

Buttons and If Statements

Textbook

Buttons and If Statements



Add a Forever Function

Now let's learn to use the **A and B buttons**. If you look at the face of the micro:bit, you can see two buttons labeled **A** and **B**. Let's make something happen if we press these buttons!

For this to work, we need to put the code inside a forever block, so that it will work whenever we press the button, not just one time.

```
1 def on_forever():  
2  
3  
4 basic.forever(on_forever)  
5
```

Add an If Statement for the A Button

Now let's add an [if statement](#) that checks to see if the A button is pressed. Use the following code.

```
1 def on_forever():
2     if input.button_is_pressed(Button.A):
3         basic.show_icon(IconNames.HAPPY)
4 basic.forever(on_forever)
5
```

INDENTATION MATTERS!!!

Notice that the code you want to run IF the button is pressed needs to be indented with the TAB key. You can add as many lines of code to your IF statement that you'd like!

```
1 def on_forever():
2     if input.button_is_pressed(Button.A):
3         basic.show_icon(IconNames.HAPPY)
4         basic.show_icon(IconNames.HEART)
5 basic.forever(on_forever)
6
```

Add an If Statement for the B Button

To change it to the B button, just update the Button.A code to Button.B

```
1 def on_forever():
2     if input.button_is_pressed(Button.B):
3         basic.show_icon(IconNames.HAPPY)
4         basic.show_icon(IconNames.HEART)
5 basic.forever(on_forever)
6
```

Indentation Matters!

Don't forget to correctly indent your functions and your if statements! Use consistent tabs. This is a feature of Python that helps keep your code clean and run smoothly. Match your indentation with the examples in this lesson.

The indent means that the code is "inside" whatever is above it. So in the above example, the if statement is inside the function. The basic commands are inside the if statement.

Try adding some if statements to your code!

Adopted from microbit.org platform

Critical Thinking Questions

- Imagine you're programming a vending machine. How would you use "if statements" to make sure the machine gives you the right drink only if you put in enough money and if you press the correct button

for that drink?

- Indentation matters in programming. How is following precise formatting rules in coding, like indenting lines, similar to or different from following specific rules when writing an essay or a musical score? What happens if you don't follow these rules in either case?
- Think about an app on your phone that responds when you tap different parts of the screen. How does the concept of a program constantly checking "if" a button (or screen area) "is pressed" allow these apps to be interactive and useful? Provide an example.

Questions (10)

1. A student wants to make the micro:bit show a heart when button A is pressed. They write this code but nothing happens when they press the button. What did they forget to include?

MULTIPLE CHOICE

```
if input.button_is_pressed(Button.A): basic.show_icon(IconNames.HEART)
```

Choose the correct answer:

- A. Quotation marks around the button name
- B. A forever function to keep checking for button presses
- C. A pause command
- D. The `basic.forever()` function call

2. You want to make a simple game where button A shows your score and button B clears the screen. How many if statements will you need inside your forever function?

MULTIPLE CHOICE

Choose the correct answer:

- A. 1 if statement
- B. 2 if statements
- C. 3 if statements
- D. 4 if statements

3. Consider the following code. What will happen when they press button A once?

```
def on_forever(): if input.button_is_pressed(Button.A): basic.show_number(1) basic.show_number(2) basic.show_number(3)
basic.forever(on_forever)
```

Choose the correct answer:

- A. Only the number 1 will show
- B. Only the number 3 will show
- C. The numbers 1, 2, and 3 will show in sequence
- D. All three numbers will show at the same time

4. A student writes this code but gets an indentation error. What needs to be fixed?

```
def on_forever(): if input.button_is_pressed(Button.A): basic.show_icon(IconNames.HAPPY) basic.forever(on_forever)
```

Choose the correct answer:

- A. The if statement needs more indentation
- B. The basic.show_icon line needs more indentation
- C. The function definition needs less indentation
- D. The basic.forever line needs more indentation

5. You want to create a program where pressing button A shows "YES" and pressing button B shows "NO". Which code structure works best?**Choose the correct answer:**

- A. Put both if statements in separate forever functions
- B. Put the if statements outside any forever function
- C. Use only one if statement that checks both buttons
- D. Put both if statements inside the same forever function

6. You want to create a program where pressing button A shows a happy face, and pressing button B shows a sad face. A friend suggests this code. Will this code work correctly?

MULTIPLE CHOICE

```
def on_forever(): if input.button_is_pressed(Button.A): basic.show_icon(IconNames.HAPPY) if
input.button_is_pressed(Button.B): basic.show_icon(IconNames.SAD) basic.forever(on_forever)
```

Choose the correct answer:

- A. No, you cannot have two if statements in one function
- B. No, the indentation is wrong
- C. Yes, this code will work as intended
- D. No, you need to use Button.A and Button.B together

7. A student wants their program to do three different things when button A is pressed: show a heart, pause for 1 second, then show a star. Where should they put the pause command?

MULTIPLE CHOICE

Choose the correct answer:

- A. Before the if statement
- B. After all the if statements
- C. Inside the if statement, between the heart and star commands
- D. In a separate forever function

8. You notice a classmate's code is not working properly. What indentation error needs to be fixed?

MULTIPLE CHOICE

```
def on_forever(): if input.button_is_pressed(Button.A): basic.show_icon(IconNames.HAPPY) basic.forever(on_forever)
```

Choose the correct answer:

- A. Only the if statement needs to be indented
- B. Only the basic.show_icon needs to be indented
- C. Both the if statement and basic.show_icon need proper indentation
- D. The indentation is correct

9. A student asks why they need to put button checking code inside a forever function instead of just writing it once. What is the best explanation?

MULTIPLE CHOICE

Choose the correct answer:

- A. Forever functions make the code run faster
- B. The micro:bit needs to continuously check if buttons are being pressed
- C. Buttons only work inside forever functions
- D. It makes the code easier to read

10. You want to create a simple counter where each press of button A adds 1 to a number displayed on screen. You start with this code. What is the main limitation of this approach?

MULTIPLE CHOICE

```
def on_forever(): if input.button_is_pressed(Button.A): basic.show_number(1) basic.forever(on_forever)
```

Choose the correct answer:

- A. The indentation is wrong
- B. It will always show 1 instead of counting up
- C. Button A is the wrong button to use
- D. The forever function is unnecessary

Robotics Challenges (5)

1. Display the Letter A

Challenge

Textbook

Display the Letter A

Create a program that will display the letter A if the A button is pressed.

Requirements

Create a forever function

Create the if statement to see if button A is pressed

If the button A is pressed, show the string A

Answer Key

Submit

Download

...

1

def on_forever():

2

pass

3

basic.forever(on_forever)

4

Explorer

▼

2. Count by 2

Challenge

Textbook

Count by 2

Create a program that will count up by 2 if the B button is pressed.

If the button B is pressed, it will display the number 2, then 4, 6, and 8.

Requirements

Create the forever function

Inside the forever function, create an if statement to see if button B is pressed.

If the button B is pressed, display the numbers 2, 4, 6, and 8

Answer Key

Submit

Step 1

Create the forever function.

Count by 2 Step 1 of 3

◀ 1 ▶ Next

Toolbox

Search

1

def on_forever():

2

pass

3

basic.forever(on_forever)

4

Download

...

1

def on_forever():

2

pass

3

basic.forever(on_forever)

4

3. Pressing Both Buttons

Challenge

Textbook

Pressing Both Buttons

Create a program that shows the **LEFT_TRIANGLE** icon if both buttons A and B are pressed.

Here is the code for pressing both buttons.

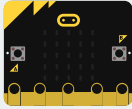
```
input.button_is_pressed(Button.A
B)
```

If the buttons are not pressed, display the **SCISSORS** icon.

Requirements

- Create the forever function.
- Inside the forever function, check to see if button AB is pressed
- If button AB is pressed, display the **LEFT_TRIANGLE** icon
- If the buttons are not pressed display the

1 def on_forever():
2 pass
3 basic.forever(on_forever)
4



Download

4. Secret Button

Challenge

Textbook

Secret Button

Did you know that the micro:bit logo is also a button? Create a program that displays the **FABULOUS** icon when the logo is pressed.

The code for when the logo is pressed is here.

```
if input.logo_is_pressed():
```

Requirements

- Create the forever function.
- Inside the forever function, use an if statement to see if the logo is pressed.
- If the logo is pressed, display the **FABULOUS** icon.

Step 1

Create the forever function.

Secret Button Step 1 of 2

1 Next

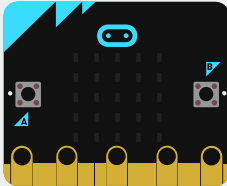
Toolbox

Search

Basic

start

forever



Download

5. Display a Design

Challenge

Textbook

Display a Design

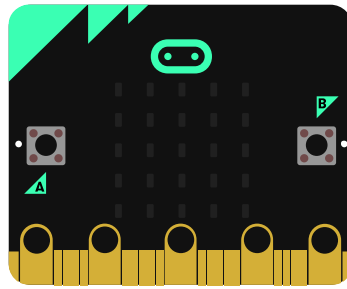
Create a program that will display a certain design if the button B is pressed.

Requirements

- Create the forever function
- Inside the forever function, create an if statement to see if the button B is pressed.
- If the button B is pressed, display an image

Answer Key

Submit



Search

Basic

start

forever

Download

Answer Keys & Solutions

Questions

1. A student wants to make the micro:bit show a heart when button A is pressed. They write this code but nothing happens when they press the button. What did they forget to include?

MULTIPLE CHOICE

Correct Answer:

- A. Quotation marks around the button name ✗ Incorrect
- B. A forever function to keep checking for button presses ✓ Correct
- C. A pause command ✗ Incorrect
- D. The `basic.forever()` function call ✗ Incorrect

Explanation:

Button checking needs to happen continuously, not just once.

2. You want to make a simple game where button A shows your score and button B clears the screen. How many if statements will you need inside your forever function?

MULTIPLE CHOICE

Correct Answer:

- A. 1 if statement ✗ Incorrect
- B. 2 if statements ✓ Correct
- C. 3 if statements ✗ Incorrect
- D. 4 if statements ✗ Incorrect

Explanation:

Each button needs its own separate if statement to check if it's pressed.

3. Consider the following code. What will happen when they press button A once?

MULTIPLE CHOICE

Correct Answer:

- A. Only the number 1 will show ✗ Incorrect
- B. Only the number 3 will show ✗ Incorrect
- C. The numbers 1, 2, and 3 will show in sequence ✓ Correct
- D. All three numbers will show at the same time ✗ Incorrect

Explanation:

All properly indented code inside an if statement runs when the condition is true.

4. A student writes this code but gets an indentation error. What needs to be fixed?

MULTIPLE CHOICE

Correct Answer:

- A. The if statement needs more indentation ✗ Incorrect
- B. The basic.show_icon line needs more indentation ✓ Correct
- C. The function definition needs less indentation ✗ Incorrect
- D. The basic.forever line needs more indentation ✗ Incorrect

Explanation:

Code inside an if statement needs to be indented further than the if line itself.

5. You want to create a program where pressing button A shows "YES" and pressing button B shows "NO". Which code structure works best?

MULTIPLE CHOICE

Correct Answer:

- A. Put both if statements in separate forever functions ✗ Incorrect
- B. Put the if statements outside any forever function ✗ Incorrect
- C. Use only one if statement that checks both buttons ✗ Incorrect

D. Put both if statements inside the same forever function

✓ Correct

Explanation:

You can check multiple buttons within the same continuous loop.

6. You want to create a program where pressing button A shows a happy face, and pressing button B shows a sad face. A friend suggests this code. Will this code work correctly?

MULTIPLE CHOICE

Correct Answer:

A. No, you cannot have two if statements in one function

✗ Incorrect

B. No, the indentation is wrong

✗ Incorrect

C. Yes, this code will work as intended

✓ Correct

D. No, you need to use Button.A and Button.B together

✗ Incorrect

Explanation:

Multiple if statements can exist independently in the same function.

7. A student wants their program to do three different things when button A is pressed: show a heart, pause for 1 second, then show a star. Where should they put the pause command?

MULTIPLE CHOICE

Correct Answer:

A. Before the if statement

✗ Incorrect

B. After all the if statements

✗ Incorrect

C. Inside the if statement, between the heart and star commands

✓ Correct

D. In a separate forever function

✗ Incorrect

Explanation:

The pause should be part of the sequence that happens when the button is pressed.

8. You notice a classmate's code is not working properly. What indentation error needs to be fixed?

MULTIPLE CHOICE

Correct Answer:

- A. Only the if statement needs to be indented ✗ Incorrect
- B. Only the basic.show_icon needs to be indented ✗ Incorrect
- C. Both the if statement and basic.show_icon need proper indentation ✓ Correct
- D. The indentation is correct ✗ Incorrect

Explanation:

Each level of code nesting needs its own level of indentation.

9. A student asks why they need to put button checking code inside a forever function instead of just writing it once. What is the best explanation?

MULTIPLE CHOICE

Correct Answer:

- A. Forever functions make the code run faster ✗ Incorrect
- B. The micro:bit needs to continuously check if buttons are being pressed ✓ Correct
- C. Buttons only work inside forever functions ✗ Incorrect
- D. It makes the code easier to read ✗ Incorrect

Explanation:

Think about when you might press a button – it could be at any time.

10. You want to create a simple counter where each press of button A adds 1 to a number displayed on screen. You start with this code. What is the main limitation of this approach?

MULTIPLE CHOICE

Correct Answer:

- A. The indentation is wrong ✗ Incorrect

B. It will always show 1 instead of counting up

✓ Correct

C. Button A is the wrong button to use

✗ Incorrect

D. The forever function is unnecessary

✗ Incorrect

Explanation:

This code shows the same number every time instead of keeping track of a count.