

Variables

Textbook

Variables



Foundation

A [variable](#) is a building block of programming that allows you to store data. Think of it like a "box" that can store anything you put in it.

1. You can name "the box", or variable anything you want and you can put any type of data inside. You'll learn more about different data types in a later section. **The name of the variable must either be one word, or connected with underscores.** `pet = "dog"` or `my_pet = "dog"`
2. You do not need to specify anything before declaring a variable, unlike other programming languages.
3. You use the equal sign `=` to assign the data to the variable or "put it in the box."



Abstraction

Using variables is a simple way to refer to complicated information. Sometimes variables can hold lots of information. Rather than type out the entire code every time, we can just refer to the variable name.

Using simple symbols to represent more complicated information is called abstraction.

Abstraction is commonly used in our everyday life. A red light represents "stop the vehicle before proceeding forward." Pressing "call" on your phone is really telling it to enter a number and connect to your friend's phone. It's easier to use abstraction to simplify things.

Creating Variables

The example below includes a variable named `friend` that has the word `"Fred"` stored inside it. It also includes a variable named `age` that has the number `13` stored inside it.

```
1 friend = "Fred"
2 age = 13
3
4 print(friend)
5 print(age)
6
```

Try it!

Creating variables is also known as declaring variables.

Variables can only be one word long. We can get around this rule by separating words with underscores.

```
1 my_friend = "Jasmine"
2 her_age = 12
3
4 print(my_friend)
5 print(her_age)
```

Try it!

When you want to store a word or phrase inside a variable you need to put quotes around it. Numbers, on the other hand, don't need any extra punctuation.

- This will be explained in greater detail in later sections.

```
1 my_name = "My name is Fred"
2 number = 13
3
4 print(my_name)
5 print(number)
```

Try it!

Naming Variables

When deciding what to name your variables, try to name them something that refers to what it's doing. It is [best practice](#) to name your variables something that refers to what the variable will hold.

Here are a few more rules when it comes to naming variables.

- Variables cannot be more than one word long (We get around this rule by adding an underscore character `_` to join multiple words together. For example `first_name` is a valid variable name.)
- A variable name must start with a letter or an underscore character `_`
- A variable name **cannot** start with a number
- A variable name can only use capital A-Z, lowercase a-z, numbers 0-9, and the underscore character `_`. No special characters are allowed.
- Variables **cannot** be named a word that is reserved for uses in Python. The list below shows words that cannot be used for variable names.

Words that cannot be used for variable names.

'False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield'

Reassigning Variables

Once a variable has been assigned to a value, it can be reassigned to a new value in the same program.

```
1 age = 14
2
3 print(age)
4
5 age = 18
6
7 print(age)
```

Try it!

Outputting Variables

After creating these variables in your python code, you can use the print function we used earlier to [output](#) the variables. The code below will print out the name and number that we stored in the name and number variables.

Printing a variable is a way to output a variable.

```
1 print(name)
2
3 print(number)
```

Deleting Variables

The easiest way to delete a variable is to just remove it from the code. There is also a code snippet that can delete a variable as well. This is done using `del` .

```
1 name = "Fred"
2 del name
3
4 print(name)
```

Try it!

Now, if you try to print the variable named `name` it will throw an error because it was deleted.

Initializing vs Declaring Variables

Declaring a variable is when you create a new variable, often without assigning a value (though in Python, this happens when you assign a value).

Initializing a variable is when you give the variable its initial value.

(For Python this happens at the same time: `name = "Sally"`)

VARIABLES

Python



Skill
Struck TM

▶ 0:00 / 1:01



Checkpoint

Variables

In this checkpoint, you will be creating and printing out variables.

1. Create **three** variables.
2. Print out each of these on separate lines of code.

Requirements:

- Three variables

Questions (10)

1. What is a building block of code that allows you to store data in it like a box?

MULTIPLE CHOICE

Choose the correct answer:

- A. attribute
- B. output
- C. variable
- D. name

2. True or False: Creating a variable is the same as declaring a variable.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

3. What is the best practice when creating variables?

MULTIPLE CHOICE

Choose the correct answer:

- A. Use letters like x, y, or n to make code as short as possible.
- B. Use more than one word when naming a variable to make it specific.
- C. Use one syllable words when creating variables.
- D. Name your variable something that refers to what it holds.

4. Edit the text box below to debug (fix) the code:

DEBUG CODE

Code to Debug:

```
1 class_pet : "snake"
```

5. Edit the text box below to debug (fix) the code:

DEBUG CODE

Code to Debug:

```
1 friend = "Marie"
```

6. Edit the text box below to debug (fix) the code:

DEBUG CODE

Code to Debug:

```
1 age : 14
```

7. "Using simple symbols to represent more complicated information" is a definition for what?

MULTIPLE CHOICE

Choose the correct answer:

- A. abstraction
- B. variable
- C. box
- D. syntax

8. Which of the following are rules for naming variables in Python? Select all that apply

SELECT MULTIPLE

Select all that apply:

- A. A variable name must start with a letter or an underscore character _
- B. A variable name cannot start with a number
- C. A variable name can only use capital A-Z, lowercase a-z, numbers 0-9, and the underscore character _ . No special characters are allowed.
- D. A variable name can't have two or more words separated with a space.
- E. Variables must be 10 characters or less
- F. Variables must have a number in them

9. True or False: Once a variable has been assigned to a value, it can not be reassigned to a new value in the same program.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

10. True or False: Printing a variable is a way to output a variable.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

Challenges (5)

1. Wood Chuck

Create a variable named `how_much_wood_would_a_wood_chuck_chuck_if_a_wood_chuck_could_chuck_wood`

Assign it to your best guess to what the answer is.

Print the variable.

Requirements:

- Create the variable.
- Print the variable.

2. Restaurant Favorites

In the last lesson you wrote three print statements that printed out your favorite food. Imagine you are at your very favorite restaurant. You walk through the doors and can smell the delicious food.

Create **three** variables named `main_course`, `side`, and `drink` .

Assign each variable to your favorite menu item associated with that variable.

Print out each of these variables in a **separate print statement**.

Requirements:

- Create a variable named `main_course` and assign it to what you want to order.
- Create a variable named `side` and assign it to what you want to order.
- Create a variable named `drink` and assign it what you want to order.
- Print out each of the variables.

3. How many in a beetle?

The Volkswagen beetle is an iconic car. Imagine if you were to fill one up with different objects.

How many basketballs could you fit inside?

How many people could you fit inside?

How many hamburgers could you fit inside?

Create **three variables** that have your number guess in them. Print out all three numbers in **separate** print statements.

Requirements:

- Create 3 variables
- Assign numbers to the variables
- Print the variables

4. Who is your hero?

Who are your heroes? A hero can be someone real or fictional, living or dead. Who inspires you?

Create **3 variables**: `hero1` , `hero2` , and `hero3` .

Assign a name to each variable.

Print each variable in a **separate** print statement.

Requirements:

- Create 3 Variables
- Assign a name to each variable
- Print each variable

5. Riddle Variables

Create several variables with the following names:

```
what_goes_up_when_rain_comes_down
```

```
what_is_full_of_holes_but_holds_water
```

```
what_can_you_not_keep_until_you_give_it
```

Assign the variables to your guesses!

Print the three variables in separate print statements.

Requirements:

- Create a variable named `what_goes_up_when_rain_comes_down` and print it.
- Create a variable named `what_is_full_of_holes_but_holds_water` and print it.
- Create a variable named `what_can_you_not_keep_until_you_give_it` and print it.

Answer Keys & Solutions

Checkpoint Solutions

Variables

```
1 name = "Chelsea"
2 age = 15
3 favorite_color = "red"
4
5 print(name)
6 print(age)
7 print(favorite_color)
```

Questions

1. What is a building block of code that allows you to store data in it like a box?

MULTIPLE CHOICE

Correct Answer:

- A. attribute ✗ Incorrect
- B. output ✗ Incorrect
- C. variable ✓ Correct
- D. name ✗ Incorrect

Explanation:

Variables can hold lots of kinds of information--kind of like a box.

2. True or False: Creating a variable is the same as declaring a variable.

MULTIPLE CHOICE

Correct Answer:

- A. True ✓ Correct
- B. False ✗ Incorrect

Explanation:

Programmers use the word "create" and "declare" to mean the same thing.

3. What is the best practice when creating variables?

MULTIPLE CHOICE

Correct Answer:

- A. Use letters like x, y, or n to make code as short as possible. ✗ Incorrect
- B. Use more than one word when naming a variable to make it specific. ✗ Incorrect
- C. Use one syllable words when creating variables. ✗ Incorrect
- D. Name your variable something that refers to what it holds. ✓ Correct

Explanation:

Programs have many variables--so it's important to name them something that makes sense in the program.

4. Edit the text box below to debug (fix) the code:

DEBUG CODE

Incorrect Code:

```
1 class_pet : "snake"
```

Correct Solution:

```
1 class_pet = "snake"
```

Explanation:

Variables are assigned with an equals sign, not a colon.

5. Edit the text box below to debug (fix) the code:

DEBUG CODE

Incorrect Code:

```
1 friend = "Marie
```

Correct Solution:

```
1 friend = "Marie"
```

Explanation:

There is a missing quotation mark "

6. Edit the text box below to debug (fix) the code:

DEBUG CODE

Incorrect Code:

```
1 age : 14
```

Correct Solution:

```
1 age = 14
```

Explanation:

Python doesn't need a semicolon at the end. Numbers don't need quotation marks.

7. "Using simple symbols to represent more complicated information" is a definition for what?

MULTIPLE CHOICE

Correct Answer:

A. abstraction

✓ Correct

B. variable

✗ Incorrect

C. box

✗ Incorrect

D. syntax

✗ Incorrect

Explanation:

Another definition is "simplifying complex topics by using variables."

8. Which of the following are rules for naming variables in Python? Select all that apply

SELECT MULTIPLE

Correct Answers:

A. A variable name must start with a letter or an underscore character _

✓ Correct

B. A variable name cannot start with a number

✓ Correct

- C. A variable name can only use capital A-Z, lowercase a-z, numbers 0-9, and the underscore character `_`. No special characters are allowed. ✓ Correct
- D. A variable name can't have two or more words separated with a space. ✓ Correct
- E. Variables must be 10 characters or less ✗ Incorrect
- F. Variables must have a number in them ✗ Incorrect

Explanation:

Variable names can be any length. Variable names do not need to have numbers in them.

9. True or False: Once a variable has been assigned to a value, it can not be reassigned to a new value in the same program.

MULTIPLE CHOICE

Correct Answer:

- A. True ✗ Incorrect
- B. False ✓ Correct

Explanation:

Variables often get reassigned values

10. True or False: Printing a variable is a way to output a variable.

MULTIPLE CHOICE

Correct Answer:

- A. True ✓ Correct
- B. False ✗ Incorrect

Explanation:

Outputs are ways to see or use variables

Challenges

1. Wood Chuck

Solution:

```
1 how_much_wood_would_a_wood_chuck_chuck_if_a_wood_chuck_could_chuck_wood = 34
2
3 print(how_much_wood_would_a_wood_chuck_chuck_if_a_wood_chuck_could_chuck_wood )
```

2. Restaurant Favorites

Solution:

```
1 main_course = "Pasta"
2 side = "soup"
3 drink = "Sprite"
4
5 print(main_course)
6 print(side)
7 print(drink)
```

3. How many in a beetle?

Solution:

```
1 basketballs = 30
2 people = 7
3 hamburgers = 200
4
5 print(basketballs)
6 print(people)
7 print(hamburgers)
```

4. Who is your hero?

Solution:

```
1 hero1 = "Batman"
2 hero2 = "Mr Rogers"
3 hero3 = "My dad"
4
5 print(hero1)
6 print(hero2)
7 print(hero3)
```

5. Riddle Variables

Solution:

```
1 what_goes_up_when_rain_comes_down = "umbrella"
2 what_is_full_of_holes_but_holds_water = "sponge"
```

```
3 what_can_you_not_keep_until_you_give_it = "promise"  
4  
5 print(what_goes_up_when_rain_comes_down)  
6 print(what_is_full_of_holes_but_holds_water)  
7 print(what_can_you_not_keep_until_you_give_it)
```