

Converting

Textbook

Converting Data Types



Foundation

There will be times when you want to make sure that your variable is a certain data type.

`"23"` is a **string**

`23` is an **integer**

`23.0` is a **float**

If you tried to complete the command `"23" * 4`, it would print out `"23232323"`. It is duplicating the string of `"23"` four times. If you wanted to do math with the string `"23"`, you would need to first change it to an integer.

(We will talk more about math in an upcoming lesson. For now, learn how to convert between data types and it will help you during the math lesson.)

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

Converting to an Integer

Converting is done by specifying the variable type when assigning a value to the variable.

You can change the `"3"` from a string to an integer using the `int()` function.

```
int()
```

```
1 cheese = "3"  
2 cheese = int("3")  
3 print(cheese)
```

Try it!

This will change the string `"3"` to the integer `3`.

You can change the `2.8` from a float to an integer using the `int()` function.

```
1 peppers = 2.8  
2 peppers = int(2.8)  
3 print(peppers)
```

Try it!

This will change the float `2.8` to the integer `2`. This is done by dropping the decimal and any numbers that follow it.

To be clear: when changing a float to an integer, it **DOES NOT ROUND DOWN**. It **drops the decimal**, which is not exactly the same thing.

Converting to a String

You can also change data into a string using the following function.

```
str()
```

```
1 fries = 30  
2 fries = str(30)  
3 print(fries)
```

Try it!

Converting to a Float

You can also change data into a float using the following function.

```
float()
```

```
1 pizza = 4  
2 pizza = float(4)
```

```
3 print(pizza)
```

Try it!

Converting Inputs

Remember the lesson on inputs/outputs? This is very important. **Inputs are automatically accepted as a string.** Take this code for example:

```
family = input("How many family members do you have?")
```

If the user answers with the number 5, the computer accepts it as a **string** with the number 5 in it. Like this: `"5"`.

This becomes very important when you need to use that number to do math. The following code will throw an error.

```
1 family = input("How many family members do you have?")
2 baby = family + 1
3 print(baby)
```

Try it!

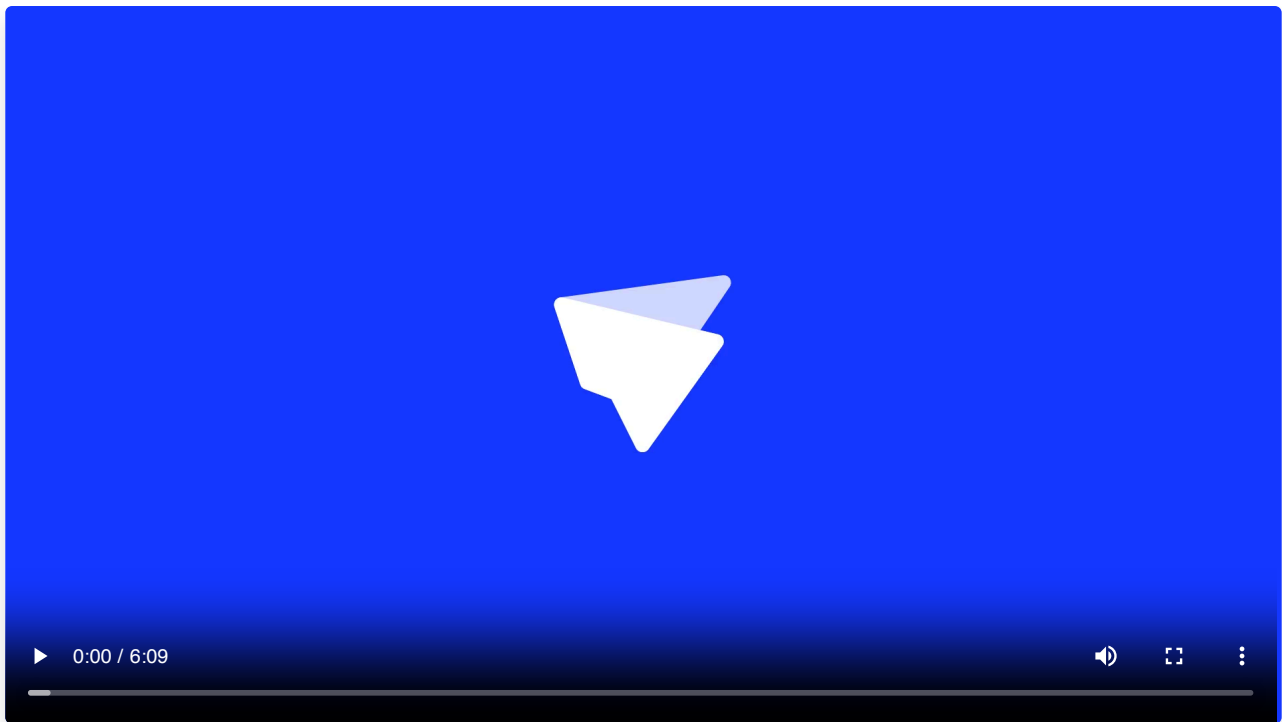
You will need to change your input into an [integer](#) before you can do math.

```
1 family = int(input("How many family members do you have?"))
2 baby = family + 1
3 print(baby)
```

Try it!

To be very clear: Inputs are automatically accepted as a [string](#)! To do math with an integer input, it will need to be converted into an integer.

We'll talk more about how to do math with Python in upcoming lessons



Checkpoint

Converting

Create three variables:

1. One **integer** variable.
2. One **float** variable.
3. One **string** variable that is a number (e.g. "23").
4. Create a new variable and assign it to the float variable you created earlier converted to an integer. Then print the converted variable in a separate print statement.
5. Create a new variable and assign it to the string variable you created earlier converted to an integer. Then print the converted variable in a separate print statement.
6. Create a new variable and assign it to the integer you created earlier converted to a float. Then print the converted variable in a separate print statement.

For example, your code could look something like this:

```
watermelons = 0.5  
  
converted_float = int(watermelons)  
  
print(converted_float)
```

As a reminder, there are often several ways to write code that are all correct. For this particular checkpoint, the autograder is looking for the above approach.

Requirements:

- Create one integer variable.
- Create one float variable.
- Create one string variable with a number value.
- Create a new variable and assign it to the float variable converted to an integer. Then print the converted variable separately.
- Create a new variable and assign it to the string variable converted to an integer. Then print the converted variable separately.
- Create a new variable and assign it to the integer variable converted to a float. Then print the converted variable separately.

Questions (10)

1. What happens when you change a float to an integer?

MULTIPLE CHOICE

Choose the correct answer:

- A. It rounds the decimal up.
- B. It rounds the decimal down.
- C. It drops the decimal.
- D. It moves the decimal to the right one space.

2. What data type is this? "7.45"

MULTIPLE CHOICE

Choose the correct answer:

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

3. What will print out with the following code? `print("4" * 3)`

MULTIPLE CHOICE

Choose the correct answer:

- A. 12
- B. 444
- C. 7
- D. 1

4. What data type are inputs automatically accepted as?

MULTIPLE CHOICE

Choose the correct answer:

- A. string
- B. integer
- C. boolean
- D. float

5. Debug the following code. Assume you are converting the variable named pumpkins to an integer.

DEBUG CODE

Code to Debug:

```
1 pumpkins = int"3"
```

6. Debug the following code. Assume you are changing the variable named price to an integer.

DEBUG CODE

Code to Debug:

```
1 price = integer:6.89
```

7. Debug the following code. Assume you are changing the variable named candy to a string.

DEBUG CODE

Code to Debug:

```
1 candy = string(5)
```

8. Debug the following code. Assume you are changing the variable named rating to a float.

DEBUG CODE

Code to Debug:

```
1 rating = fl(7)
```

9. What would the following value turn into if it were converted to an integer? 5.68

MULTIPLE CHOICE

Choose the correct answer:

- A. 5.7
- B. 6
- C. 5
- D. 5.60

10. What data type is this? 3.0

Choose the correct answer:

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

Challenges (3)

1. Age in 18 years

Write a program that asks the user to input their age. **Convert that input to an integer**, and print how old they will be in 18 years.

First example:

Input: 14

Output: 32

Second example:

Input: 10

Output: 28

Hint: We haven't covered math yet, but in Python you can create a variable and assign it to a math statement.

For example:

```
height = 10 + 3
```

```
print(height)
```

This will print out 13

READ THIS! Important information about the autograder!

As you know, the autograder is very detail oriented. If the challenge or checkpoint asks for a specific **input** and a specific **output**, the *method of grading is a bit different*. Do you see how there's not requirements? The autograder is checking to see that your program will present the correct output with certain inputs.

Follow these tips and tricks to make sure your correct code can pass the autograder.

- Make sure you have only one print statement! The auto grader is going to be looking at all the print statements on your code page, so make sure you just have one to represent your answer!

- The autograder will also be checking for the required answer EXACTLY. Make sure to match capitalization, spacing, and punctuation. If you miss a period or a space, it won't pass. This helps to show just how detail oriented coding is.

- Keep in mind that just because the example input is generating the correct output, it doesn't mean your code works for other inputs. Make sure a variety of inputs follow the requirements of the challenge or checkpoint in order to pass.

2. Decimal Dilemma

Write a program that gets a number with a decimal from the user as input.

Convert this input to a **float**, then convert it to an **integer** and print the result.

This should remove the decimal place.

Example 1:

Input: 5.7

Output: 5

Example 2:

Input: 4.3

Output: 4

Another example, if the input is 3.4 your program should print 3.

READ THIS! Important information about the autograder!

As you know, the autograder is very detail oriented. If the challenge or checkpoint asks for a specific **input** and a specific **output**, the method of grading is a bit different. The autograder is checking to see that your program will present the correct output with certain inputs.

Follow these tips and tricks to make sure your correct code can pass the autograder.

- Make sure you have only one print statement! The auto grader is going to be looking at all the print statements on your code page, so make sure you just have one to represent your answer!

- The autograder will also be checking for the required answer EXACTLY. Make sure to match capitalization, spacing, and punctuation. If you miss a period or a space, it won't pass. This helps to show just how detail oriented coding is.

- Keep in mind that just because the example input is generating the correct output, it doesn't mean your code works for other inputs. Make sure a variety of inputs follow the requirements of the challenge or checkpoint in order to pass.

3. Name Repeater

Write a program that takes in **two inputs** – the user's name and the user's age. Convert the age input into an **integer**, and multiply the user's name by the number that was inputted for age.

Note: The name input must come before the age input.

Print the multiplied name.

For example:

Inputs: `George` and `4`

Output: `GeorgeGeorgeGeorgeGeorge`

Another example:

Inputs: `Tanner` and `2`

Output: `TannerTanner`

Hint: We haven't covered math yet in this course, but you can do arithmetic in Python.

For example:

```
apples = 3  
  
total = apples * 8  
  
print(total)
```

Play around with multiplying strings and see what happens!

```
dog = "dalmation"  
  
print(dog * 4)
```

READ THIS! Important information about the autograder!

As you know, the autograder is very detail oriented. If the challenge or checkpoint asks for a specific **input** and a specific **output**, the method of grading is a bit different. The autograder is checking to see that your program will present the correct output with certain inputs.

- Make sure you have only one print statement! The auto grader is going to be looking at all the print statements on your code page, so make sure you just have one to represent your answer!

- The autograder will also be checking for the required answer EXACTLY. Make sure to match capitalization, spacing, and punctuation. If you miss a period or a space, it won't pass. This helps to show just how detail oriented coding is.

- Keep in mind that just because the example input is generating the correct output, it doesn't mean your code works for other inputs. Make sure a variety of inputs follow the requirements of the challenge or checkpoint in order to pass.

Answer Keys & Solutions

Checkpoint Solutions

Converting

```
1 apples = 5
2 watermelons = 0.5
3 pears = "13"
4
5 converted_float = int(watermelons)
6 converted_string = int(pears)
7 converted_integer = float(apples)
8
9 print(converted_float)
10 print(converted_string)
11 print(converted_integer)
```

Questions

1. What happens when you change a float to an integer?

MULTIPLE CHOICE

Correct Answer:

- A. It rounds the decimal up. ✗ Incorrect
- B. It rounds the decimal down. ✗ Incorrect
- C. It drops the decimal. ✓ Correct
- D. It moves the decimal to the right one space. ✗ Incorrect

Explanation:

Floats have decimals while integers do not.

2. What data type is this? "7.45"

MULTIPLE CHOICE

Correct Answer:

- A. String ✓ Correct
- B. Integer ✗ Incorrect

C. Float

✗ Incorrect

Explanation:

Strings have quotation marks around them. Sometimes strings can look like numbers.

3. What will print out with the following code? `print("4" * 3)`

MULTIPLE CHOICE

Correct Answer:

A. 12

✗ Incorrect

B. 444

✓ Correct

C. 7

✗ Incorrect

D. 1

✗ Incorrect

Explanation:

Remember that "4" is a string, not a number. So the string will duplicate when it's multiplied.

4. What data type are inputs automatically accepted as?

MULTIPLE CHOICE

Correct Answer:

A. string

✓ Correct

B. integer

✗ Incorrect

C. boolean

✗ Incorrect

D. float

✗ Incorrect

Explanation:

Inputs are automatically accepted as this data type " "

5. Debug the following code. Assume you are converting the variable named pumpkins to an integer.

DEBUG CODE

Incorrect Code:

```
1 pumpkins = int"3"
```

Correct Solution:

```
1 pumpkins = int("3")
```

Explanation:

Make sure to include parentheses. The input of your conversion should still be a string.

6. Debug the following code. Assume you are changing the variable named price to an integer.

DEBUG CODE

Incorrect Code:

```
1 price = integer:6.89
```

Correct Solution:

```
1 price = int(6.89)
```

Explanation:

You need parentheses on your conversion. int()

7. Debug the following code. Assume you are changing the variable named candy to a string.

DEBUG CODE

Incorrect Code:

```
1 candy = string(5)
```

Correct Solution:

```
1 candy = str(5)
```

Explanation:

str()

8. Debug the following code. Assume you are changing the variable named rating to a float.

DEBUG CODE

Incorrect Code:

```
1 rating = fl(7)
```

Correct Solution:

```
1 rating = float(7)
```

Explanation:

`float()`

9. What would the following value turn into if it were converted to an integer? 5.68

MULTIPLE CHOICE

Correct Answer:

- A. 5.7 ✗ Incorrect
- B. 6 ✗ Incorrect
- C. 5 ✓ Correct
- D. 5.60 ✗ Incorrect

Explanation:

When converting a float to an integer, it drops the decimal.

10. What data type is this? 3.0

MULTIPLE CHOICE

Correct Answer:

- A. String ✗ Incorrect
- B. Integer ✗ Incorrect
- C. Float ✓ Correct
- D. Boolean ✗ Incorrect

Explanation:

Anything with a decimal is not an integer.

Challenges

1. Age in 18 years

Solution:

```
1 age = int(input("How old are you?"))
2
3 print(age+18)
```

2. Decimal Dilemma

Solution:

```
1 month = float(input("How many apples did you eat? Input as a decimal. "))
2
3 whole_number = int(month)
4
5 print(whole_number)
```

3. Name Repeater

Solution:

```
1 name = input("What is your name?")
2 age = input("What is your age?")
3
4 repeat_number = int(age)
5 print(repeat_number * name)
```