

Homework 3 Review

Goal: Recreate Twitter UI to Post a tweet

Write a tweet

Call me Ishmael. Some years ago—
never mind how long precisely—having

-37

Post Tweet

POSTS

chilton Most recent post

chilton More ambitious third post

chilton Second post

chilton First post

HTML: Why do I have the tweet_length_feedback div?

```
17 #tweet_length_feedback{
18   width: 20px;
19   display: inline-block;
20 }
21
```

Write a tweet

30

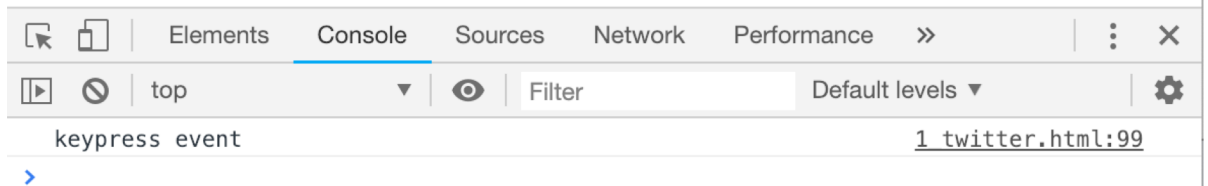
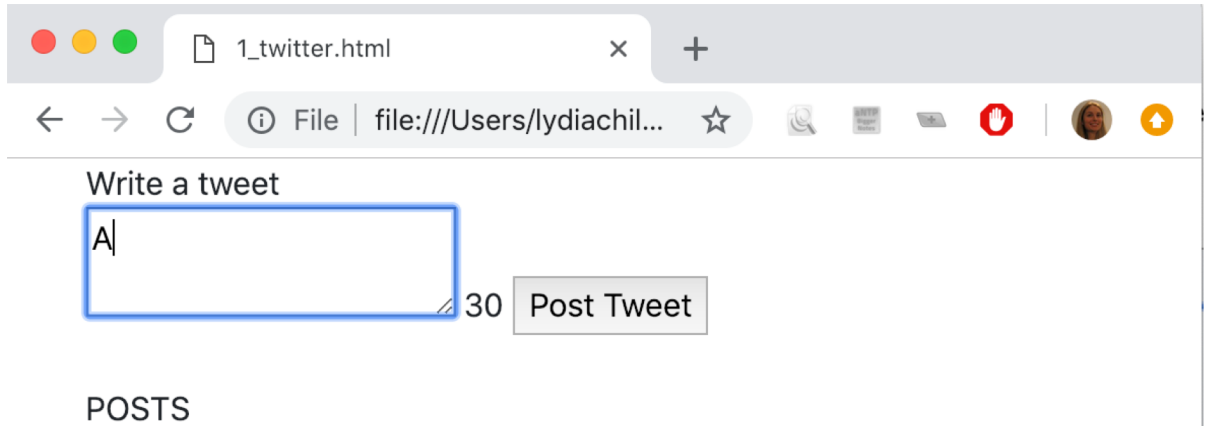
Post Tweet

```
100 <div>Write a tweet</div>
101 <textarea id="write_tweet"></textarea>
102 <div id="tweet_length_feedback"></div>
103 <button id="post_tweet"> Post Tweet</button>
```

JS: What does this do?

```
95
96   $(document).ready(function(){
97     $("#write_tweet").keypress(function(){
98       var tweet_text = $(this).val()
99       console.log("keypress event")
100     })
101   })
102
```

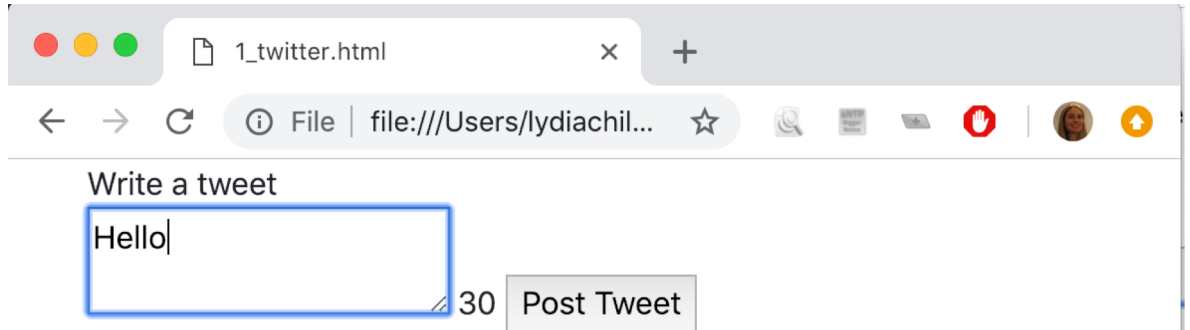
Line 97 attaches a “keypress” event listener to the textarea



What does this do?

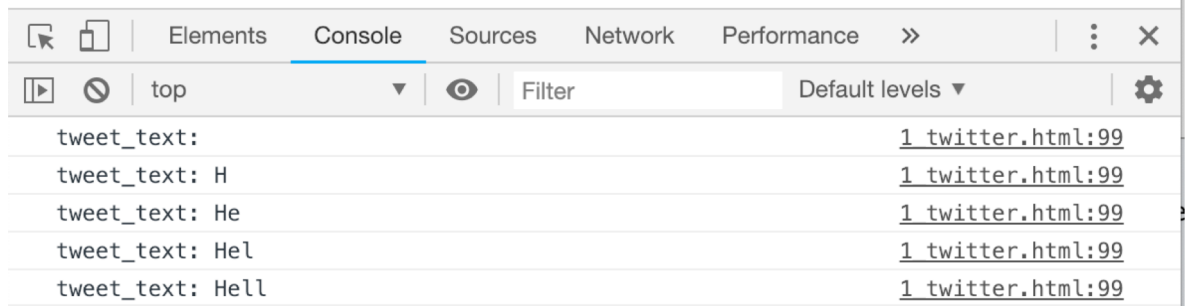
Why is this a good next step?

```
96  
97     $("#write_tweet").keypress(function(){  
98         var tweet_text = $(this).val()  
99         console.log("tweet_text: "+tweet_text)  
100  
101  
102     })  
103
```



It allows us to check if line 98 does what we expect.

Does it?



Not quite. What might be the problem?

```
96
97     $("#write_tweet").keypress(function(){
98         var tweet_text = $(this).val()
99         console.log("tweet_text: "+tweet_text)
100
101     })
102
103
```

oogle

keypress event



All

News

Videos

Images

Maps

More

Settings

Tools

About 3,800,000 results (0.60 seconds)

The **keypress()** method triggers the **keypress event**, or attaches a function to run when a **keypress event** occurs. The **keypress event** is similar to the **keydown event**. The **event** occurs when a button is pressed down. However, the **keypress event** is not fired for all keys (e.g. ALT, CTRL, SHIFT, ESC).

[jQuery keypress\(\) Method - W3Schools](https://www.w3schools.com/jquery/event_keypress.asp)

https://www.w3schools.com/jquery/event_keypress.asp

About this result Feedback

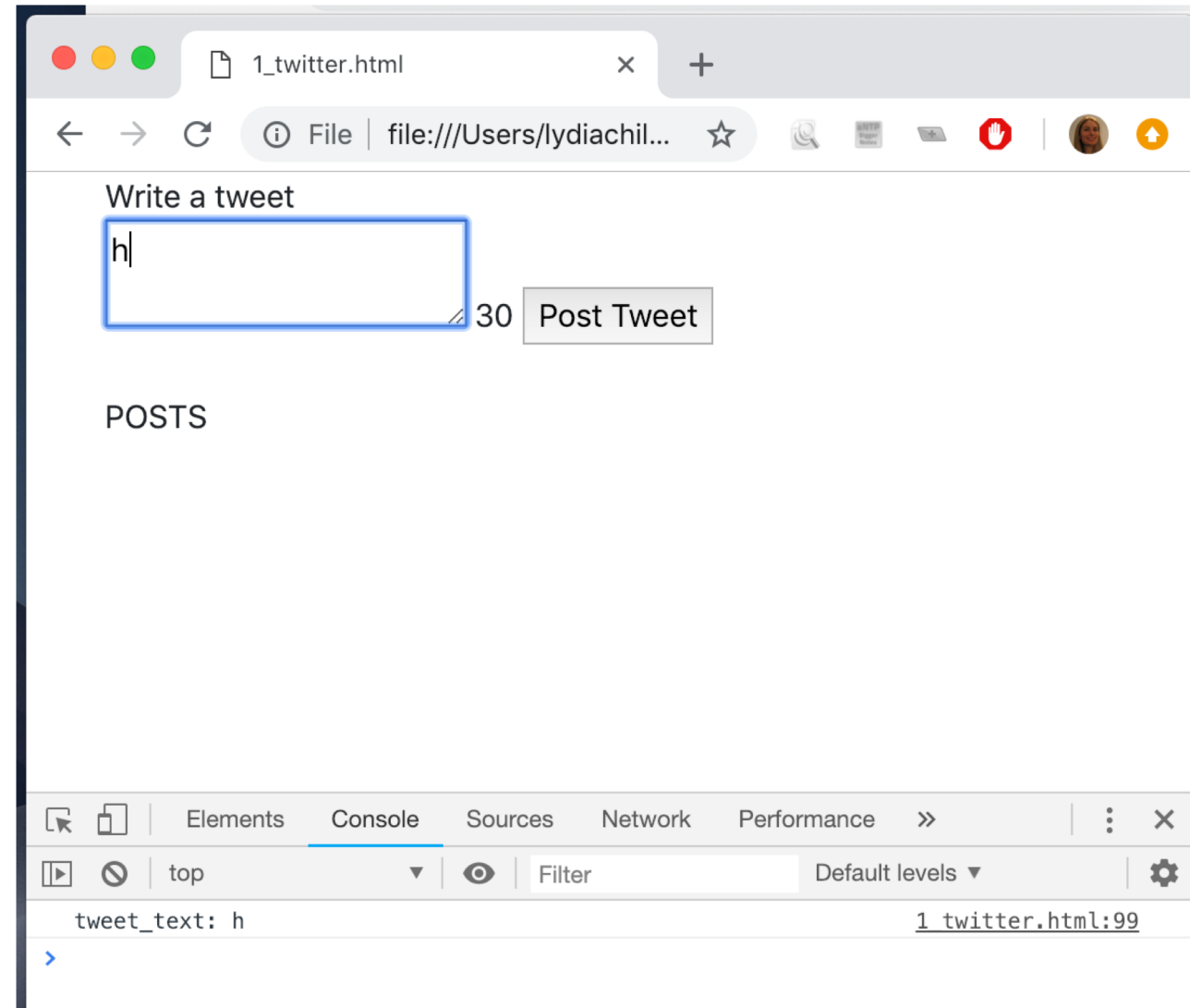
What should we try to fix?

Keyup event seems better. Does it work?

```
96  
97     $("#write_tweet").keypress(function(){  
98         var tweet_text = $(this).val()  
99         console.log("tweet_text: "+tweet_text)  
100  
101  
102     })  
103
```

```
96  
97     $("#write_tweet").keyup(function(){  
98         var tweet_text = $(this).val()  
99         console.log("tweet_text: "+tweet_text)  
100  
101  
102     })  
103
```

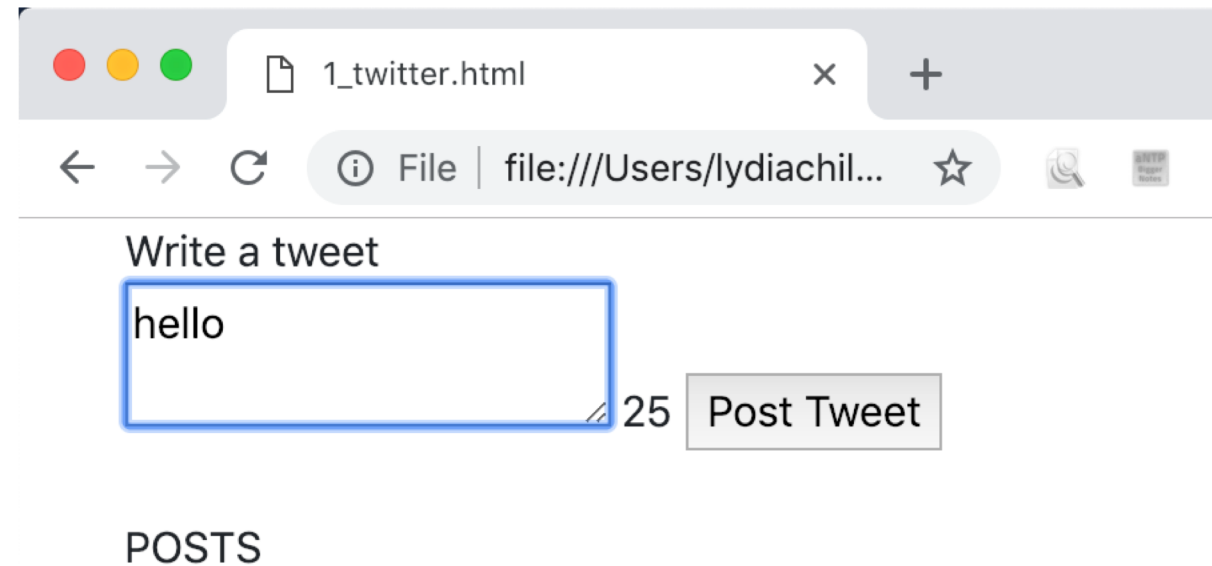
Now what?



The screenshot shows a web browser window with a single tab titled "1_twitter.html". The address bar shows the file path "file:///Users/lydiachil...". The page content includes a text input field with the label "Write a tweet" containing the letter "h", a "30" character count, and a "Post Tweet" button. Below the input is a section labeled "POSTS". The browser's developer console is open, showing a log entry: "tweet_text: h" from "1_twitter.html:99".

Display the characters left

```
96  
97     $("#write_tweet").keyup(function(){  
98         var tweet_text = $(this).val()  
99         console.log("tweet_text: "+tweet_text)  
100  
101         var tweet_length = tweet_text.length  
102         var chars_left = max_tweet_length - tweet_length  
103  
104         $("#tweet_length_feedback").html(chars_left)  
105     })  
106
```



Now what?

Turn negative characters red. Like this?

```
96
97     $("#write_tweet").keyup(function(){
98         var tweet_text = $(this).val()
99         console.log("tweet_text: "+tweet_text)
100
101         var tweet_length = tweet_text.length
102         var chars_left = max_tweet_length-tweet_length
103
104
105         if(chars_left<0){
106             $("#tweet_length_feedback").css("color", "red").css("font-weight", "bold")
107         }else{
108             // ???
109         }
110
111
112
113         $("#tweet_length_feedback").html(chars_left)
114
115     })
116
```

Now what?

Add a class

```
97     $("#write_tweet").keyup(function(){
98         var tweet_text = $(this).val()
99         console.log("tweet_text: "+tweet_text)
100
101         var tweet_length = tweet_text.length
102         var chars_left = max_tweet_length - tweet_length
103
104
105         if(chars_left < 0){
106             //turn characters red
107             // NO
108             //$("#tweet_length_feedback").css("color", "red").css("font-weight", "bold")
109
110             // YES
111             $("#tweet_length_feedback").addClass("length_warning")
112
113         }else{
114             // ???
115
116         }
117
118         $("#tweet_length_feedback").html(chars_left)
119
120     })
121
```

```
.length_warning{
    color: red;
    font-weight: bold;
}
```

Now what?

Next - remove the class. When?

```
96
97     $("#write_tweet").keyup(function(){
98         var tweet_text = $(this).val()
99         console.log("tweet_text: "+tweet_text)
100
101         var tweet_length = tweet_text.length
102         var chars_left = max_tweet_length - tweet_length
103
104
105         if(chars_left < 0){
106             //turn characters red
107             // NO
108             //$("#tweet_length_feedback").css("color", "red").css("font-weight", "bold")
109
110             // YES
111             $("#tweet_length_feedback").addClass("length_warning")
112
113         }else{
114             $("#tweet_length_feedback").removeClass("length_warning")
115
116         }
117
118         $("#tweet_length_feedback").html(chars_left)
119
120     })
121
```

What next?

Write a tweet

Call me Ishmael. Some years ago—
never mind how long precisely—having

-37

Post Tweet

POSTS

chilton Most recent post
chilton More ambitious third post
chilton Second post
chilton First post



DO THIS

How do we start?

```
34 <script>
35   var username = "chilton"
36   var max_tweet_length = 30
37
38   $(document).ready(function(){
39     $("#post_tweet").click(function(){
40       var tweet_text = $("#write_tweet").val()
41       create_new_post(username, tweet_text)
42     })
43
```

Add a click event listener (handler)
to the post_tweet button

What does create_post do?

```
96  
97   var create_new_post = function(username, tweet_text){  
98     var newDiv = $("99     var nameSpan = $("100    nameSpan.html(username)  
101  
102    var tweetDiv = $("103    tweetDiv.html(tweet_text)  
104  
105    newDiv.append(nameSpan)  
106    newDiv.append(tweetDiv)  
107    $("#posts_container").prepend(newDiv)  
108  }  
109
```

POSTS

chilton Hello

div.single_post_container | 300 × 26

chilton Hello

span.post_username | 63.22 × 18

chilton Hello

div#posts_container | 510 × 26

chilton Hello

POSTS

span.tweettext | 36.97 × 18

chilton Hello

What else does it need to do?

```
34 <script>
35   var username = "chilton"
36   var max_tweet_length = 30
37
38   $(document).ready(function(){
39     $("#post_tweet").click(function(){
40       var tweet_text = $("#write_tweet").val()
41       create_new_post(username, tweet_text)
42     })
43
44     // clear text
45     $("#write_tweet").val("")
```

- a. When the user presses the “post tweet” button the following things must happen:
 - i. The post must appear with the poster’s username (not an image). The username can be hard coded in JavaScript (but not in HTML). The post text cannot be hard coded (obviously).
 - ii. New posts must appear at the top of the list of posts, so that users see the latest tweets at the top of the list.
 - iii. The text in the box where users write tweets must disappear.
 - iv. The number that counts the number of characters remaining must return to the maximum number.
 - v. The cursor must return to the box where the user writes tweets (so that they can immediately start writing another tweet!)

After the text clears, then what?

```
34 <script>
35   var username = "chilton"
36   var max_tweet_length = 30
37
38   $(document).ready(function(){
39     $("#post_tweet").click(function(){
40       var tweet_text = $("#write_tweet").val()
41       create_new_post(username, tweet_text)
42     })
43
44     // clear text
45     $("#write_tweet").val("")
46
47     //clear counter
48     $("#tweet_length_feedback").html(max_tweet_length)
49     $("#tweet_length_feedback").removeClass("length_warning")
```

a. When the user presses the “post tweet” button the following things must happen:

- i. The post must appear with the poster’s username (not an image). The username can be hard coded in JavaScript (but not in HTML). The post text cannot be hard coded (obviously).
- ii. New posts must appear at the top of the list of posts, so that users see the latest tweets at the top of the list.
- iii. The text in the box where users write tweets must disappear.
- iv. The number that counts the number of characters remaining must return to the maximum number.
- v. The cursor must return to the box where the user writes tweets (so that they can immediately start writing another tweet!)

After the tweet length = 30m then what?

```
34 <script>
35   var username = "chilton"
36   var max_tweet_length = 30
37
38   $(document).ready(function(){
39     $("#post_tweet").click(function(){
40       var tweet_text = $("#write_tweet").val()
41       create_new_post(username, tweet_text)
42     })
43
44     // clear text
45     $("#write_tweet").val("")
46
47     //clear counter
48     $("#tweet_length_feedback").html(max_tweet_length)
49     $("#tweet_length_feedback").removeClass("length_warning")
50
51     //return focus to textarea
52     $("#write_tweet").focus()
```

a. When the user presses the “post tweet” button the following things must happen:

- i. The post must appear with the poster’s username (not an image). The username can be hard coded in JavaScript (but not in HTML). The post text cannot be hard coded (obviously).
- ii. New posts must appear at the top of the list of posts, so that users see the latest tweets at the top of the list.
- iii. The text in the box where users write tweets must disappear.
- iv. The number that counts the number of characters remaining must return to the maximum number.
- v. The cursor must return to the box where the user writes tweets (so that they can immediately start writing another tweet!)

There's one more thing I do. Why?

```
34 <script>
35   var username = "chilton"
36   var max_tweet_length = 30
37
38   $(document).ready(function(){
39     $("#write_tweet").focus()
40     $("#tweet_length_feedback").html(max_tweet_length)
41
42
```

Write a tweet

30

Post Tweet

```
100 <div>Write a tweet</div>
101 <textarea id="write_tweet"></textarea>
102 <div id="tweet_length_feedback"></div>
103 <button id="post_tweet"> Post Tweet</button>
```