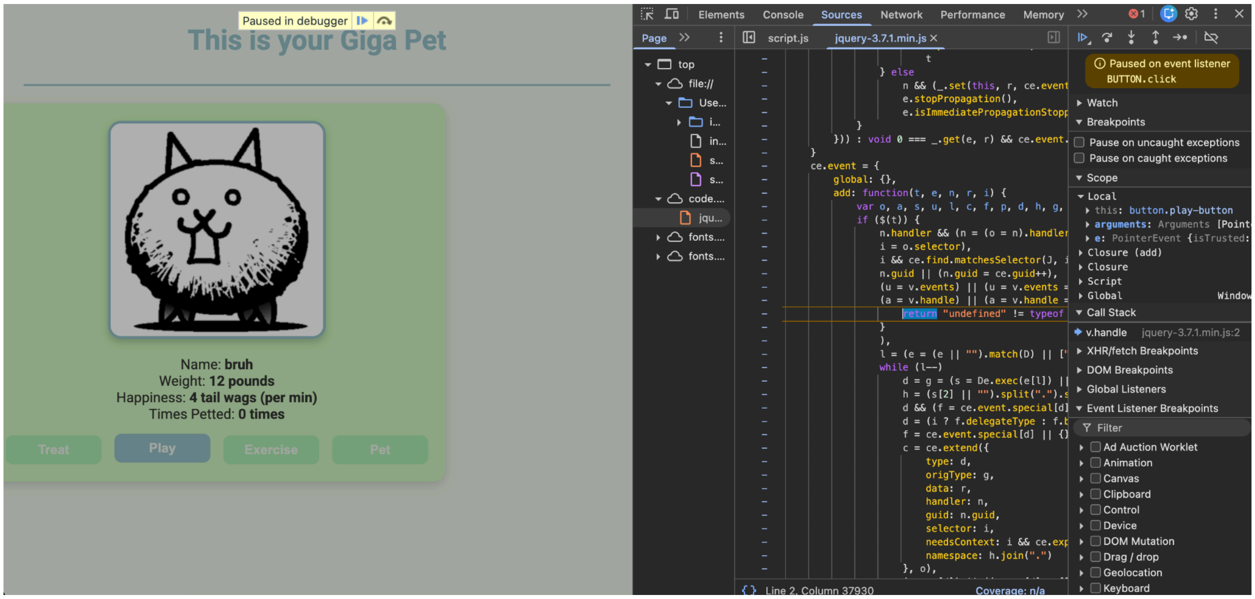


Name: **bruh**
Weight: **12 pounds**
Happiness: **4 tail wags (per min)**
Times Petted: **0 times**

```
                            t
                    } else
                        n && (_.set(this, r, ce.event
                        e.stopPropagation(),
                        e.isImmediatePropagationStopp
                }
            })) : void 0 === _.get(e, r) && ce.event.
        }
        ce.event = {
            global: {},
            add: function(t, e, n, r, i) {
                var o, a, s, u, l, c, f, p, d, h, g,
                if ($(t)) {
                    n.handler && (n = (o = n).handler
                    i = o.selector),
                    i && ce.find.matchesSelector(J, i
                    n.guid || (n.guid = ce.guid++),
                    (u = v.events) || (u = v.events =
                    (a = v.handle) || (a = v.handle =
                        return "undefined" != typeof
                    }
                    ),
                    l = (e = (e || "").match(D) || ["
                    while (l--)
                        d = g = (s = De.exec(e[l]) ||
                        h = (s[2] || "").split(".").s
                        d && (f = ce.event.special[d]
                        d = (i ? f.delegateType : f.b
                        f = ce.event.special[d] || {}
                        c = ce.extend({
                            type: d,
                            origType: g,
                            data: r,
                            handler: n,
                            guid: n.guid,
                            selector: i,
                            needsContext: i && ce.exp
                            namespace: h.join(".")
                        }, o),
```

Line 2, Column 37930 Coverage: n/a

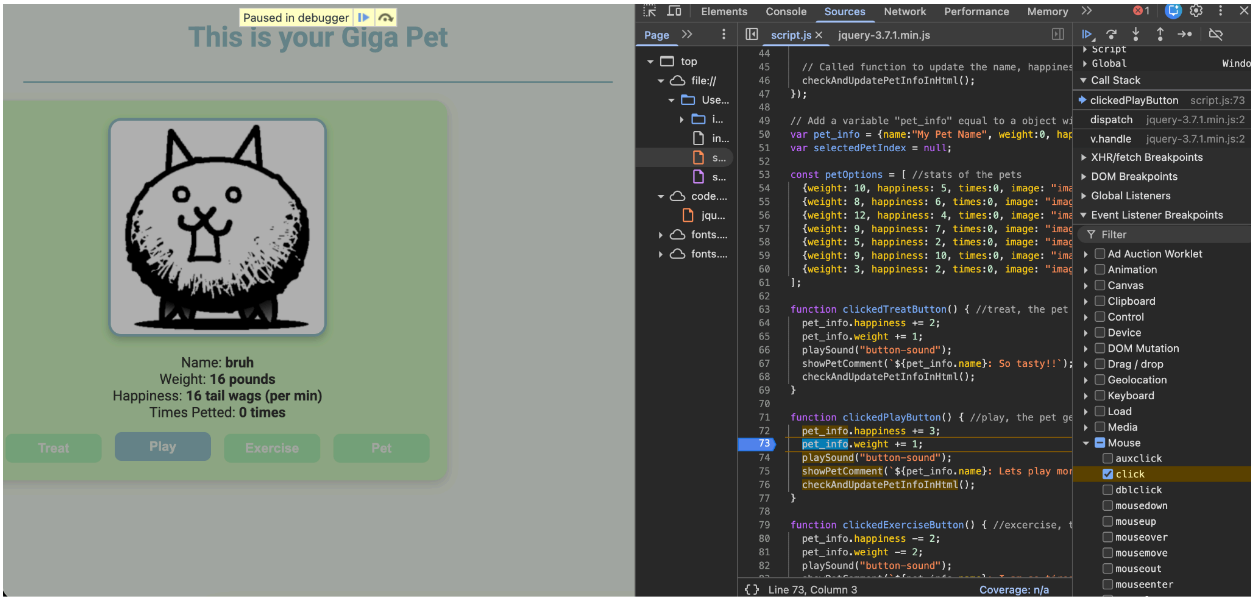
Paused on event listener BUTTON.click

☐ Pause on uncaught exceptions
☐ Pause on caught exceptions

```
Local
  this: button.play-button
  arguments: Arguments [Point
  e: PointerEvent {isTrusted:
Closure (add)
Closure
Script
Global                     Window
```

Call Stack
v.handle jquery-3.7.1.min.js:2

I first open the debugger by right clicking and pressing inspect. We then go to the sources tab to find the debugger. This allows us to set breakpoints and see how the code is executed and see any local variables, etc. Such as in the scope, we can see in the script tab, the array of the pets attributes.



Name: **bruh**
Weight: **16 pounds**
Happiness: **16 tail wags (per min)**
Times Petted: **0 times**

```

  // Called function to update the name, happines
  checkAndUpdatePetInfoInHtml();
});

// Add a variable "pet_info" equal to a object wi
var pet_info = {name:"My Pet Name", weight:0, hap
var selectedPetIndex = null;

const petOptions = [ //stats of the pets
  {weight: 10, happiness: 5, times:0, image: "ima
  {weight: 8, happiness: 6, times:0, image: "imag
  {weight: 12, happiness: 4, times:0, image: "ima
  {weight: 9, happiness: 7, times:0, image: "imag
  {weight: 5, happiness: 2, times:0, image: "imag
  {weight: 9, happiness: 10, times:0, image: "ima
  {weight: 3, happiness: 2, times:0, image: "imag
];

function clickedTreatButton() { //treat, the pet
  pet_info.happiness += 2;
  pet_info.weight += 1;
  playSound("button-sound");
  showPetComment(`${pet_info.name}: So tasty!!`);
  checkAndUpdatePetInfoInHtml();
}

function clickedPlayButton() { //play, the pet ge
  pet_info.happiness += 3;
  pet_info.weight += 1;
  playSound("button-sound");
  showPetComment(`${pet_info.name}: Lets play mor
  checkAndUpdatePetInfoInHtml();
}

function clickedExerciseButton() { //excercise, t
  pet_info.happiness -= 2;
  pet_info.weight -= 2;
  playSound("button-sound");
```

Line 73, Column 3 Coverage: n/a

```
Script
Global                     Windo
```

Call Stack
clickedPlayButton script.js:73
dispatch jquery-3.7.1.min.js:2
v.handle jquery-3.7.1.min.js:2

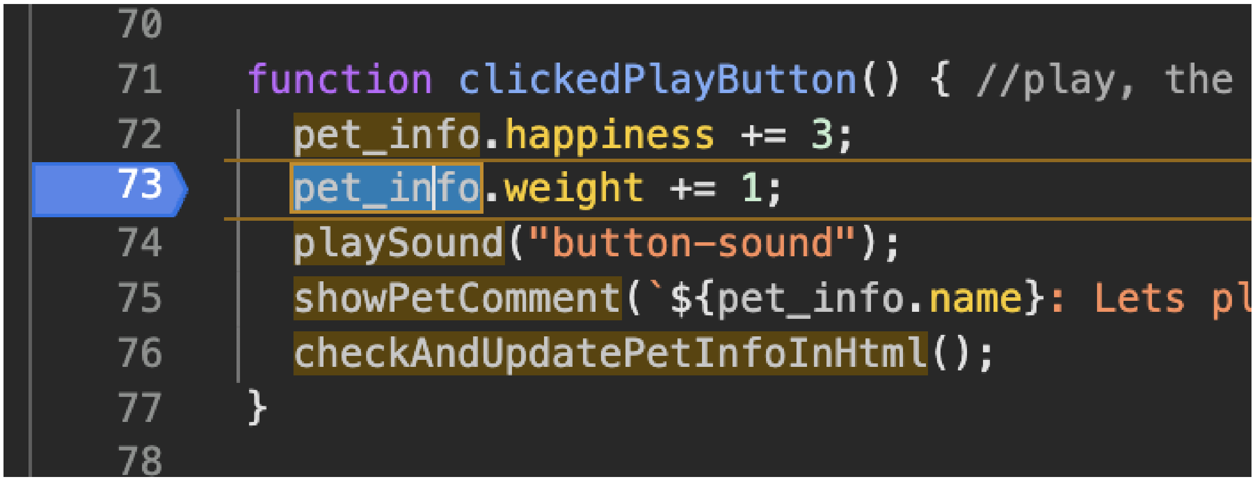
☑ click

The play button is not functioning correctly, we can debug it by creating a breakpoint to where we think the bug or mistake is. We can do this by also clicking on the Event Listener

Breakpoints tab and making it so that anytime we click, a breakpoint will happen and the code will be paused. So when we clicked the Play button, it paused the code to the breakpoint we set earlier.

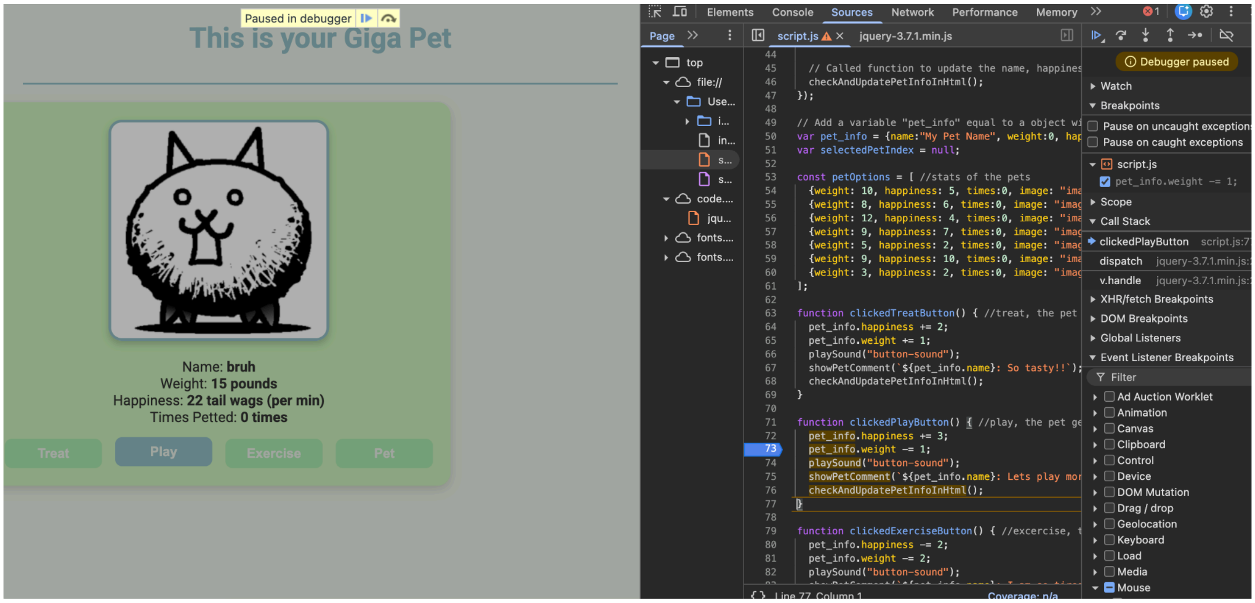
I stepped through the clickedPlayButton function until we got to the mistake.

`pet_info.weight += 1;` is set to += instead of -=.



```
function clickedPlayButton() { //play, the
  pet_info.happiness += 3;
  pet_info.weight += 1;
  playSound("button-sound");
  showPetComment(`${pet_info.name}: Lets pl
  checkAndUpdatePetInfoInHtml();
}
```

We can fix this now and by changing the code right in the debugger and see if the next button press is fixed.



Paused in debugger

This is your Giga Pet

Name: **bruh**
Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**

```
  // Called function to update the name, happines
  checkAndUpdatePetInfoInHtml();
});

// Add a variable "pet_info" equal to a object wi
var pet_info = {name:"My Pet Name", weight:0, hap
var selectedPetIndex = null;

const petOptions = [ //stats of the pets
  {weight: 10, happiness: 5, times:0, image: "ima
  {weight: 8, happiness: 6, times:0, image: "imag
  {weight: 12, happiness: 4, times:0, image: "ima
  {weight: 9, happiness: 7, times:0, image: "imag
  {weight: 5, happiness: 2, times:0, image: "imag
  {weight: 9, happiness: 10, times:0, image: "ima
  {weight: 3, happiness: 2, times:0, image: "imag
];

function clickedTreatButton() { //treat, the pet
  pet_info.happiness += 2;
  pet_info.weight += 1;
  playSound("button-sound");
  showPetComment(`${pet_info.name}: So tasty!!`);
  checkAndUpdatePetInfoInHtml();
}

function clickedPlayButton() { //play, the pet ge
  pet_info.happiness += 3;
  pet_info.weight -= 1;
  playSound("button-sound");
  showPetComment(`${pet_info.name}: Lets play mor
  checkAndUpdatePetInfoInHtml();
}

function clickedExerciseButton() { //excercise, t
  pet_info.happiness -= 2;
  pet_info.weight -= 2;
  playSound("button-sound");
```

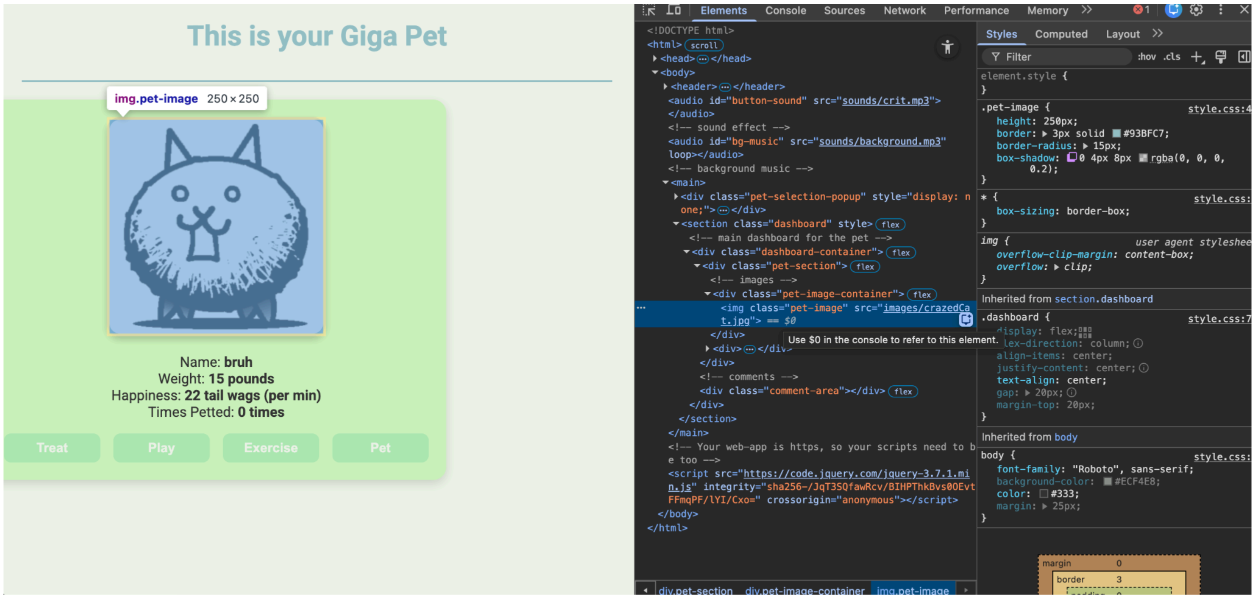
Line 77, Column 1 Coverage: n/a

Debugger paused

Breakpoints
□ Pause on uncaught exceptions
□ Pause on caught exceptions
script.js
☑ pet_info.weight -= 1;

Call Stack
clickedPlayButton script.js:7
dispatch jquery-3.7.1.min.js:
v.handle jquery-3.7.1.min.js:

As we can see that the fix was good, and the weight of bruh was changed from 16 to 15 which is what is intended.



This is your Giga Pet

img.pet-image 250 × 250

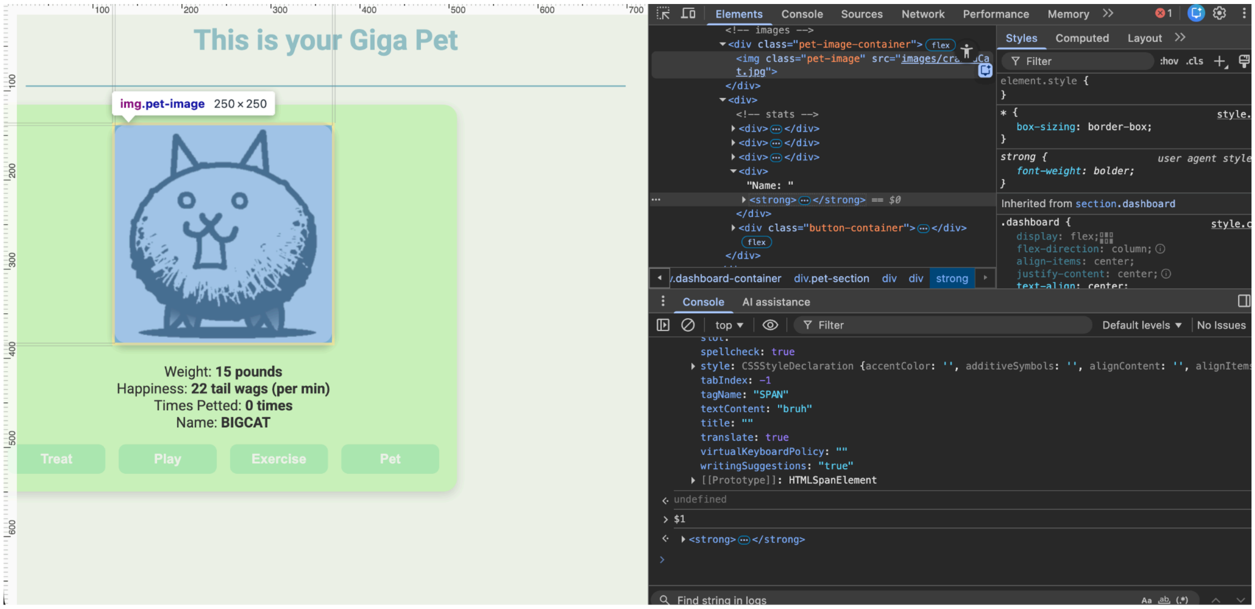
Name: **bruh**
Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**

```
<!DOCTYPE html>
<html> scroll
 <head> ⋯ </head>
 <body>
  <header> ⋯ </header>
  <audio id="button-sound" src="sounds/crit.mp3">
  </audio>
  <!-- sound effect -->
  <audio id="bg-music" src="sounds/background.mp3"
  loop></audio>
  <!-- background music -->
  <main>
   <div class="pet-selection-popup" style="display: n
   one;"> ⋯ </div>
   <section class="dashboard" style> flex
    <!-- main dashboard for the pet -->
    <div class="dashboard-container"> flex
     <div class="pet-section"> flex
      <!-- images -->
      <div class="pet-image-container"> flex
       <img class="pet-image" src="images/crazedCa
       t.jpg"> == $0
      </div>
      <div> ⋯ </div>
     </div>
     <!-- comments -->
     <div class="comment-area"></div> flex
    </div>
   </section>
  </main>
  <!-- Your web-app is https, so your scripts need to b
  e too -->
  <script src="https://code.jquery.com/jquery-3.7.1.mi
  n.js" integrity="sha256-/JqT3SQfawRcv/BIHPThkBvs0OEvt
  FFmqPF/lYI/Cxo=" crossorigin="anonymous"></script>
 </body>
</html>
```

Use $0 in the console to refer to this element.

```
element.style {
}
.pet-image {                                style.css:4
   height: 250px;
   border: ▸ 3px solid #93BFC7;
   border-radius: ▸ 15px;
   box-shadow: 0 4px 8px rgba(0, 0, 0,
       0.2);
}
* {                                         style.css:
   box-sizing: border-box;
}
img {                          user agent styleshee
   overflow-clip-margin: content-box;
   overflow: ▸ clip;
}
Inherited from section.dashboard
.dashboard {                                style.css:7
   display: flex;
   flex-direction: column;
   align-items: center;
   justify-content: center;
   text-align: center;
   gap: ▸ 20px;
   margin-top: 20px;
}
Inherited from body
body {                                      style.css:
   font-family: "Roboto", sans-serif;
   background-color: #ECF4E8;
   color: #333;
   margin: ▸ 25px;
}
```

I inspected a node and it highlighted the image and showed the size of the image. <img class="pet-image" src="images/crazedCat.jpg"> It automatically highlights what your mouse hovers over to allow for you to get more information about the node. We are also able to navigate using a keyboard. This is done by using the arrow keys.



This is your Giga Pet

img.pet-image 250 × 250

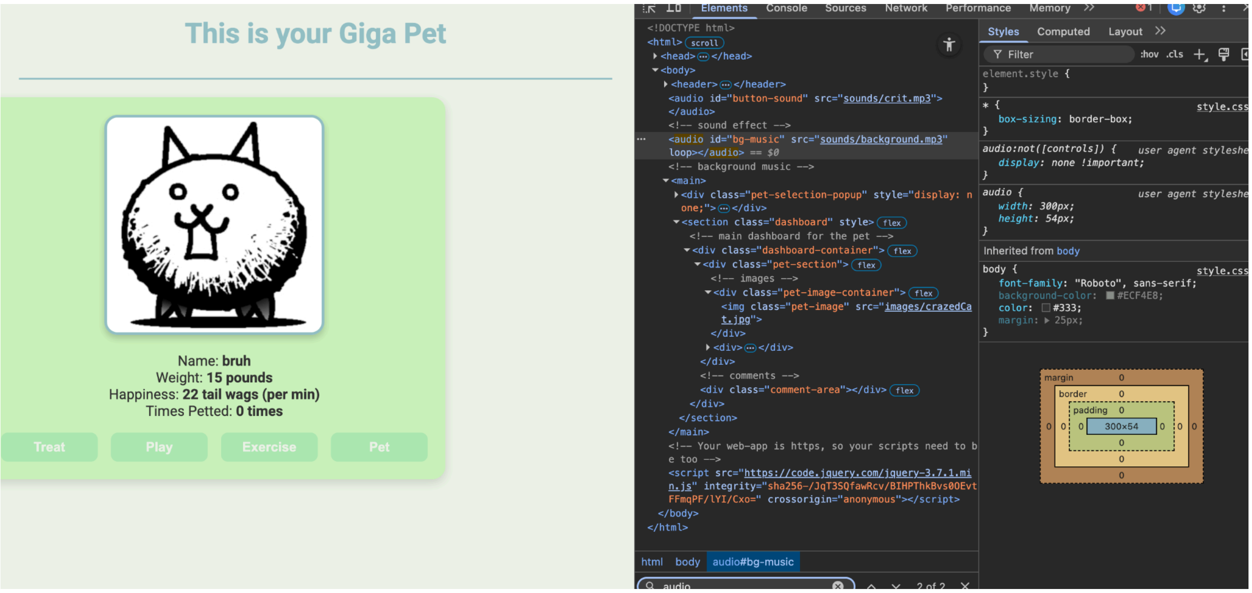
Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**
Name: **BIGCAT**

```
      <!-- images -->
      <div class="pet-image-container"> flex
       <img class="pet-image" src="images/cra  dCa
       t.jpg">
      </div>
      <div>
       <!-- stats -->
        <div> ⋯ </div>
        <div> ⋯ </div>
        <div> ⋯ </div>
        <div>
         "Name: "
         <strong> ⋯ </strong> == $0
        </div>
        <div class="button-container"> ⋯ </div>
        flex
      </div>
```

```
element.style {
}
* {                                   style.
   box-sizing: border-box;
}
strong {                   user agent style
   font-weight: bolder;
}
Inherited from section.dashboard
.dashboard {                          style.c
   display: flex;
   flex-direction: column;
   align-items: center;
   justify-content: center;
   text-align: center;
```

```
   spellcheck: true
 ▸ style: CSSStyleDeclaration {accentColor: '', additiveSymbols: '', alignContent: '', alignItems
   tabIndex: -1
   tagName: "SPAN"
   textContent: "bruh"
   title: ""
   translate: true
   virtualKeyboardPolicy: ""
   writingSuggestions: "true"
 ▸ [[Prototype]]: HTMLSpanElement
< undefined
> $1
< ▸ <strong> ⋯ </strong>
>
```

We can also enable rulers by going to Settings, Preferences, Elements, and clicking on Show rulers on hover. This helps us see the size of elements on the website.



This is your Giga Pet

Name: **bruh**
Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**

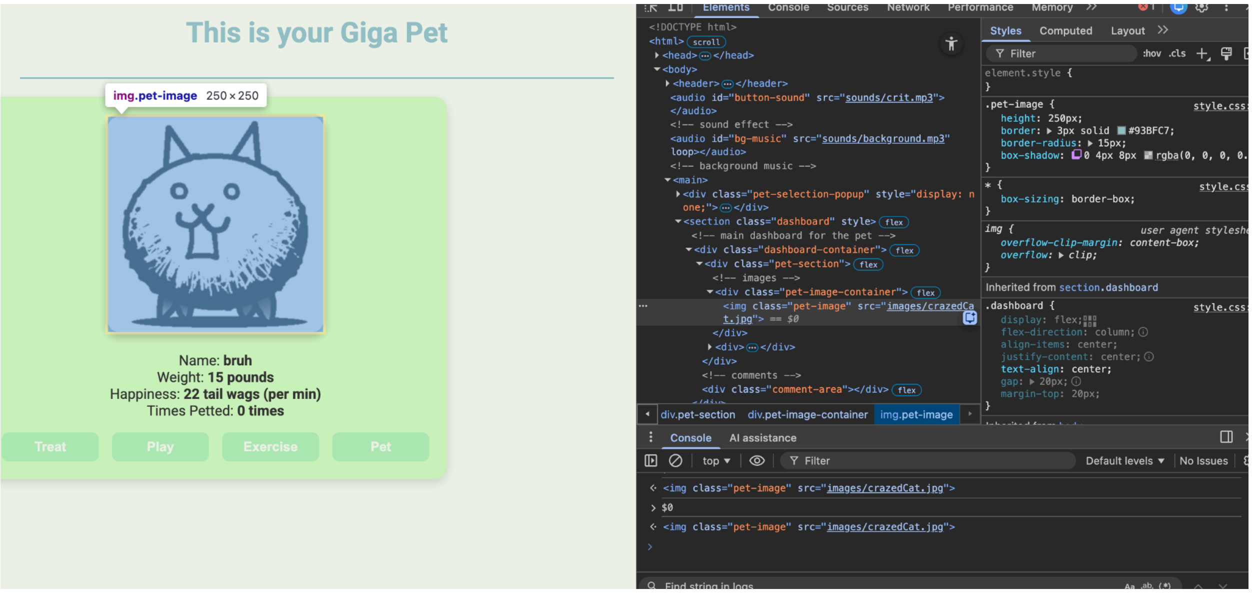
```
<!DOCTYPE html>
<html> scroll
  <head> ··· </head>
  <body>
    <header> ··· </header>
    <audio id="button-sound" src="sounds/crit.mp3">
    </audio>
    <!-- sound effect -->
    <audio id="bg-music" src="sounds/background.mp3"
    loop></audio> == $0
    <!-- background music -->
    <main>
      <div class="pet-selection-popup" style="display: n
      one;"> ··· </div>
      <section class="dashboard" style> flex
        <!-- main dashboard for the pet -->
        <div class="dashboard-container"> flex
          <div class="pet-section"> flex
            <!-- images -->
            <div class="pet-image-container"> flex
              <img class="pet-image" src="images/crazedCa
              t.jpg">
            </div>
            <div> ··· </div>
          </div>
          <!-- comments -->
          <div class="comment-area"></div> flex
        </div>
      </section>
    </main>
    <!-- Your web-app is https, so your scripts need to b
    e too -->
    <script src="https://code.jquery.com/jquery-3.7.1.mi
    n.js" integrity="sha256-/JqT3SQfawRcv/BIHPThkBvs0OEvt
    FFmqPF/lYI/Cxo=" crossorigin="anonymous"></script>
  </body>
</html>
```

html body audio#bg-music

audio 2 of 2

```
element.style {
}
* {                                   style.css
  box-sizing: border-box;
}
audio:not([controls]) {     user agent styleshe
  display: none !important;
}
audio {                     user agent styleshe
  width: 300px;
  height: 54px;
}
Inherited from body
body {                                style.css
  font-family: "Roboto", sans-serif;
  background-color: #ECF4E8;
  color: #333;
  margin: ▸ 25px;
}
```

I can also search through the website by command + F to search through nodes to find what I am looking for such as the audio.



This is your Giga Pet

img.pet-image 250 × 250

Name: **bruh**
Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**

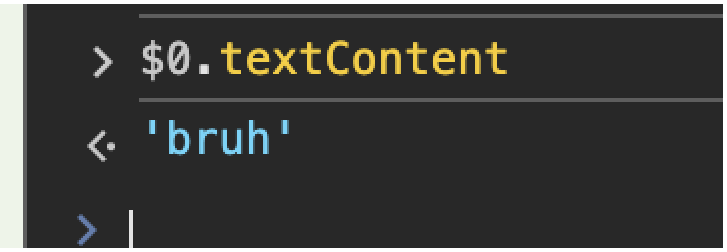
```
<!DOCTYPE html>
<html> scroll
  <head> ··· </head>
  <body>
    <header> ··· </header>
    <audio id="button-sound" src="sounds/crit.mp3">
    </audio>
    <!-- sound effect -->
    <audio id="bg-music" src="sounds/background.mp3"
    loop></audio>
    <!-- background music -->
    <main>
      <div class="pet-selection-popup" style="display: n
      one;"> ··· </div>
      <section class="dashboard" style> flex
        <!-- main dashboard for the pet -->
        <div class="dashboard-container"> flex
          <div class="pet-section"> flex
            <!-- images -->
            <div class="pet-image-container"> flex
              <img class="pet-image" src="images/crazedCa
              t.jpg"> == $0
            </div>
            <div> ··· </div>
          </div>
          <!-- comments -->
          <div class="comment-area"></div> flex
```

div.pet-section div.pet-image-container img.pet-image

```
element.style {
}
.pet-image {                          style.css
  height: 250px;
  border: ▸ 3px solid #93BFC7;
  border-radius: ▸ 15px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.
}
* {                                   style.css
  box-sizing: border-box;
}
img {                       user agent styleshe
  overflow-clip-margin: content-box;
  overflow: ▸ clip;
}
Inherited from section.dashboard
.dashboard {                          style.css
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  text-align: center;
  gap: ▸ 20px;
  margin-top: 20px;
}
```

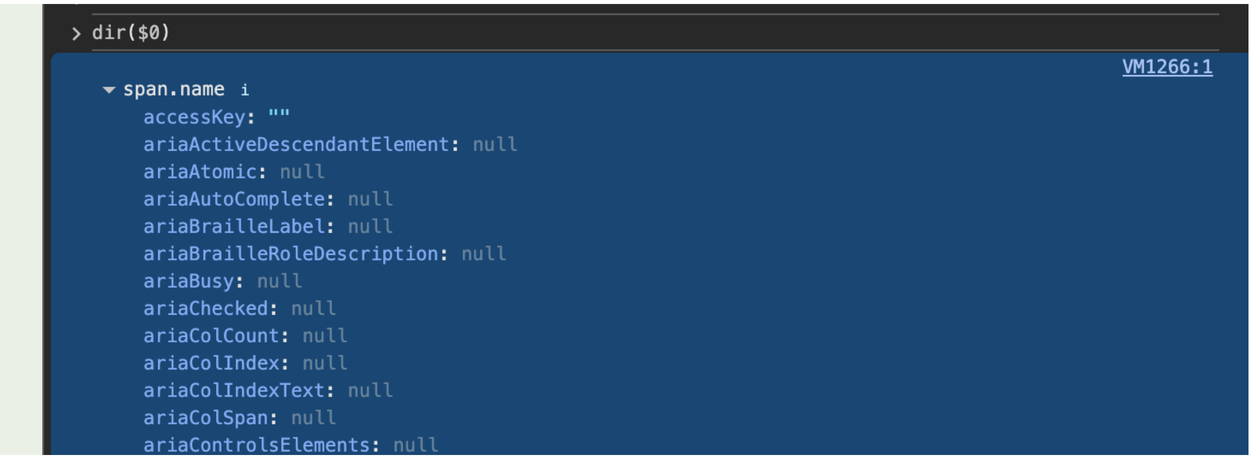
```
< <img class="pet-image" src="images/crazedCat.jpg">
> $0
< <img class="pet-image" src="images/crazedCat.jpg">
>
```

I can also use $0 to find the most recently selected element which was the image that I selected in the last picture.



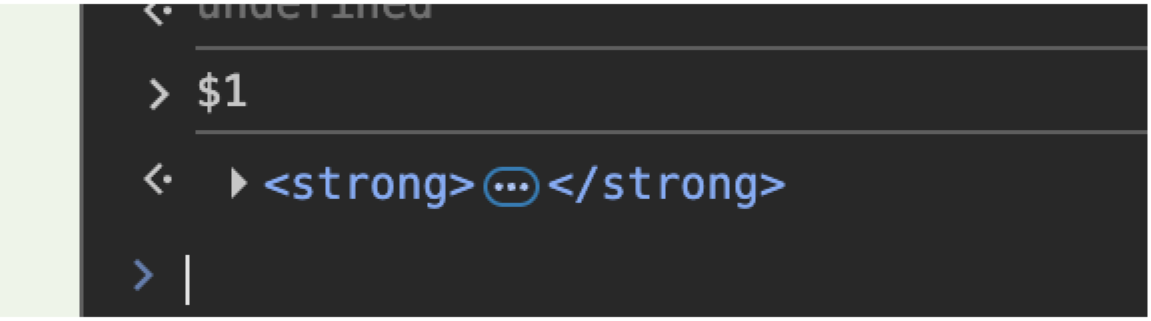
```
> $0.textContent
<· 'bruh'
>
```

I can also use $0.textContent to see information about the node. I chose the name of the cat and then typed in the console $0.textContent. This returned ‘bruh’ as the name is written as bruh and was the last thing that I selected.



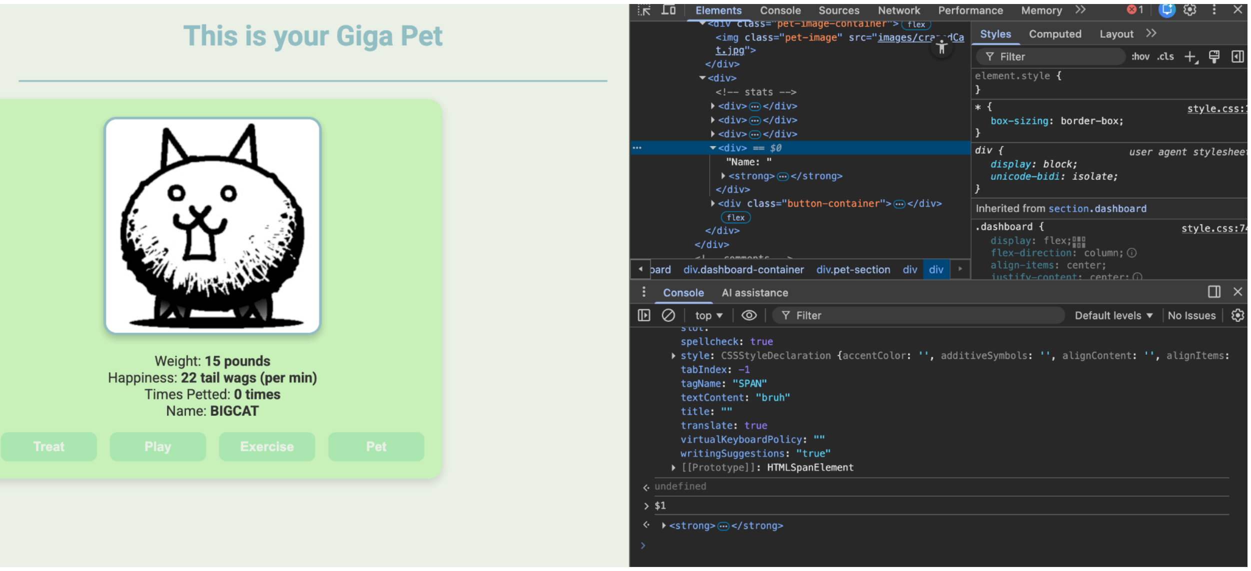
```
> dir($0)
                                                          VM1266:1
  ▾ span.name i
      accessKey: ""
      ariaActiveDescendantElement: null
      ariaAtomic: null
      ariaAutoComplete: null
      ariaBrailleLabel: null
      ariaBrailleRoleDescription: null
      ariaBusy: null
      ariaChecked: null
      ariaColCount: null
      ariaColIndex: null
      ariaColIndexText: null
      ariaColSpan: null
      ariaControlsElements: null
```

We can also use dir($0) to see the properties of the element which is in this case span.name. This is useful for learning about the attributes of the element.



```
<· undefined
> $1
<·  ▸ <strong>…</strong>
>
```

$1 is used to see the previous selected element. I used this in the console and got <strong><span class="name">BIGCAT</span></strong>



Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**
Name: **BIGCAT**

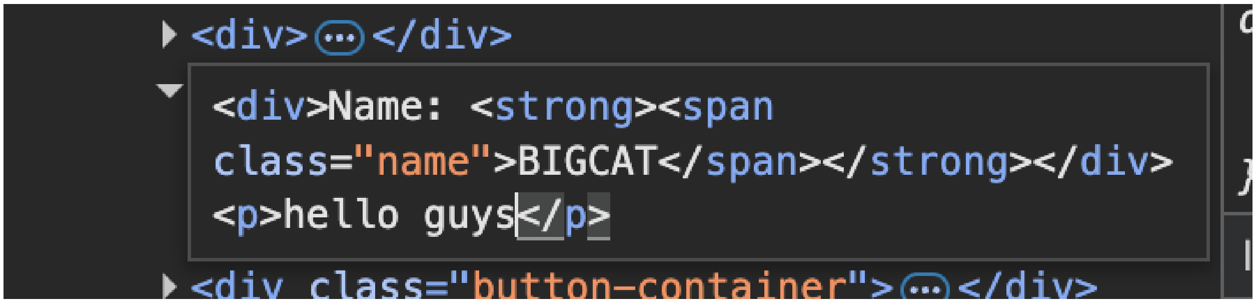
```
<img class="pet-image" src="images/cra··dCat.jpg">
</div>
<div>
  <!-- stats -->
  <div>…</div>
  <div>…</div>
  <div>…</div>
  <div> == $0
    "Name: "
    <strong>…</strong>
  </div>
  <div class="button-container">…</div> flex
</div>
</div>
```

```
element.style {
}
* {                                   style.css:
  box-sizing: border-box;
}
div {                                 user agent stylesheet
  display: block;
  unicode-bidi: isolate;
}
Inherited from section.dashboard
.dashboard {                          style.css:74
  display: flex;
  flex-direction: column;
  align-items: center;
```

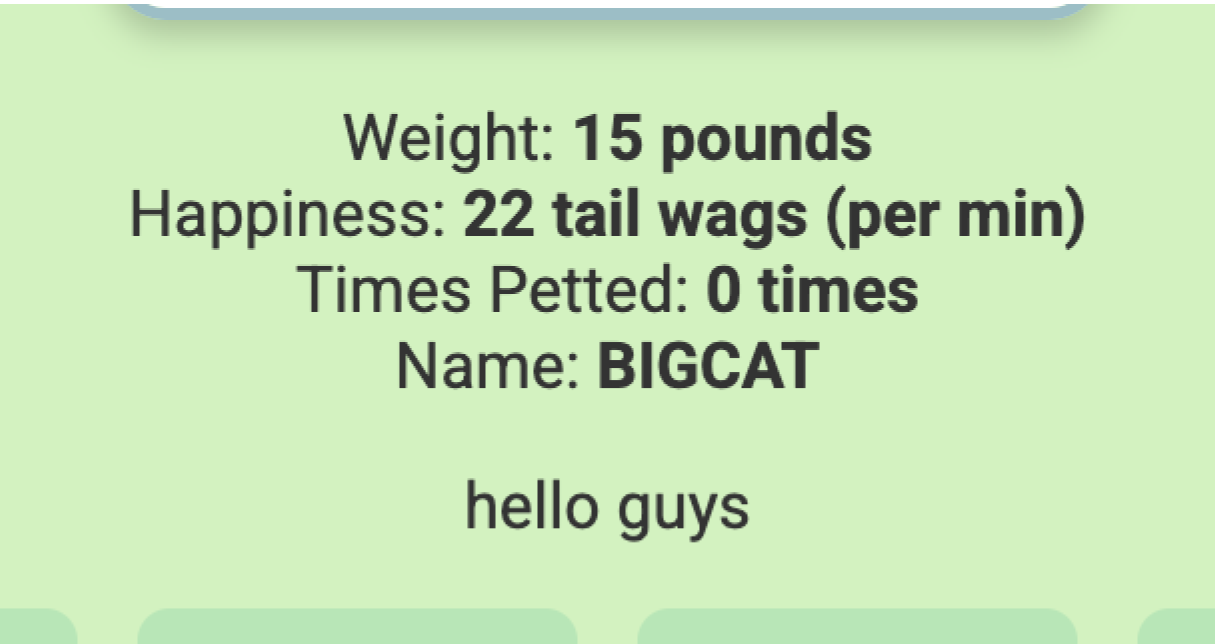
ɔard div.dashboard-container div.pet-section div div

```
  spellcheck: true
▸ style: CSSStyleDeclaration {accentColor: '', additiveSymbols: '', alignContent: '', alignItems:
  tabIndex: -1
  tagName: "SPAN"
  textContent: "bruh"
  title: ""
  translate: true
  virtualKeyboardPolicy: ""
  writingSuggestions: "true"
▸ [[Prototype]]: HTMLSpanElement
< undefined
> $1
< ▸ <strong>…</strong>
>
```

The last thing we can do in the DOM tree is to edit the elements directly. What I did in this screenshot was rename bruh to BIGCAT and then I moved it down to the bottom where it is below Times Petted. We are also able to change attributes, delete elements, etc.



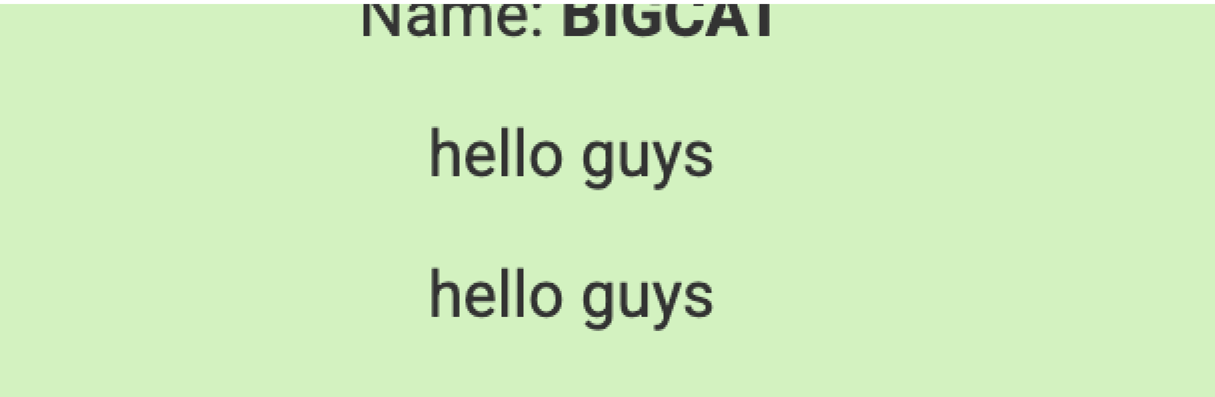
```
▸ <div>…</div>
▾ <div>Name: <strong><span
  class="name">BIGCAT</span></strong></div>
  <p>hello guys</p>
▸ <div class="button-container">…</div>
```



Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**
Name: **BIGCAT**

hello guys

We can also edit HTML and add different line of text such as <p> hello guys </p>

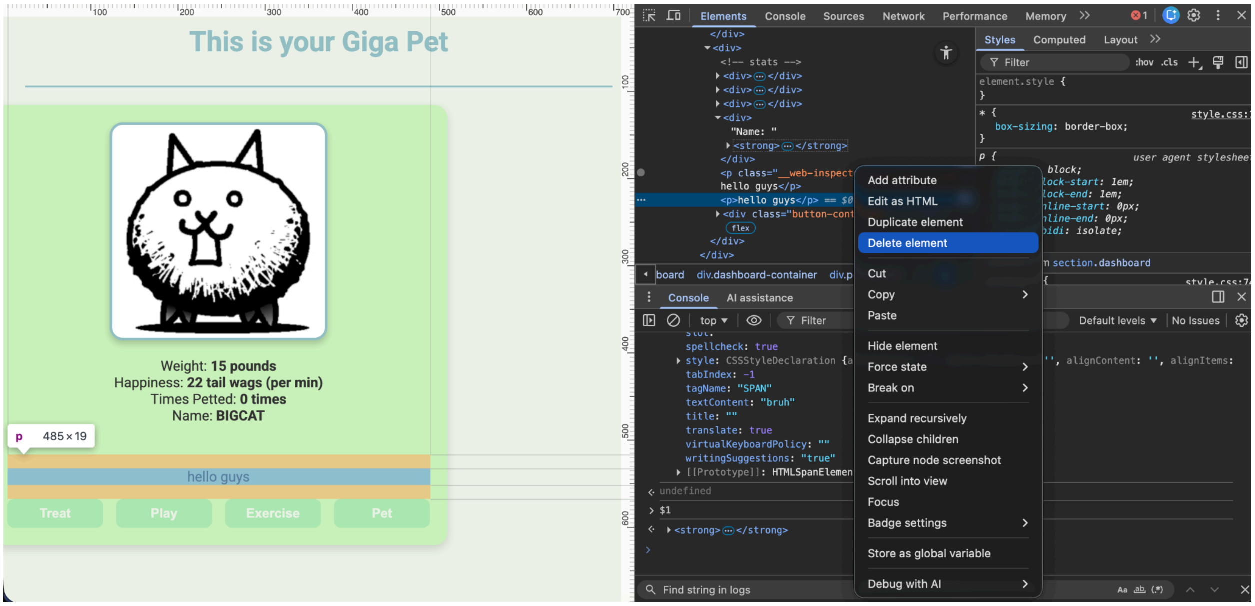


Name: **BIGCAT**

hello guys

hello guys

We can also duplicate by pressing Duplicate element



This is your Giga Pet

Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**
Name: **BIGCAT**

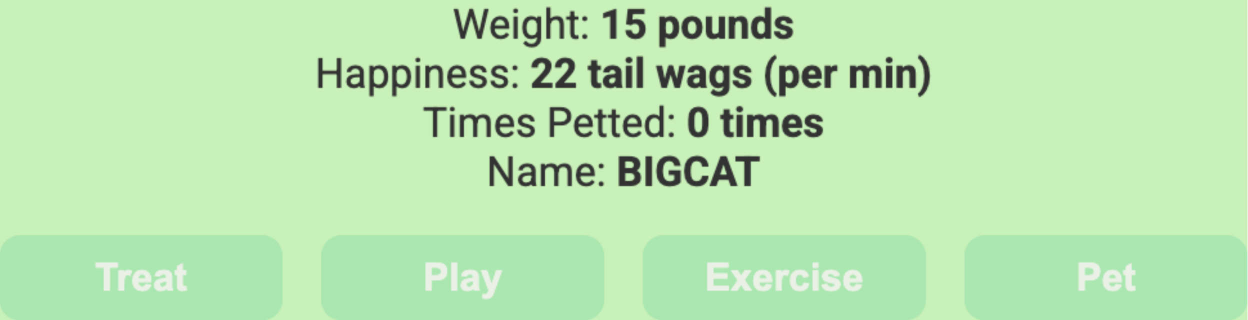
hello guys

```
    </div>
  ▾<div>
      <!-- stats -->
    ▸<div>…</div>
    ▸<div>…</div>
    ▸<div>…</div>
    ▾<div>
        "Name: "
      ▸<strong>…</strong>
      </div>
      <p class="__web-inspect
      hello guys</p>
…     <p>hello guys</p> == $0
    ▸<div class="button-cont
       flex
    </div>
  </div>
```

```
element.style {
}
* {                                style.css:
  box-sizing: border-box;
}
p {                      user agent styleshee
            block;
  lock-start: 1em;
  lock-end: 1em;
  nline-start: 0px;
  nline-end: 0px;
  bidi: isolate;
```

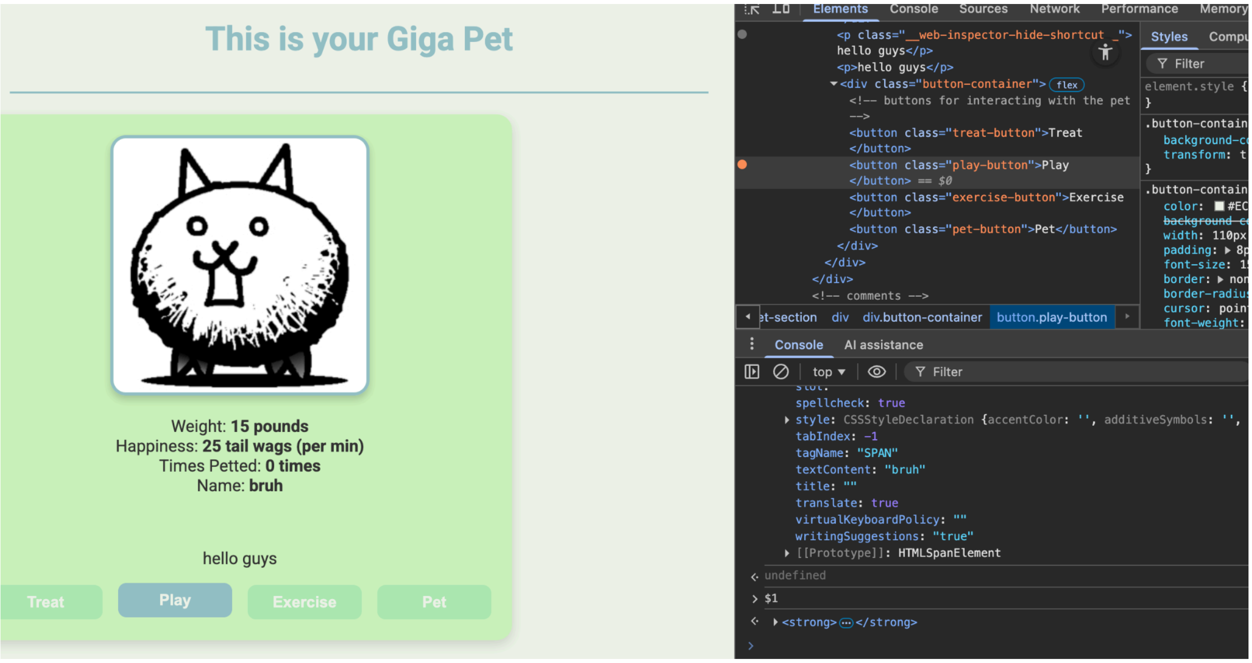
```
    spellcheck: true
  ▸ style: CSSStyleDeclaration {a  '', alignContent: '', alignItems:
    tabIndex: -1
    tagName: "SPAN"
    textContent: "bruh"
    title: ""
    translate: true
    virtualKeyboardPolicy: ""
    writingSuggestions: "true"
  ▸ [[Prototype]]: HTMLSpanElemen
< undefined
> $1
< ▸<strong>…</strong>
>
```

We can also hide or delete nodes.



Weight: **15 pounds**
Happiness: **22 tail wags (per min)**
Times Petted: **0 times**
Name: **BIGCAT**

We also have the ability to take screenshots of nodes directly from the DOM tree by right clicking and pressing on Capture node screenshot.



This is your Giga Pet

Weight: **15 pounds**
Happiness: **25 tail wags (per min)**
Times Petted: **0 times**
Name: **bruh**

hello guys

```
<p class="__web-inspector-hide-shortcut__">
hello guys</p>
<p>hello guys</p>
▾<div class="button-container"> flex
  <!-- buttons for interacting with the pet
  -->
  <button class="treat-button">Treat
  </button>
  <button class="play-button">Play
  </button> == $0
  <button class="exercise-button">Exercise
  </button>
  <button class="pet-button">Pet</button>
</div>
</div>
</div>
<!-- comments -->
```

```
element.style {
}
.button-contain
  background-c
  transform: t
}
.button-contain
  color: #EC
  ~~background-c~~
  width: 110px
  padding: ▸ 8p
  font-size: 1
  border: ▸ non
  border-radiu
  cursor: poin
  font-weight:
```

et-section div div.button-container button.play-button

```
    spellcheck: true
  ▸ style: CSSStyleDeclaration {accentColor: '', additiveSymbols: '',
    tabIndex: -1
    tagName: "SPAN"
    textContent: "bruh"
    title: ""
    translate: true
    virtualKeyboardPolicy: ""
    writingSuggestions: "true"
  ▸ [[Prototype]]: HTMLSpanElement
<· undefined
> $1
<· ▸ <strong>…</strong>
>
```

We can also force a state such as hover. I did it to the play button as shown here