

Math and Python Turtles

Textbook

Math and Python Turtles



In programming, you can do math with numbers by adding, subtracting, multiplying, and dividing them. Let's focus on these four basic operations for now. The examples below will demonstrate how to use math with variables.

1. Addition: `my_variable = a + b`.
2. Subtraction: `my_variable = a - b`.
3. Multiplication: `my_variable = a * b`.
4. Division: `my_variable = a / b`.

Examples

Here's an example using multiplication.

```
1 octopus_number = 10
```

```
2 legs = 8
3
4 total_legs = octopus_number * legs
5 print(total_legs)
```

This will print out the number `80` in the console.

Here's an example using addition.

```
1 pies = 5
2 new = 3
3
4 total = pies + new
5 print(total)
```

This will print out the number `8` to the console.

Division

When you divide in Python the resulting number is the float variable type. This is the default behavior. Even if the resulting number is a whole number like `6`, it will print out `6.0`.

Here's an example using division.

```
1 candies = 30
2 people = 5
3
4 candies_per_person = candies / people
5 print(candies_per_person)
```

This will print out the number `6.0` in the console.

To change the variable type, use conversion, which is taught in another lesson.

Problems with Math & Inputs

If you wanted to do math with the string `"14"`, you would need to first change it to an integer.

```
1 candies = "14"
2 print(candies * 3)
```

This will print out `"141414"`. Why is this? It is duplicating the string of `"14"` three times. If we want to do math with a string, we need to change it to an integer.

```
1 candies = "14"
2 print(int(candies) * 3)
```

Since inputs behave as a string, we will run into the same problem

```
1 candies = input("How many candies do you have?")
2 print(candies * 3)
```

To fix this, we change the input to an integer.

Changing a variable type is done with [converting](#).

Order of Operations

There is a certain order to execute math statements. This designated order is called the order of operations.

Take the following arithmetic statement. How do you know what order to perform the calculations?

```
fruits = (3 * 5)4 + 9 - 5(4 - 3)
```

Depending on the order that you do the calculations, you'll get different answers. So what is the correct order?

The designated order is as follows.

1. Parentheses
2. Exponents
3. Multiplication/Division
4. Addition/Subtraction

When we have multiple examples of multiplication or division in the statement, perform them in turn from left to right. The same goes for times when we have multiple examples of addition and subtraction in a statement. Start from the left and do them in order from left to right.

We won't get into the specifics of exponents in this lesson.

So let's visit this equation again.

```
fruits = (3 * 5)4 + 9 - 5(4 - 3)
```

We do what's inside the parentheses first, so it simplifies to this.

```
fruits = 15 * 4 + 9 - 5 * 1
```

Next, since there are not exponents, we do multiplication and division in order from left to right so it simplifies to this.

```
fruits = 60 + 9 - 5
```

Once we are through with all the multiplication and division, we do addition and subtraction in order from left to right. In this problem, we start with addition since it is first, then we do the subtraction.

```
fruits = 64
```

Following the order of operations will help guide you to get the correct answer when using math in Python.

Using Inequalities

Besides doing calculations, math in programming also lets us compare values using inequalities. Inequalities help us determine if one value is greater than, less than, or equal to another.

In Python, we use specific symbols for these comparisons:

- `>` (greater than)
- `<` (less than than)
- `>=` (greater than or equal to)
- `<=` (less than or equal to)
- `==` (equal to – remember to use *two* equals signs for comparison, and one `=` for assigning a value to a variable!)
- `!=` (not equal to)

These comparisons are very important because they allow our programs to **make decisions**. For example, you might write code that says: "if the player's score is greater than 100, then give them an extra life."

Inequality comparisons always result in a `True` or `False` answer, which then guides what the program does next.

Reminder – Ask for feedback to improve your mathematical calculations! This can help improve your efficiency. Adapt procedures to apply them to a new context.

Let's Talk Math! Discussing Our Thinking

Math and computational thinking aren't just about finding the right answer; they're also about *how* you get there. Every person approaches a problem a little differently, and by sharing our thinking, we can all learn and grow!

Your Invitation: After we work on a challenging math or coding problem, you'll have the chance to discuss your thinking with your classmates. Be ready to communicate your own mathematical ideas, vocabulary, and methods clearly. But it's not just about sharing your way! Actively listen and analyze the mathematical thinking of others. How did they approach it? Was it different from yours? Can you compare the efficiency of their method to your own? If you spot an error, think about how you might respectfully suggest a correct solution. Finally, practice justifying your results by explaining your methods and processes step-by-step, building strong arguments based on evidence. This kind of discussion helps everyone understand problems more deeply and discover new, even better ways to solve them!

Checkpoint

Math and Python Turtles

Practice doing math with Python code!

1. Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
2. Create a variable named `first` and assign it to the integer `10`.
3. Create a variable named `second` and assign it to the integer `5`.
4. Create a print statement that prints the variable `first` plus the variable `second`.
5. Create a print statement that prints the variable `first` minus the variable `second`.
6. Create a print statement that prints the variable `first` multiplied by the variable `second`.
7. Create a print statement that prints the variable `first` divided by the variable `second`.

Requirements:

- Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
- Create a variable named `first` and assign it to the integer `10`.
- Create a variable named `second` and assign it to the integer `5`.

- Create a print statement that prints the variable `first` plus the variable `second` .
- Create a print statement that prints the variable `first` minus the variable `second` .
- Create a print statement that prints the variable `first` multiplied by the variable `second` .
- Create a print statement that prints the variable `first` divided by the variable `second` .

Questions (10)

1. Which symbol is used for division in Python?

MULTIPLE CHOICE

Choose the correct answer:

- A. `*`
- B. `/`
- C. `#`
- D. `|`

2. Which symbol is used for multiplication in Python code?

MULTIPLE CHOICE

Choose the correct answer:

- A. `*`
- B. `x`
- C. `#`
- D. `&`

3. What would be the value of `my_variable` if `a = 10` and `b = 5`?

MULTIPLE CHOICE

```
my_variable = a * b
```

Choose the correct answer:

- A. 2
- B. 50
- C. 5
- D. 15

4. What would the value of total be if pies = 12 and new = 3

total = pies + new

Choose the correct answer:

- A. 15
- B. 9
- C. 36
- D. 4

5. In Python, what type of variable does the result of the division operation have by default?

MULTIPLE CHOICE

Choose the correct answer:

- A. Integer
- B. Float
- C. String
- D. Boolean

6. If you have the string "14" and you want to perform math with it, what should you do in Python?

MULTIPLE CHOICE

Choose the correct answer:

- A. Use it directly in mathematical operations.
- B. Convert it to a string.
- C. Convert it to an integer.
- D. Change it to a Boolean.

7. What will be the output of the following code if candies is assigned the value "14"?

MULTIPLE CHOICE

```
print(candies * 3)
```

Choose the correct answer:

- A. "141414"
- B. 42
- C. "1413"
- D. 11

8. In the order of operations, what is the first step to execute when solving the following expression?

MULTIPLE CHOICE

fruits = $4 - 3 + 1 + 9 - 2 + 3 \times 5$

Choose the correct answer:

- A. $4 - 3$
- B. 3×5
- C. $1 + 9$
- D. $9 - 2$

9. What is the final value of fruits in the following expression?

MULTIPLE CHOICE

fruits = $(3 + 2)2 + 8 - 4(3 * 1)$

Choose the correct answer:

- A. 6
- B. 14
- C. 33
- D. 12

10. What would the value of "total" be in the following code?

MULTIPLE CHOICE

total = $5 * 2 / 2 + 1$

Choose the correct answer:

- A. 6
- B. 10
- C. 5
- D. 12

Challenges (5)

1. Making Money

Let's practice some math with python! Imagine you have a deal with a neighbor to walk their dog every day for \$3 a day. You want to know how much money you'll have after a week, a month, and a year. Create a program to calculate it for you.

1. Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
2. Create a variable named `per_day` and assign it to the integer `3`.
3. Create a variable named `per_week` and assign it to the variable `per_day` multiplied by 7.
4. Create a variable named `per_month` and assign it to the variable `per_day` multiplied by 30.
5. Create a variable named `per_year` and assign it to the variable `per_day` multiplied by 365.

Requirements:

- Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
- Create a variable named `per_day` and assign it to the integer `3`.
- Create a variable named `per_week` and assign it to the variable `per_day` multiplied by 7.
- Create a variable named `per_month` and assign it to the variable `per_day` multiplied by 30.
- Create a variable named `per_year` and assign it to the variable `per_day` multiplied by 365.

2. Pinata Party

You are trying to decide how many candies to buy to fill your pinata at a party. You want to know how many candies each person would get, depending on how many candies you buy total.

1. Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
2. Create a variable named `candies` and assign it to an input that asks `How many candies?` Make sure to convert the input to an integer in the same line using `int()`.
3. Create a variable named `people` and assign it to an input that asks `How many people?` Make sure to convert the input to an integer in the same line using `int()`.
4. Create a variable named `candies_per_person` and assign it to a math statement. Use the variables `candies` and `people` to calculate the appropriate value of candies per person.
5. Create a print statement. Inside the print statement, concatenate the variable `candies_per_person` to say the following phrase. Don't forget to convert `candies_per_person` into a string using `str()`.

```
Each person will get 10 candies.
```

Requirements:

- Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
- Create a variable named `candies` and assign it to an input that asks `How many candies?` Make sure to convert the input to an integer in the same line using `int()`.
- Create a variable named `people` and assign it to an input that asks `How many people?` Make sure to convert the input to an integer in the same line using `int()`.
- Create a variable named `candies_per_person` and assign it to a math statement. Use the variables `candies` and `people` to calculate the appropriate value of candies per person.
- Create a print statement. Inside the print statement, concatenate the variable `candies_per_person` to say the following phrase, `Each person will get 10 candies.` Don't forget to convert `candies_per_person` into a string using `str()`.

3. Math and Movement

Practice using math while drawing with your turtle!

1. Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
2. Create a variable named `response` and assign it to an input that asks the user `How far should the turtle go?`
`Enter a number 1-10` . Convert the input to an integer using `int()` .
3. Create a variable named `move` . Assign it to the variable named `response` multiplied by 10.
4. Put the variable named `move` into the `turtle.forward()` command.

Requirements:

- Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
- Create a variable named `response` and assign it to an input that asks the user `How far should the turtle go?`
`Enter a number 1-10` . Convert the input to an integer using `int()` .
- Create a variable named `move` . Assign it to the variable named `response` multiplied by 10.
- Put the variable named `move` into the `turtle.forward()` command.

4. Turtle Race

You have 3 turtles on your screen who are racing! See which one makes it the farthest!

The turtle named sam goes twice as far as the default turtle. The turtle named nia goes three times as far as the default turtle.

1. Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
2. Create 2 extra turtle variables. Name one `sam` and one `nia`.
3. Add at least 3 comments to the code (One for each turtle you design.)
4. Give each turtle a unique `shape()`. (Options are turtle, square, circle, arrow, or classic.)
5. Give each turtle a unique `pencolor()`.
6. Create a variable named `distance`. Assign it to the integer `30`.
7. Move the default `turtle` forward by using the `distance` variable.
8. Move the turtle named `sam` forward twice as far as the `distance` variable. Use the `distance` variable and multiplication.
9. Move the turtle named `nia` forward three times as far as the `distance` variable. Use the `distance` variable and multiplication.

Requirements:

- Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
- Create 2 extra turtle variables. Name one `sam` and one `nia`.
- Add at least 3 comments to the code (One for each turtle you design.)
- Give each turtle (at least three) a unique `shape()`. Options are turtle, square, circle, arrow, or classic.
- Give each turtle (at least three) a unique `pencolor()`.
- Create a variable named `distance`. Assign it to the integer `30`.
- Move the default `turtle` forward by using the `distance` variable.
- Move the turtle named `sam` forward twice as far as the `distance` variable. Use the `distance` variable and multiplication.
- Move the turtle named `nia` forward three times as far as the `distance` variable. Use the `distance` variable and multiplication.

5. Area and Perimeter of a Square

Create a program that asks the user how long a side of a square should be. It will then draw the square using turtles. Then it will print out the perimeter and area of the square.

The perimeter of a square is the total distance around the edge.

The area is the length of a side multiplied by the length of a side.

Note: You will need to convert data types.

1. Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
2. Create a variable named `side` and assign it to an input that asks the user `How long should a side be?` Convert the input to an integer using `int()`.
3. Draw the square. Use the code `turtle.forward()` four times. Put the variable named `side` into the parentheses. Use `turtle.left(90)` four times.
4. Create a variable named `perimeter`. Assign it to the variable named `side` multiplied by 4.
5. Create a variable named `area`. Assign it to the variable named `side` multiplied by `side`.
6. Create a print statement that concatenates the string `The perimeter is:` and the variable named `perimeter`. (You will need to convert the variable named `perimeter` into a string. Do this within the print statement.)
7. Create a print statement that concatenates the string `The area is:` and the variable named `area`. (You will need to convert the variable named `area` into a string. Do this within the print statement.)

Requirements:

- Include the necessary code to start up a Python screen. (Import the library and generate a screen.)
- Create a variable named `side` and assign it to an input that asks the user `How long should a side be?` Convert the input to an integer using `int()`.
- Draw the square. Use the code `turtle.forward()` four times. Put the variable named `side` into the parentheses. Use `turtle.left(90)` four times.
- Create a variable named `perimeter`. Assign it to the variable named `side` multiplied by 4.
- Create a variable named `area`. Assign it to the variable named `side` multiplied by `side`.
- Create a print statement that concatenates the string `The perimeter is:` and the variable named `perimeter`. (You will need to convert the variable named `perimeter` into a string. Do this within the print statement.)
- Create a print statement that concatenates the string `The area is:` and the variable named `area`. (You will need to convert the variable named `area` into a string. Do this within the print statement.)

Answer Keys & Solutions

Checkpoint Solutions

Math and Python Turtles

```
1 import turtle
2 turtle.getscreen()
3
4 first = 10
5 second = 5
6
7 print(first + second)
8 print(first - second)
9 print(first * second)
10 print(first / second)
```

Questions

1. Which symbol is used for division in Python?

MULTIPLE CHOICE

Correct Answer:

- A. * ✗ Incorrect
- B. / ✓ Correct
- C. # ✗ Incorrect
- D. | ✗ Incorrect

Explanation:

It's called a forward slash

2. Which symbol is used for multiplication in Python code?

MULTIPLE CHOICE

Correct Answer:

- A. * ✓ Correct
- B. x ✗ Incorrect
- C. # ✗ Incorrect

D. &

✗ Incorrect

Explanation:

It's called an asterisk

3. What would be the value of my_variable if a = 10 and b = 5?

MULTIPLE CHOICE

Correct Answer:

A. 2

✗ Incorrect

B. 50

✓ Correct

C. 5

✗ Incorrect

D. 15

✗ Incorrect

Explanation:

This symbol * means multiplication

4. What would the value of total be if pies = 12 and new = 3

MULTIPLE CHOICE

Correct Answer:

A. 15

✓ Correct

B. 9

✗ Incorrect

C. 36

✗ Incorrect

D. 4

✗ Incorrect

Explanation:

This is an addition example

5. In Python, what type of variable does the result of the division operation have by default?

MULTIPLE CHOICE

Correct Answer:

A. Integer

✗ Incorrect

B. Float

✓ Correct

C. String

✗ Incorrect

D. Boolean

✗ Incorrect

Explanation:

Dividing a value always results in a decimal value.

6. If you have the string "14" and you want to perform math with it, what should you do in Python?

MULTIPLE CHOICE

Correct Answer:

A. Use it directly in mathematical operations.

✗ Incorrect

B. Convert it to a string.

✗ Incorrect

C. Convert it to an integer.

✓ Correct

D. Change it to a Boolean.

✗ Incorrect

Explanation:

Math is done with integers and floats.

7. What will be the output of the following code if candies is assigned the value "14"?

MULTIPLE CHOICE

Correct Answer:

A. "141414"

✓ Correct

B. 42

✗ Incorrect

C. "1413"

✗ Incorrect

D. 11

✗ Incorrect

Explanation:

The string with duplicate.

8. In the order of operations, what is the first step to execute when solving the following expression?

MULTIPLE CHOICE

Correct Answer:

- | | |
|-----------------|-------------|
| A. $4 - 3$ | ✗ Incorrect |
| B. 3×5 | ✓ Correct |
| C. $1 + 9$ | ✗ Incorrect |
| D. $9 - 2$ | ✗ Incorrect |

Explanation:

Parentheses, Exponents, Multiplication, Division, Addition, Subtraction

9. What is the final value of fruits in the following expression?

MULTIPLE CHOICE

Correct Answer:

- | | |
|-------|-------------|
| A. 6 | ✓ Correct |
| B. 14 | ✗ Incorrect |
| C. 33 | ✗ Incorrect |
| D. 12 | ✗ Incorrect |

Explanation:

Parentheses, Exponents, Multiplication, Division, Addition, Subtraction

10. What would the value of "total" be in the following code?

MULTIPLE CHOICE

Correct Answer:

- | | |
|-------|-------------|
| A. 6 | ✓ Correct |
| B. 10 | ✗ Incorrect |

C. 5

✖ Incorrect

D. 12

✖ Incorrect

Explanation:

Parentheses, Exponents, Multiplication, Division, Addition, Subtraction

Challenges

1. Making Money

Solution:

```
1 import turtle
2 turtle.getscreen()
3
4 per_day = 3
5 per_week = per_day * 7
6 per_month = per_day * 30
7 per_year = per_day * 365
8
9 print("I will make " + per_week " dollars a week.")
10
11 print("I will make " + per_month " dollars a month.")
12
13 print("I will make " + per_year " dollars a year.")
```

2. Pinata Party

Solution:

```
1 import turtle
2 turtle.getscreen()
3
4 candies = int(input("How many candies?"))
5
6 people = int(input("How many people?"))
7
8 candies_per_person = candies / people
9
10 print("Each person will get " + str(candies_per_person) + " candies.")
```

3. Math and Movement

Solution:

```
1 import turtle
```



```
2 turtle.getscreen()
3
4 response = int(input("How far should the turtle go? Enter a number 1-10"))
5
6 move = response * 10
7
8 turtle.forward(move)
```

4. Turtle Race

Solution:

```
1 import turtle
2 turtle.getscreen()
3
4 #default turtle
5 turtle.shape("classic")
6 turtle.pencolor("yellow")
7
8 #sam the turtle
9 sam = turtle.Turtle()
10 sam.shape("turtle")
11 sam.pencolor("green")
12
13 #nia the turtle
14 nia = turtle.Turtle()
15 nia.shape("circle")
16 nia.pencolor("blue")
17
18
19 distance = 30
20 turtle.forward(distance)
21 sam.forward(distance*2)
22 nia.forward(distance*3)
```

5. Area and Perimeter of a Square

Solution:

```
1 import turtle
2 turtle.getscreen()
3
4 side = int(input("How long should a side be?"))
5
6 turtle.forward(side)
7 turtle.left(90)
8 turtle.forward(side)
9 turtle.left(90)
10 turtle.forward(side)
11 turtle.left(90)
12 turtle.forward(side)
13 turtle.left(90)
14
15 perimeter = side * 4
16 area = side * side
```

```
17  
18 print("The perimeter is: " + str(perimeter))  
19 print("The area is: " + str(area))
```