

Dev Tools Implementations:

### **Console Logging -**

Implemented `Console.log` and `Console.table` stat during pet interactions

```
console.log("Action:", action);  
console.table(pet_info);
```

### **Logging Helper Function -**

Centralized helper that logs the state every time the player interacts with the pet.

```
function logPetState(action) {  
  console.log("Action:", action);  
  console.table(pet_info);  
}
```

### **Calling the Logging Function -**

Inserted inside each pet action handler (Treat, Play, Exercise, Nap).

```
logPetState("Treat button clicked");
```

### **Line-of-Code Breakpoint Target -**

This line is used to demonstrate how to pause execution in the Sources panel. In Chrome DevTools, clicking on the line number next to this call will pause program execution when it is reached, allowing inspection of all variables.

```
checkAndUpdatePetInfoInHtml(); // ← Set line breakpoint here in DevTools
```

### **Debugger Statement -**

Used to pause execution only when certain data conditions occur. This replicates the Conditional Breakpoints example from the DevTools JavaScript debugging guide.

```
if (pet_info.energy <= 1) {  
  debugger; // DevTools will pause here automatically  
}
```

### **DOM Mutation Breakpoint Target -**

This is the exact line that updates the pet's message.

Setting a DOM Subtree Modification Breakpoint on this element will pause when this runs.

Chrome DevTools will pause when `.html()` changes the element's content, which helps identify what code updates the DOM.

```
msg.html(message); // ← DOM subtree modification
```

### **DOM Editing Target (Elements Panel) -**

This element is used to demonstrate live DOM editing via the Elements panel. Editing this text in DevTools updates the page immediately without modifying the source code.

```
<span class="name">Michi</span>
```

### **CSS Editing Demonstration -**

This class is used in the Elements → Styles panel to demonstrate live CSS changes you can do by toggling or editing this in DevTools to demonstrate real-time adjustments.

```
.pet-message {  
  background: #fbeaff;  
}
```