Pygame Events!

Pygame Events

Pygame can do more than just show images on a screen!

We can use it to figure out if the mouse moved or if a keyboard button was pressed!

These actions are called “events”! Let’s learn how to use them

How do we use it?

First we need to ask Pygame to tell us what has happened recently

```
while True:
  new_event = event.poll()
```

This line checks to see if there is a new event and saves it in a variable called new_event

How do we use it?

First we need to ask Pygame to tell us what has happened recently

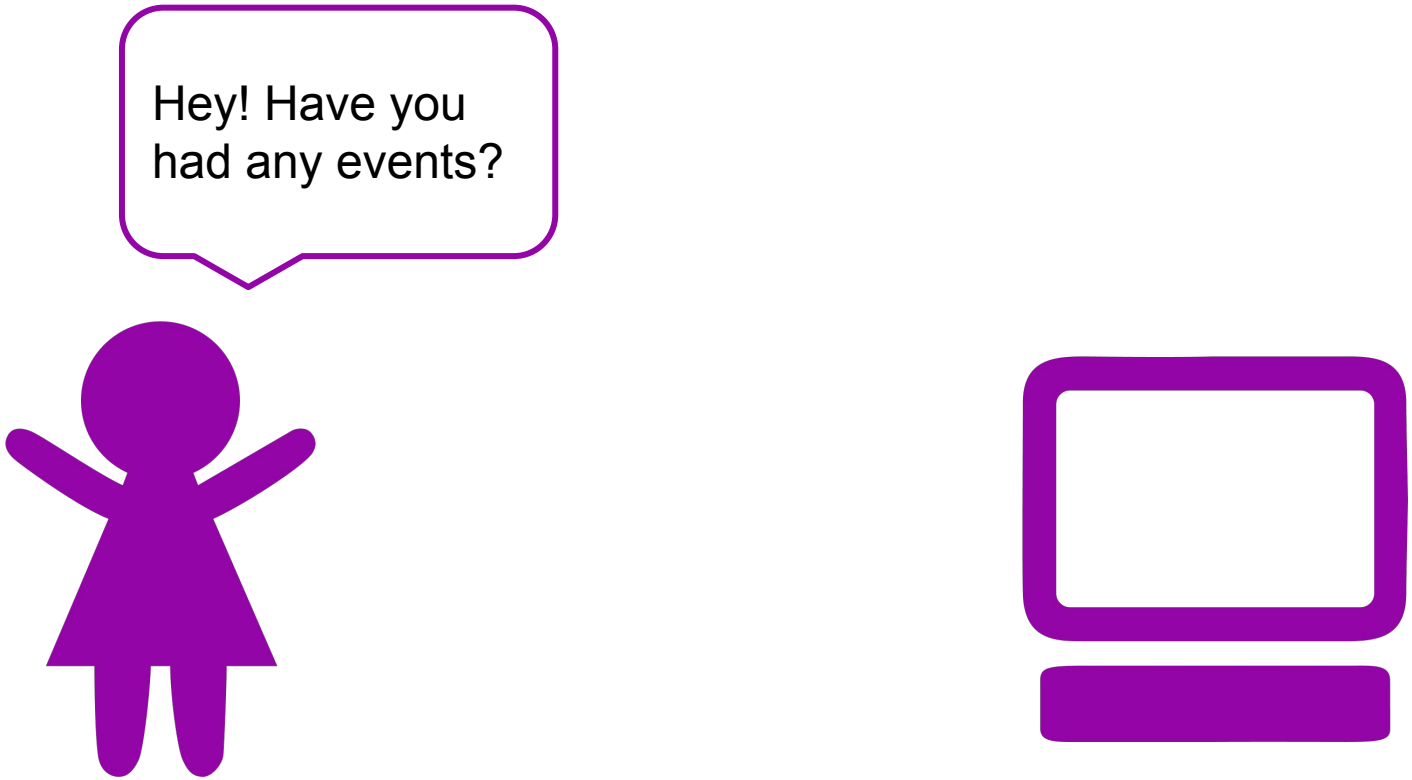
```
while True:
    new_event = event.poll()
```

But what does this line do??

This line checks to see if there is a new event and saves it in a variable called new_event

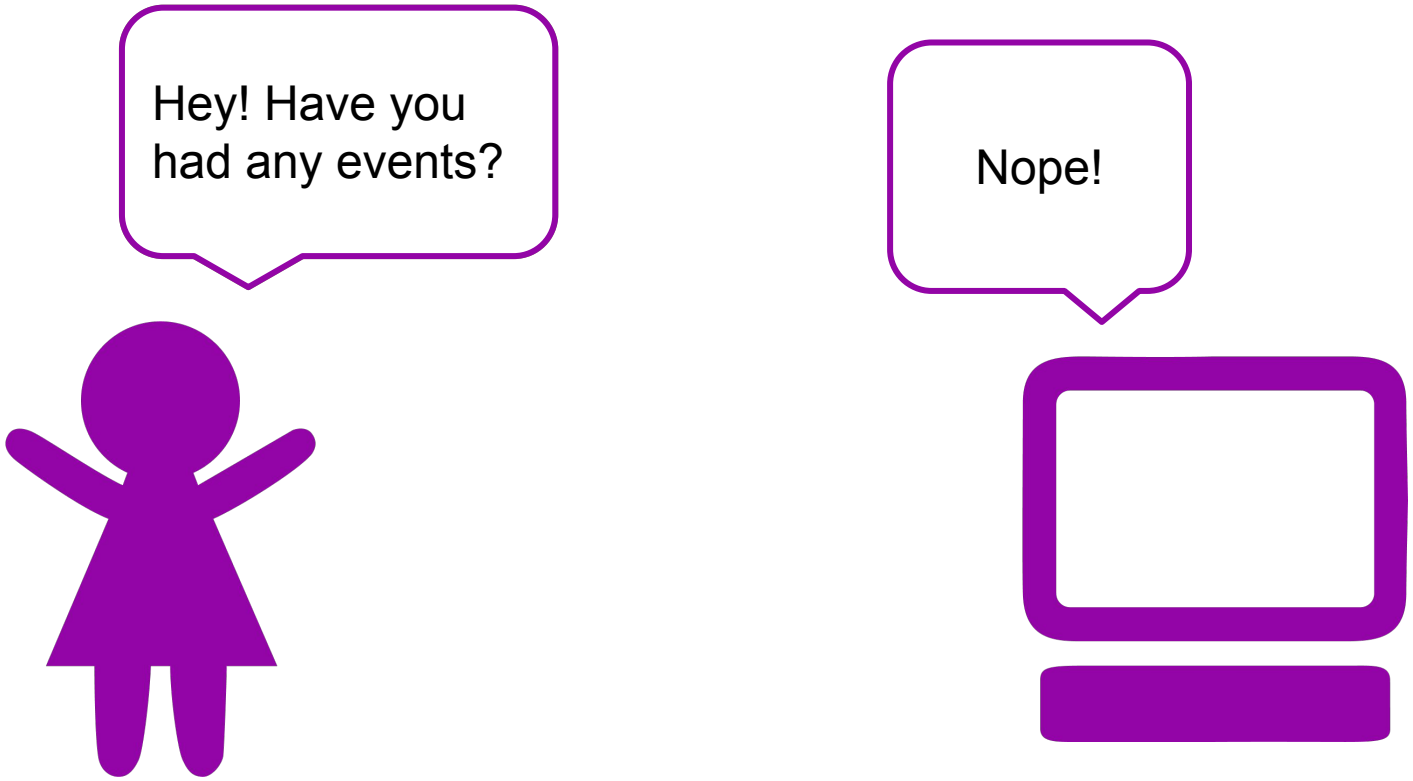
Looking for Events

Let's think of how Pygame checks for events like this:



Looking for Events

Let's think of how Pygame checks for events like this:



Looking for Events

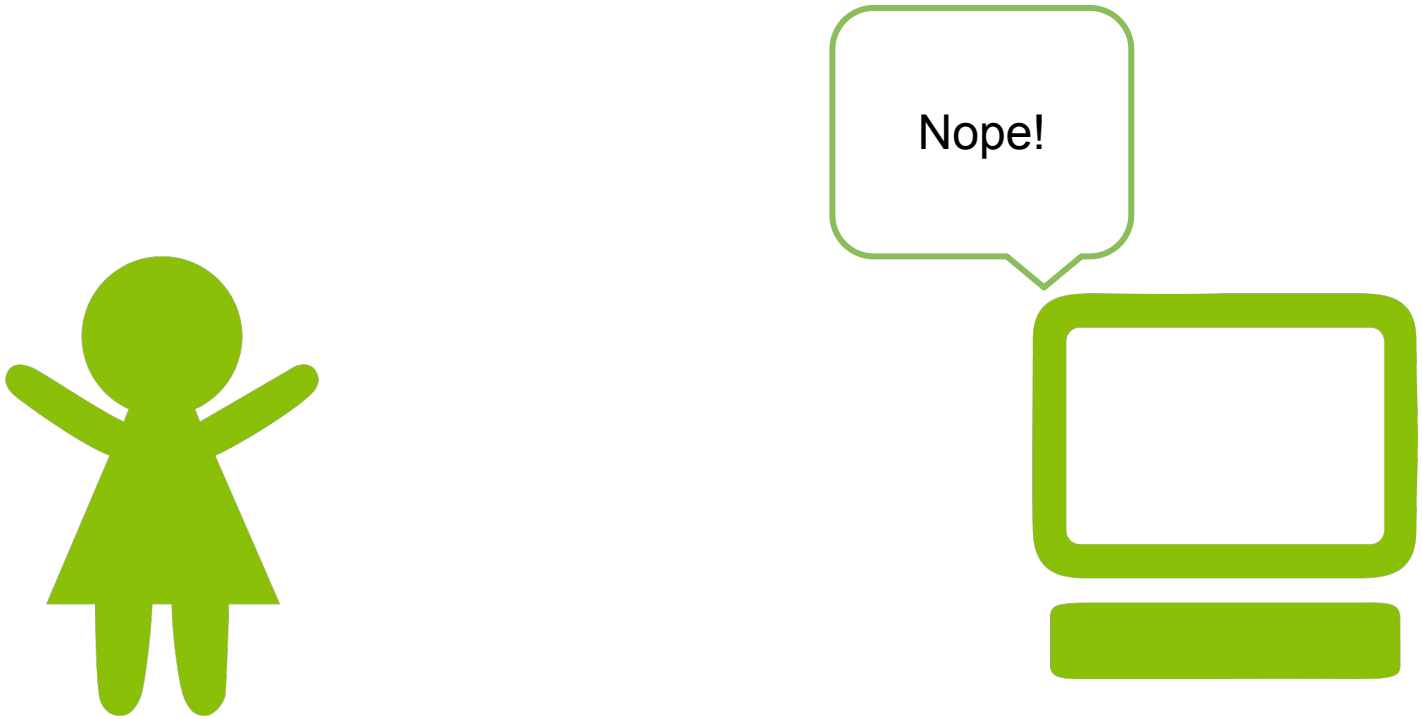
If we only ask once then we won't know if an event happens later

Looking for Events

We need to keep asking over and over again!

Looking for Events

We need to keep asking over and over again!



Looking for Events

We need to keep asking over and over again!

Looking for Events

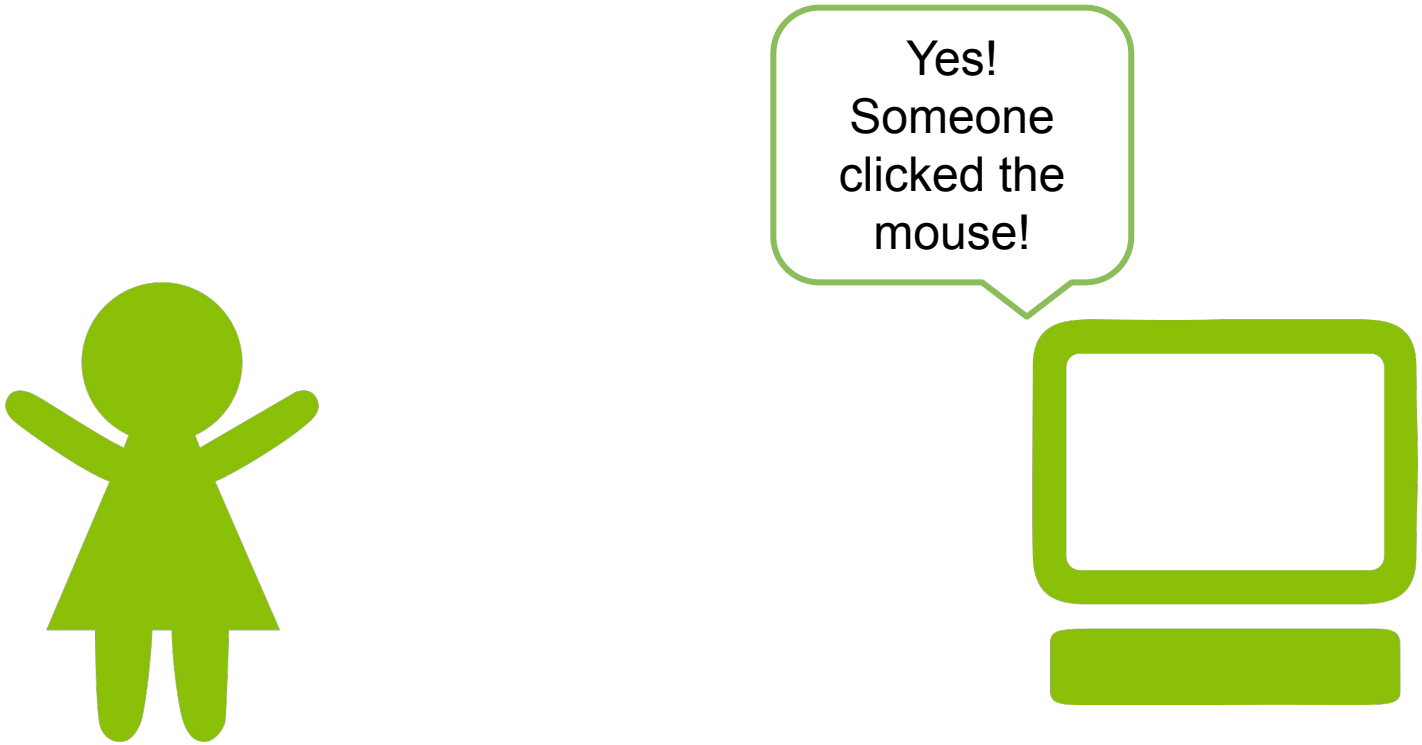
We need to keep asking over and over again!

Looking for Events

We need to keep asking over and over again!

Looking for Events

We need to keep asking over and over again!



Loops

We can do something over and over again in our code using Loops! Like this:

```
while True:
  print(“Hello”)
```

What do you think this code does?

Loops

We can do something over and over again in our code using Loops! Like this:

```
while True:
  print(“Hello”)
```

What do you think this code does?

```
Hello
Hello
Hello
Hello
Hello
Hello
Hello
```

Loops

We can do something over and over again in our code using Loops! Like this:

```
while True:
  print(“Hello”)
```

It prints “Hello” forever! Let’s have a look at what it’s doing

Loops

```
while True:
  print("Hello")
```

First, it checks to see if it should go into the loop. Because we wrote True here it will **always** go into the loop

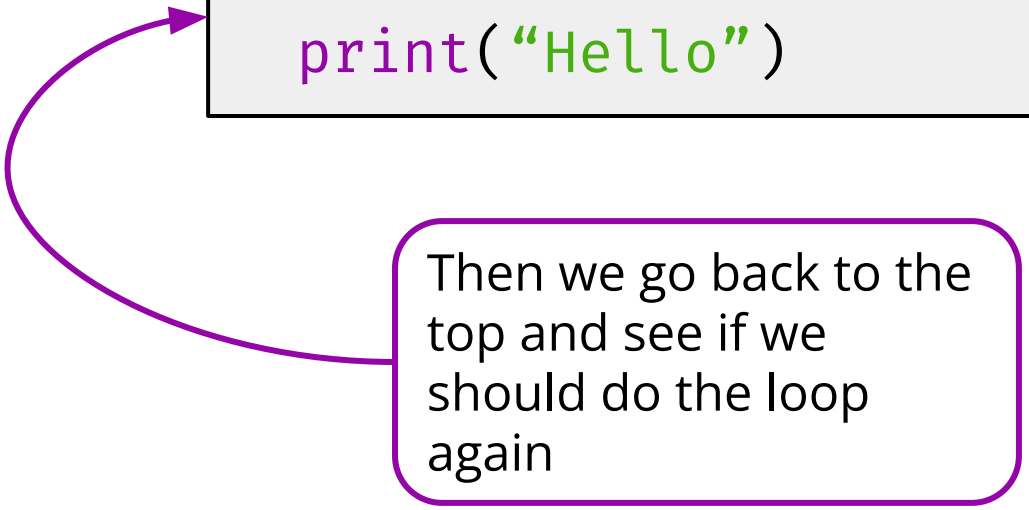
Loops

```
while True:
   print(“Hello”)
```

Then we do whatever is **inside** the loop - we print “Hello”

Loops

```
while True:
  print("Hello")
```



Loops

```
while True:
  print("Hello")
```

Because we wrote True here it will **always** go into the loop

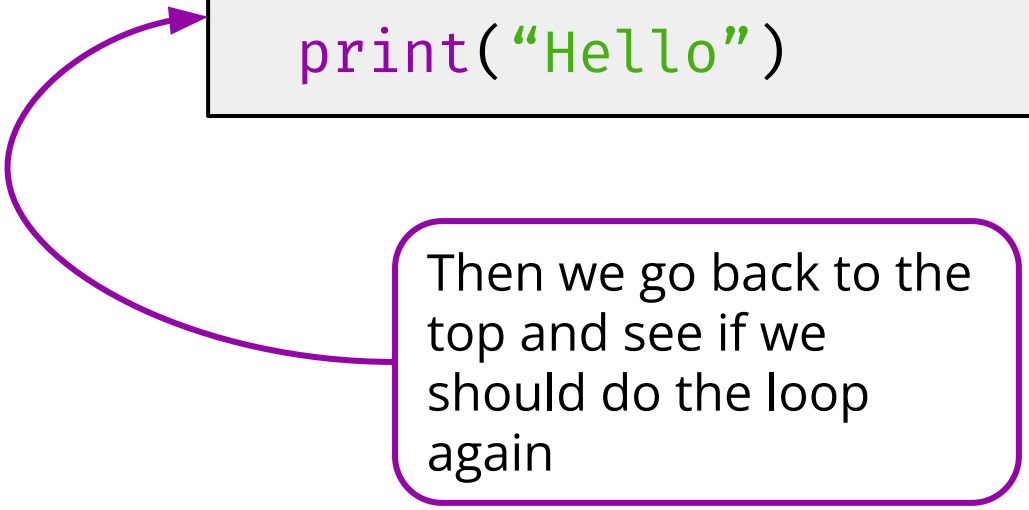
Loops

```
while True:
   print(“Hello”)
```

Then we do whatever is **inside** the loop - we print “Hello”

Loops

```
while True:
  print("Hello")
```



Loops

This pattern keeps going on and on forever! (or until you quit the program)

```
while True:
  print(“Hello”)
```

Looking for Events

Now that we understand that we need to keep asking over and over, let's have another look at that code!

Looking for Events

Now that we understand that we need to keep asking over and over, let's have another look at that code!

```
while True:
  new_event = event.poll()
```

This line checks to see if there is a new event and saves it in a variable called new_event

Looking for Events

Now that we understand that we need to keep asking over and over, let's have another look at that code!

```
while True:
    new_event = event.poll()
```

This line tells python to do it over and over. It's called a loop!

This line checks to see if there is a new event and saves it in a variable called new_event

Looking for Events

```
while True:
    new_event = event.poll()
```

First we enter the loop here

Looking for Events

```
while True:
  new_event = event.poll()
```

Hey! Have you had any events?

Then we ask if there is a new event

Looking for Events

```
while True:
  new_event = event.poll()
```

Then we ask if there is a new event

Nope!

Looking for Events

```
while True:
  new_event = event.poll()
```

Then we go back to the top and do it again

Looking for Events

```
while True:
  new_event = event.poll()
```

Hey! Have you had any events?

Now we're doing this line again!

Looking for Events

```
while True:
  new_event = event.poll()
```

Nope!

Now we're doing this line again!

Finding Events

Okay so now we know how to ask for new Events. But what do we do when we find one?

Finding Events

Okay so now we know how to ask for new Events. But what do we do when we find one?

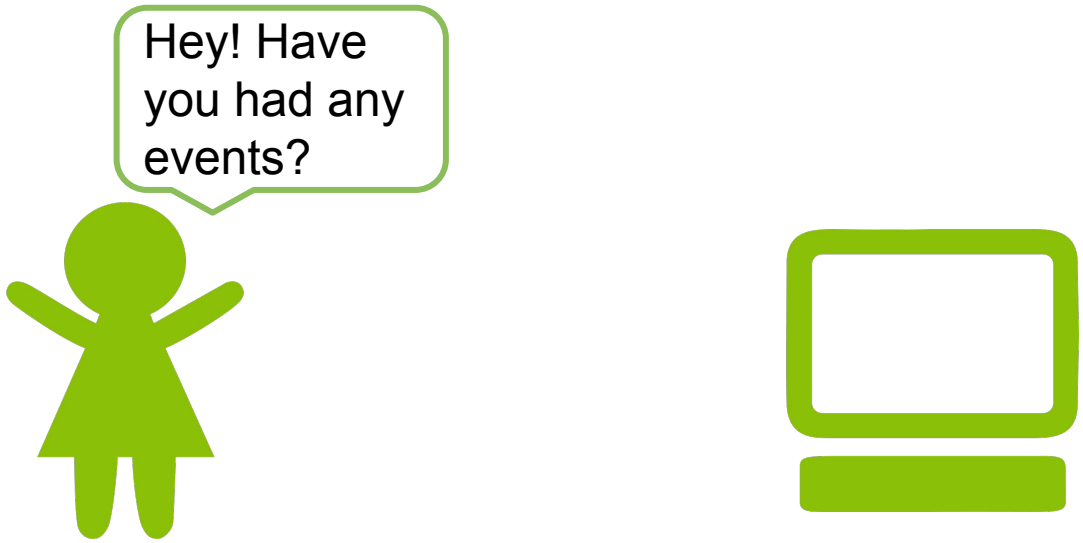
```
while True:
  new_event = event.poll()
  if new_event.type == KEYDOWN:
    print("You pressed a key!")
```

This if statement checks if the type of event was a KEY on the keyboard being pressed DOWN

Finding Events

First we check if there are any events

```
while True:
    new_event = event.poll()
    if new_event.type == KEYDOWN:
        print("You pressed a key!")
```



Finding Events

```
while True:
    new_event = event.poll()
    if new_event.type == KEYDOWN:
        print("You pressed a key!")
```

Then we check what type of event it was

Yep! It was a KEYDOWN event

Finding Events

```
while True:
    new_event = event.poll()
    if new_event.type == KEYDOWN:
        print("You pressed a key!")
```

If it's the event we want then we print this line

Pressing Keys

But we want to know *which* key they pressed! Not just if they pressed any key on the keyboard!

Pressing Keys

But we want to know *which* key they pressed! Not just if they pressed any key on the keyboard!

```
while True:
  new_event = event.poll()
  if new_event.type == KEYDOWN and new_event.key == K_SPACE:
    print("You pressed the space key!")
```

This now also checks if they key was the SPACE key

Finding Events

```
while True:
  new_event = event.poll()
  if new_event.type == KEYDOWN and new_event.key == K_SPACE:
    print("You pressed the space key!")
```

First we check if there are any events

Finding Events

```
while True:
  new_event = event.poll()
  if new_event.type == KEYDOWN and new_event.key == K_SPACE:
    print("You pressed the space key!")
```

Then we check what type of event it was

And we check what key was pressed

Yep! It was a KEYDOWN event using the SPACE key!

Finding Events

```
while True:
  new_event = event.poll()
  if new_event.type == KEYDOWN and new_event.key == K_SPACE:
    print("You pressed the space key!")
```

Now we know what the event is we can print

Project time!

The **key** to doing the next part was all in these slides

Try to do the next Part

In the **event** of confusion, the tutors will be around to help!