

## Extending the Script: Adding “Add to Cart” Functionality

This document outlines how you might extend the stock checker script to automatically add an item to your cart once it’s in stock. **Please read the disclaimers carefully before proceeding, as automating purchases carries significant risks and ethical considerations.**

### Important Disclaimers and Considerations

- **Risks:** Automating “Add to Cart” can lead to unintended purchases, issues with your account (e.g., being flagged as a bot), or even legal issues if done improperly or maliciously.
- **Website Changes:** E-commerce websites frequently change their layouts and anti-bot measures. A script that works today might break tomorrow.
- **CAPTCHAs and Bot Detection:** Many sites employ CAPTCHAs, reCAPTCHAs, and other advanced bot detection mechanisms that are very difficult to bypass programmatically.
- **Account Security:** You would be automating actions on your personal account. Ensure you understand the security implications.
- **Ethical Use:** Use such automation responsibly and ethically. Do not engage in activities that could harm the website, other users, or violate terms of service.
- **No Checkout Automation:** This guide focuses only on adding to cart. Automating the entire checkout process (including payment and shipping information) is significantly more complex, highly risky, and not recommended due to the sensitive nature of the data involved.

### Prerequisites

Before attempting to add “Add to Cart” functionality, ensure your stock checker script is reliably identifying when the item is **in stock**.

### Steps to Implement “Add to Cart”

#### 1. Identify the “Add to Cart” Button Selector

Just like we did for the “Sold Out” button, you need to find the precise CSS selector for the “Add to Cart” button when the item is available. This is crucial.

- **Find an In-Stock Item:** Navigate to a product page on Best Buy (or any other site) where the “Add to Cart” button is visible and active.
- **Inspect Element:** Right-click on the “Add to Cart” button and select “Inspect” or “Inspect Element.”
- **Copy HTML/Attributes:** Identify unique attributes (like `id`, `data-*`, or specific `class` names) that reliably target *only* the “Add to Cart” button.

For example, it might be `button[data-button-state="ADD_TO_CART"]` or a more specific `button#addToCartBtn`.

## 2. Modify the `check_stock()` Function

Once you have the reliable selector, you can modify your `check_bestbuy_stock_playwright.py` script.

Locate the `if add_to_cart_present:` block. Inside this block, you will add the code to click the button.

*# ... inside the check\_stock() function, after stock determination*

```
if add_to_cart_present:
    print("Item is IN STOCK! Attempting to add to cart...")
    try:
        # Click the Add to Cart button
        # Replace 'YOUR_ADD_TO_CART_SELECTOR_HERE' with the actual selector you
        page.click('YOUR_ADD_TO_CART_SELECTOR_HERE')
        print("Successfully clicked Add to Cart!")

        # Optional: Wait for a confirmation message or navigate to cart page
        # Example: page.wait_for_selector('.add-to-cart-confirmation', timeout=
        # Example: page.goto('https://www.bestbuy.com/cart')

    except Exception as click_e:
        print(f"Failed to click Add to Cart button: {click_e}")

elif sold_out_present:
    print("Item is SOLD OUT.")
else:
    print("Could not determine stock status...")
```

*# ... rest of the script*

## 3. Handling Post-Click Actions (Optional but Recommended)

After clicking “Add to Cart,” you might want to:

- **Verify Success:** Look for a confirmation message (e.g., “Item added to cart,” a green checkmark) or a change in the button’s text.
- **Navigate to Cart:** Automatically go to the shopping cart page to confirm the item is there.
- **Take a Screenshot:** Capture a screenshot (`page.screenshot(path='cart_added.png')`) for debugging or confirmation.

#### 4. Error Handling and Robustness

- **Explicit Waits:** Always use `page.wait_for_selector()` before `page.click()` to ensure the element is present and clickable.
- **Try-Except Blocks:** Wrap your click actions in `try-except` blocks to catch errors if the button isn't found or clickable.
- **Retries:** For critical actions, consider implementing a retry mechanism with short delays.

#### 5. Running the Script

When testing “Add to Cart” functionality, it’s highly recommended to run the script in **headful mode** (`headless=False`) so you can visually observe what’s happening in the browser. This will help you debug any issues with selectors or timing.

Remember to revert to `headless=True` for automated background runs once you are confident in the script’s behavior.