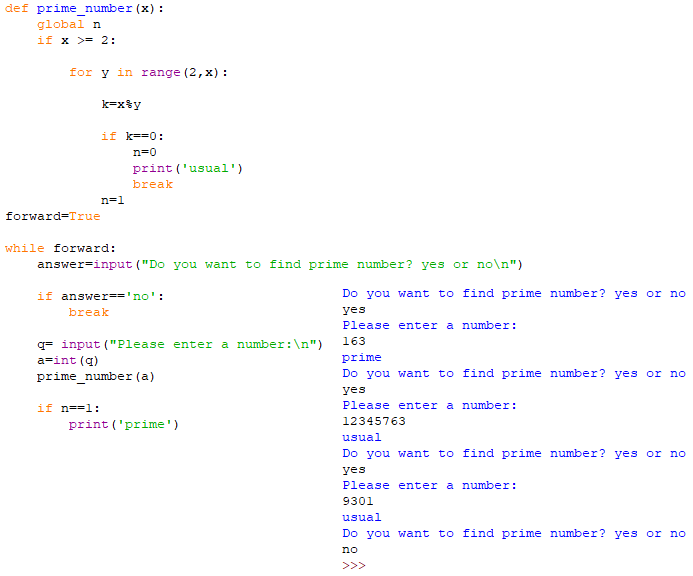


We see that the result is 10 times of digit 3.

Example: Heart pattern with strings



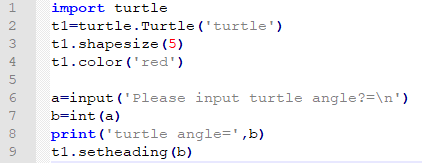
Example: Find prime number



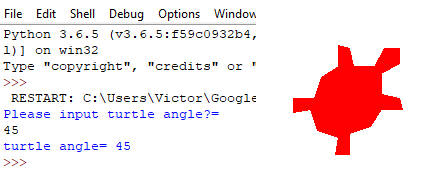
Now, Let’s demonstrate example how to turn turtle with input code

1. **Example #1** (Turtle turns 45 degrees)

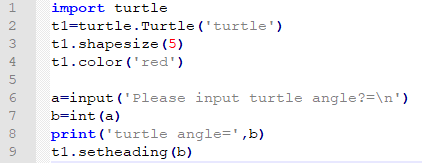
Code:

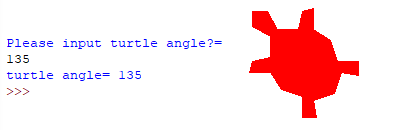


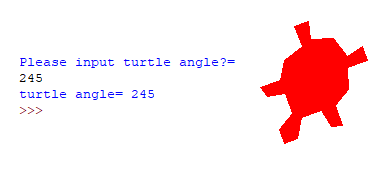
Result:

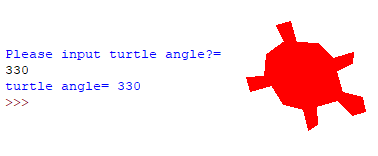


1. **Example #2**

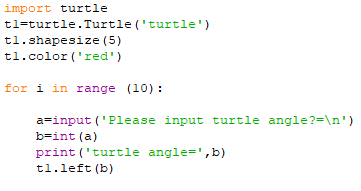








You can include input code line in the loop. In this case program requests you to input data a few times and you will see the results for each request:



**LISTS**

When you want to store a lot of data, you may need to use a list. A list can hold many items together and keep them in order. Python gives each item a number that shows its position in the list. A list is a group of values, separated by commas, between square brackets, []. We can store any value types in lists, including numbers and strings. We can store a few objects as turtles in list (**Lesson 12**). Let’s go to list example, called color\_list:

color\_list=['red', 'blue', 'gold', 'green', 'yellow', 'pink’]

This list consists of 6 string variables. If you want to do anything with one of the words from that list, you can use something called the index- the position of the item in the list. So, a list is a structure in Python where items are kept in order. Each entry is given a number that you can use to refer back to it.

color\_list[0]= 'red'

color\_list[1]= 'blue'

color\_list[2]= 'gold'

color\_list[3]= 'green'

color\_list[4]= 'yellow'

color\_list[5]= 'pink'

Below it is shown a list that includes integer numbers

number\_list=[3,5,7,11,13,17,19,23,29,31]

Now we create our first program with a list. Let’s calculate the following sums:

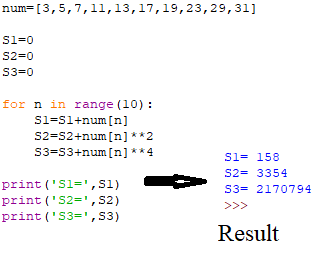
S1=3+5+7+11+13+17+19+23+29+31

S2=32+52+72+112+132+172+192+232+292+312

S3=34+54+74+114+134+174+194+234+294+314

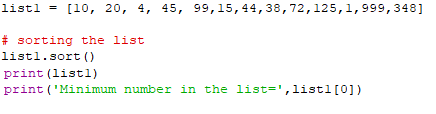
All of these integers are prime numbers. Code, shown below, calculates these values S1, S2, and S3. To make calculations we use list called *num.*

1. **Example #3** (Math calculations using List with integer values)



1. **Example #4** (Sort list numbers in the order from smallest item to largest values)

Code:



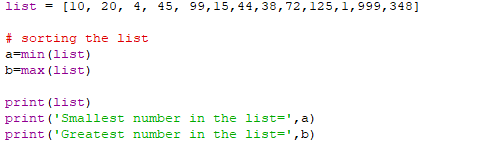
Result:

****

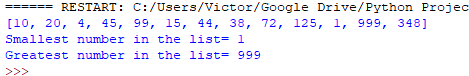
In this program we used line list1.sort that sorts all numbers in order from minimum to maximum value.

1. **Example #5** (Find smallest and greatest items in the list with numbers)

Code:



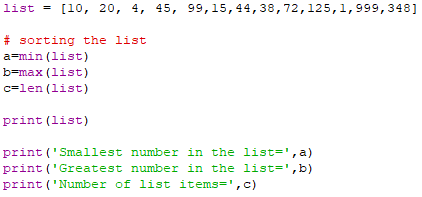
Result:



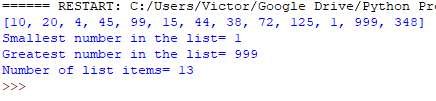
If you want to calculate the number of items in the list you have to use command len(list), for example, as shown below

1. **Example #6** (Calculate the number of items in the list)

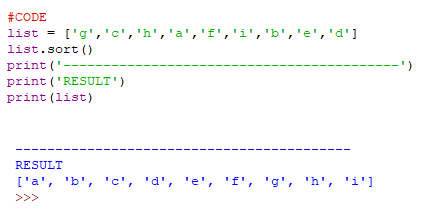
Code:



Result:



1. **Example #7** (Sorting of the list with strings variables)

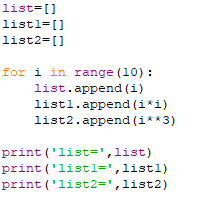


All letters are sorted according to the alphabet.

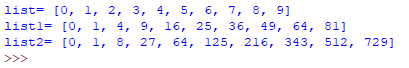
1. **Example #8** (Add to the end of a list additional items)

You can add any item numbers to the end of the list whatever you want, for example,

Code:



Result:



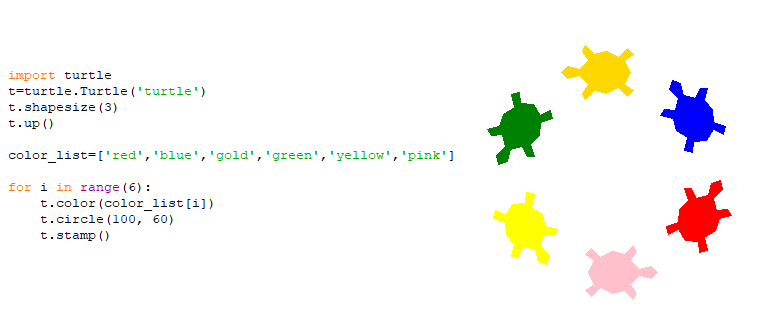
In this example we have three lists: list; list1; and list2. All of them initially are empty. However using lines “list.append(i)” , “list1.append(i\*i)”, and “list2.append(i\*\*3)” we add to the end of each list integer numbers equal

i for list,

i\*i for list1,

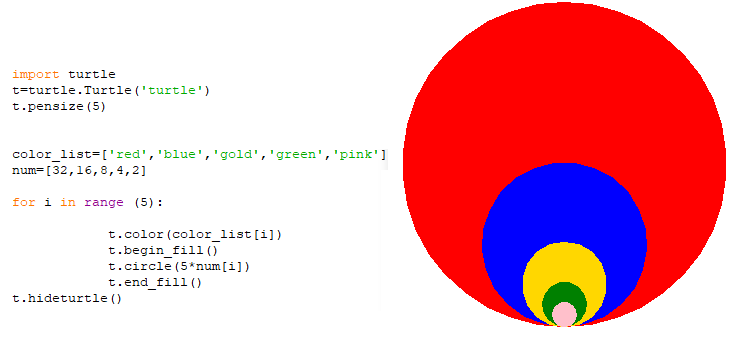
i\*\*3 for list2.

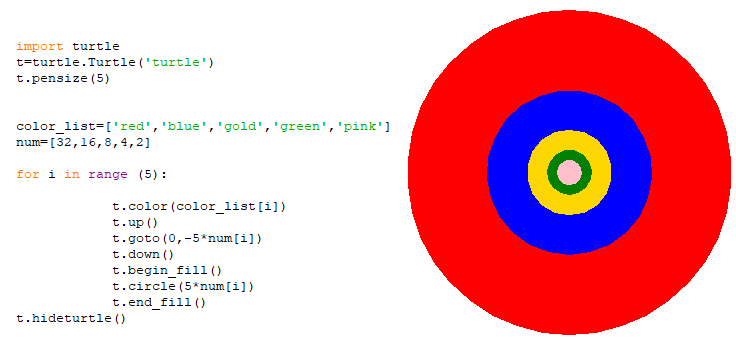
1. **Example #9** (Number of different colour turtles, code uses list with string variables)



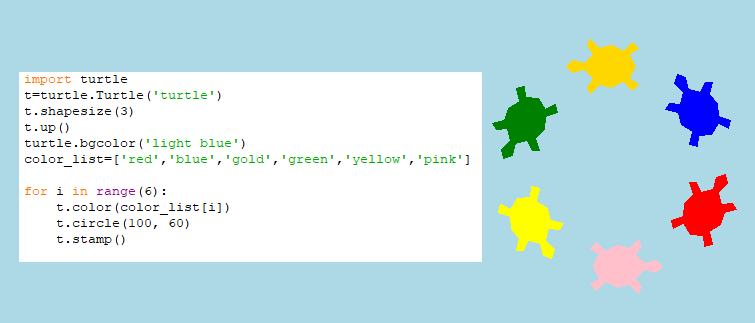
Here we set up the colour of turtle with a line t.color(color\_list[i]), where i is an integer number that takes sequentially the following values 0,1,2,3,4,5.

1. **Example #10** (Example shows the combination of number and string lists)



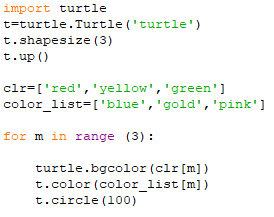


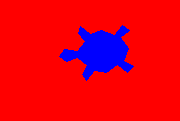
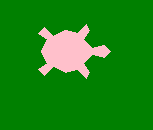
1. **Example #11** (Colour turtles with colour background)



Line turtle.bgcolor('light blue') creates background light blue colour.

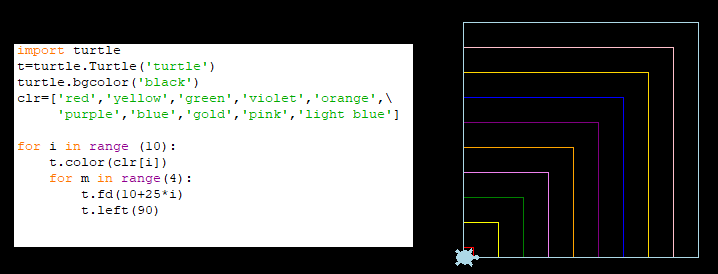
1. **Example #12** (Colour turtles with changing colour background)

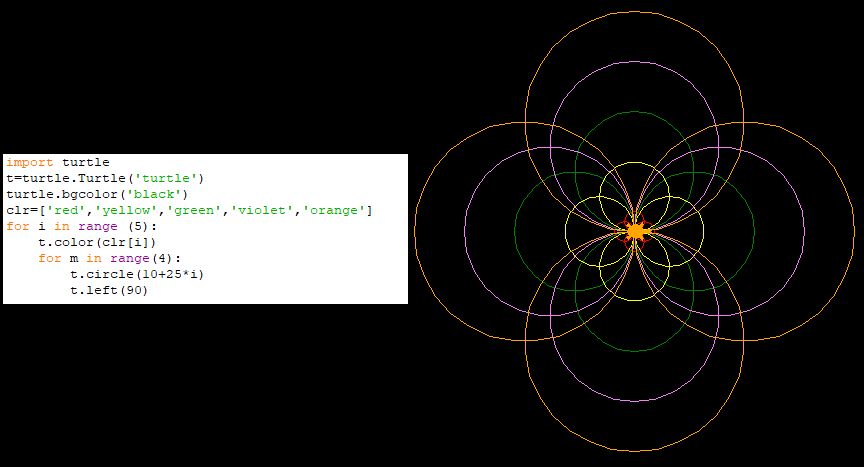


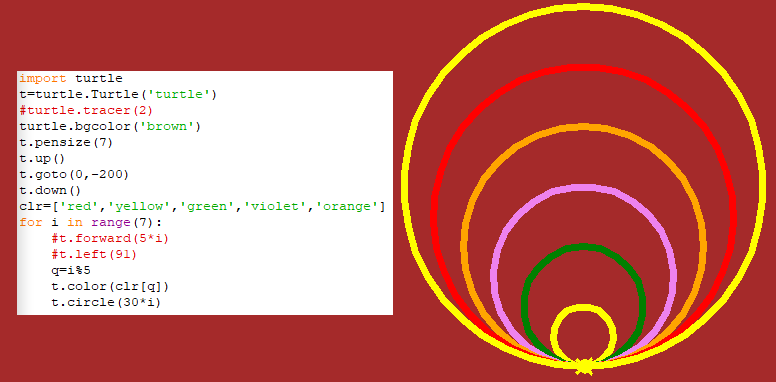
1. **Example #13** (Drawing with colour background)

We can set up black back-ground colour with the line turtle.bgcolor(‘black’)

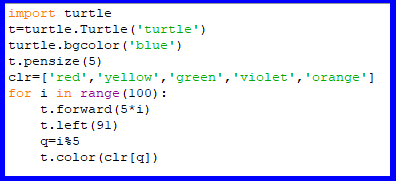




You can use any colour, you want to use with Python code. For, example, code shown below uses brown colour background



**Code** (Blue background)



Variable q=%5 (line nine in the code) tells Python that we use the only 5 colours in the colour list, numbered 0 to 4. In this case, our colour list has five colours, so, we’ll rotate through these five colours over and over.

**Result**

