CSD-2550 Final report

Ekin Pehlivan, Arshad Musejee, Kristen Rebello

Main Website

In this final report, I will be talking about the e-commerce website that was created. The product we chose to sell was DVD's. The website was created in accordance with the requirements e.g. customer registration, login, customer tracking and so on. Also, personal data would be stored on the server using MongoDB database. MongoDB will store the customer details, orders, products and so on. The reason these things are stored on a database is because it is persistent data, so whenever they are required they can be found easily, for example, if a customer order needs to be checked upon. Also, there will be a content management system (CMS), which is important because this will be used in order to let staff add new DVDs to the website, view orders, change product information and so on. There are a variety of pages such as homepage, login/register page, genres, checkout etc.

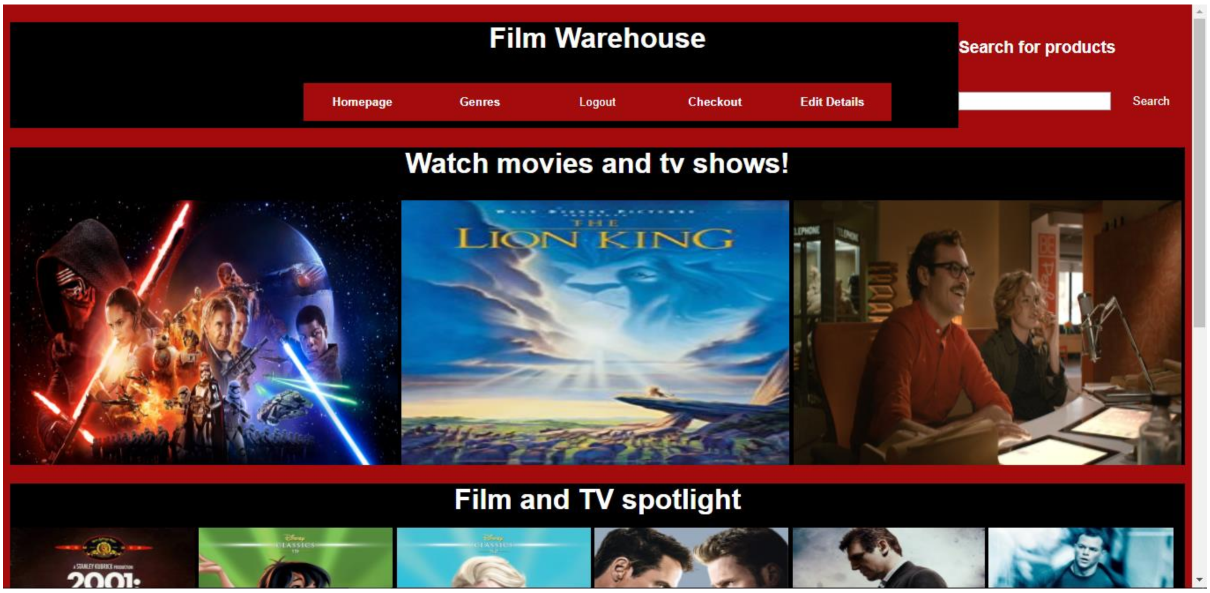
Mongo DB

Our database in MongoDB is named "ecommerce" and has the following collections:

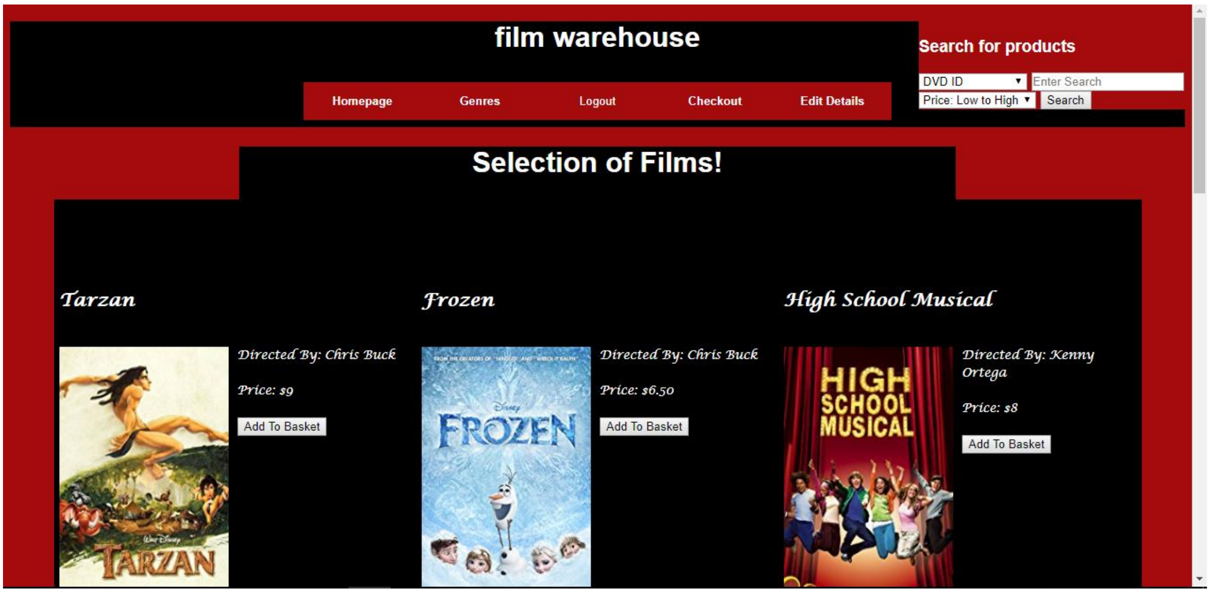
- customers
- orders
- products
- staff

Ecommerce website

The website will have various pages which the user is able to access via the navigation bar which was added to the site. The first page that the user will see when they enter the website is the homepage. What they will see is the name of the website along with a few movies/TV shows and the film and TV spotlight. There will also be a recommendations area which will show a couple of movies and names of movies which have been searched in case a customer forgets what they searched for and want to go back to it. The search recommendation will be stored using HTML local/session storage. Also, the footer will contain various social network links so a customer can find some reviews, get more information or leave comments about the website.



Another page on the website is the genres page. This page will have similar layout to the homepage minus the search and recommendations. It will have different genres such as children films, science fiction films and action films. Also, it will contain the footer with the social media links. The reason there is a genres page is because a customer could be looking for a specific type of film e.g. action movies therefore having a genres page will be beneficial to them as they will be able to view all the action movies.



film warehouse

Homepage Genres Logout Checkout Edit Details

Search for products

DVD ID ▾ Enter Search

Price: Low to High ▾ Search

Selection of Films!

Tarzan

Directed By: Chris Buck

Price: $9

Add To Basket

Frozen

Directed By: Chris Buck

Price: $6.50

Add To Basket

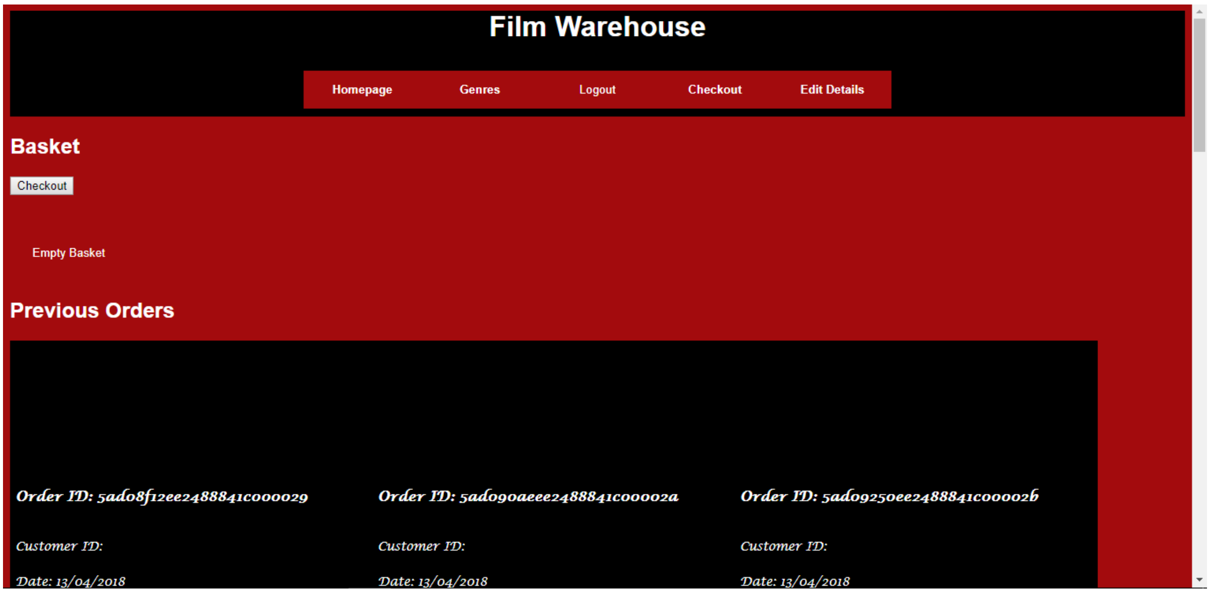
High School Musical

Directed By: Kenny Ortega

Price: $8

Add To Basket

On the website there is a log in/sign up page. This page is important because in order to purchase products from the website users will require an account. Therefore, they need to enter their details on the page and create an account. Once they have signed up they will be able to log in and make purchases without having to enter their details repeatedly. This page will also contain film and TV spotlight along with the footer containing the social media links. The customer data which is used to create accounts will be stored on the server using MongoDB as it is persistent data.



Film Warehouse

Homepage Genres Logout Checkout Edit Details

Basket

Checkout

Empty Basket

Previous Orders

Order ID: 5ad08f12ee2488841c000029

Customer ID:

Date: 13/04/2018

Order ID: 5ad090aeee2488841c00002a

Customer ID:

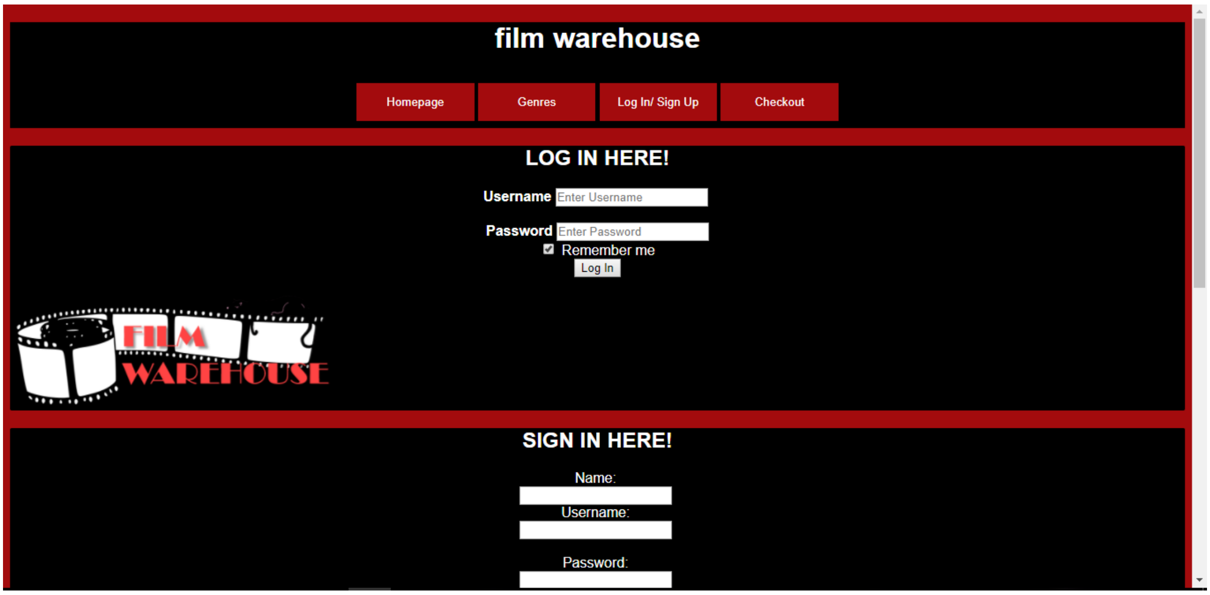
Date: 13/04/2018

Order ID: 5ad09250ee2488841c00002b

Customer ID:

Date: 13/04/2018

Another page is the checkout page. The checkout page is an essential part of the website due to the fact that it is needed in order to carry out the purchase. When the customer adds their selected items it will go to the basket where they will choose the quantity of the product. Once they are happy they will then be able to proceed and they will reach the checkout page. From here they will be able to confirm the purchase where they will have to enter their login details if they haven't already and then pay for them.



LOG IN HERE!

Username Enter Username

Password Enter Password

☑ Remember me

SIGN IN HERE!

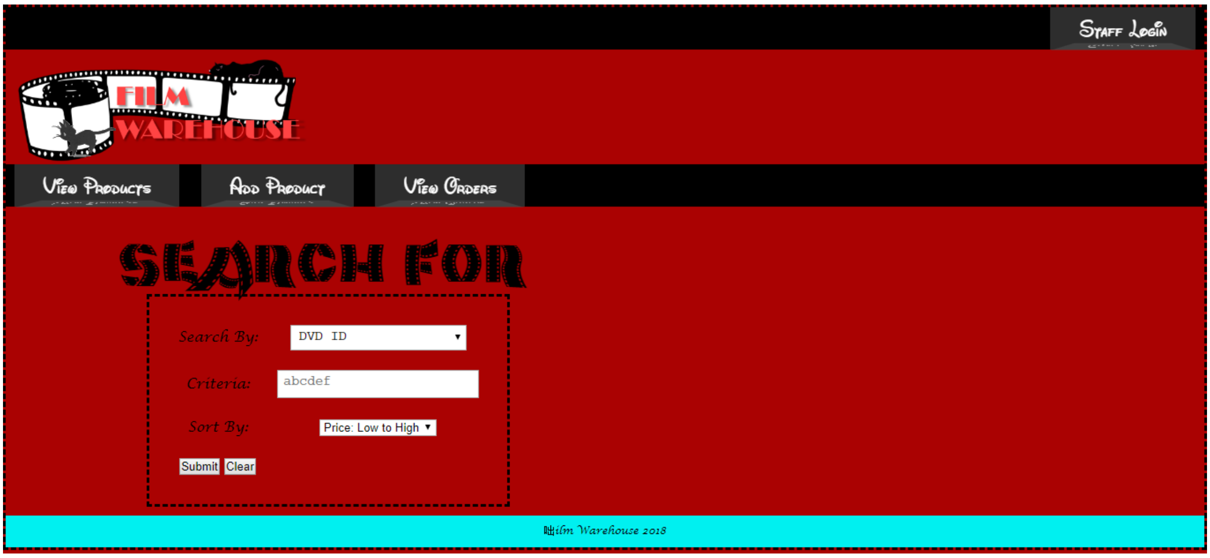
Name:

Username:

Password:

CMS

The e-commerce website has a content management system (CMS). The reason for this is that the staff are able to add products, view orders, remove products and more. Also, there is AJAX which shows a couple of recommendations to the customers.



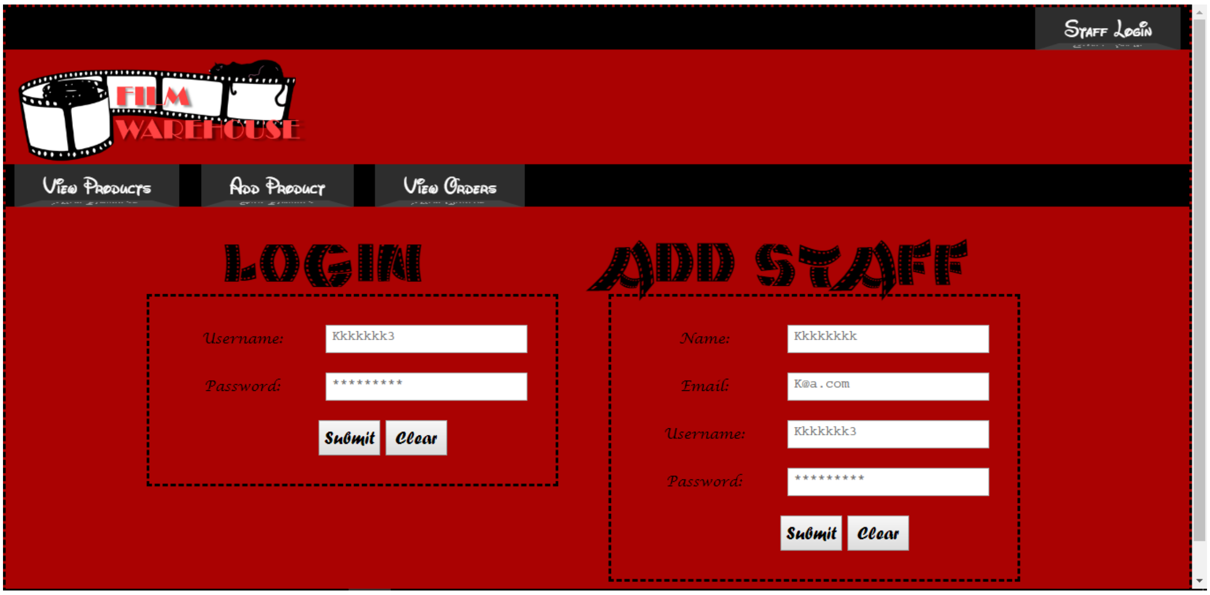
SEARCH FOR

Search By: DVD ID

Criteria: abcdef

Sort By: Price: Low to High

The above is a View Products page, where, logged in, you can view all the products that are stored in the database. It allows you to search by various keys and allows for sorting according to price and name.



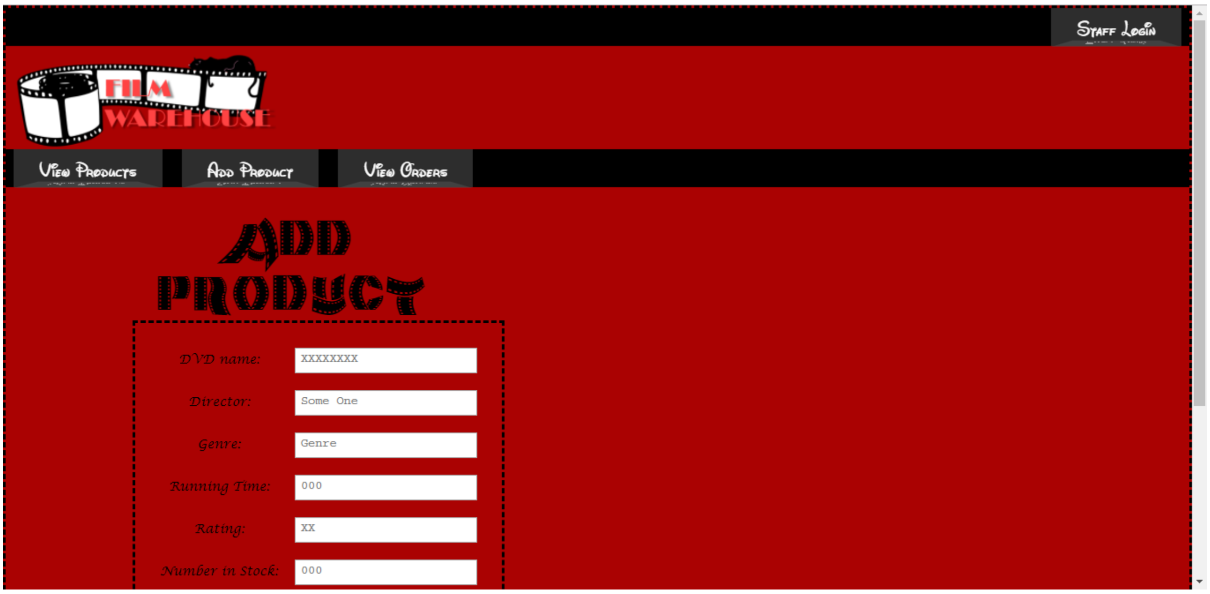
LOGIN

Username: Kkkkkkk3
Password: *********

ADD STAFF

Name: Kkkkkkkk
Email: K@a.com
Username: Kkkkkkk3
Password: *********

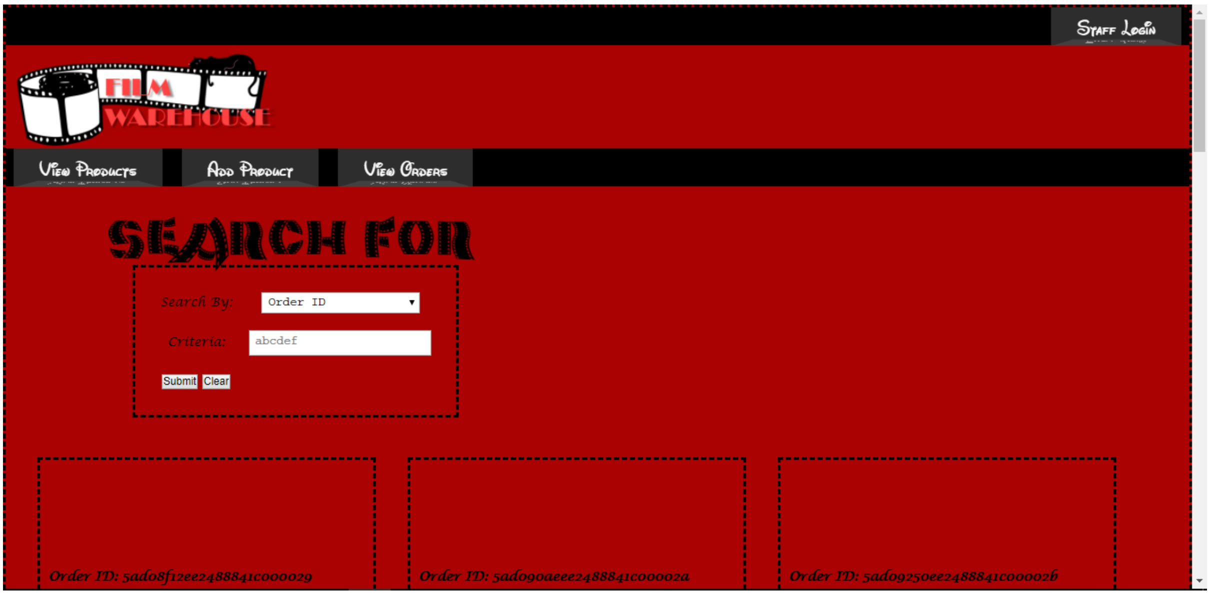
This is the login page, where you can add staff to the database, which is not the best way to go, in a real world situation. There should be an admin that is allowed to add staff once logged in, but this is a basic project, and if allowed time, that could've been implemented.



ADD PRODUCT

DVD name: xxxxxxxx
Director: Some One
Genre: Genre
Running Time: 000
Rating: xx
Number in Stock: 000

The above is a page for adding products to the database. Once you are logged in, you can successfully add products.



Search By: Order ID

Criteria: abcdef

Order ID: 5ad08f12ee2488841c000029

Order ID: 5ad090aeee2488841c00002a

Order ID: 5ad0925oee2488841c00002b

Here is where you can view and update customer orders. It would mostly be useful to update the status of the order.

Third Party libraries

Below is the third party library which was used for the footer.

https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css

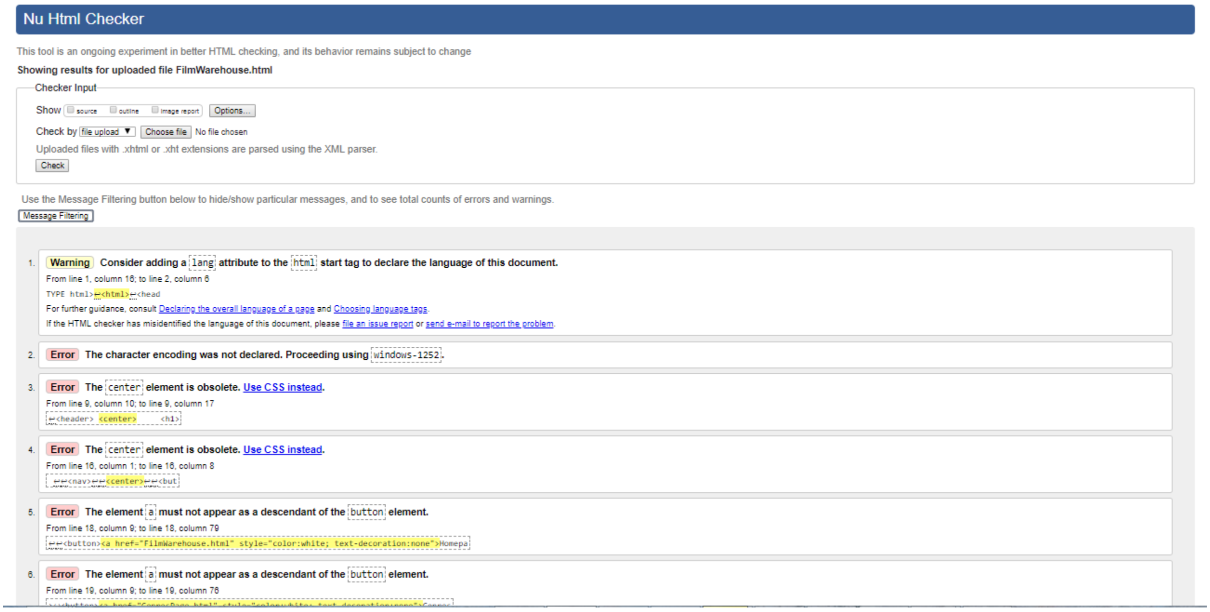
Contributions

The contributions made by each member of the group is provided in the Contribution Sheet which is submitted along with the website.

Testing

HTML and CSS validation

Index:



This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for uploaded file FilmWarehouse.html

Checker Input

Show □ source □ outline □ image report [Options...]

Check by [file upload ▼] [Choose file] No file chosen

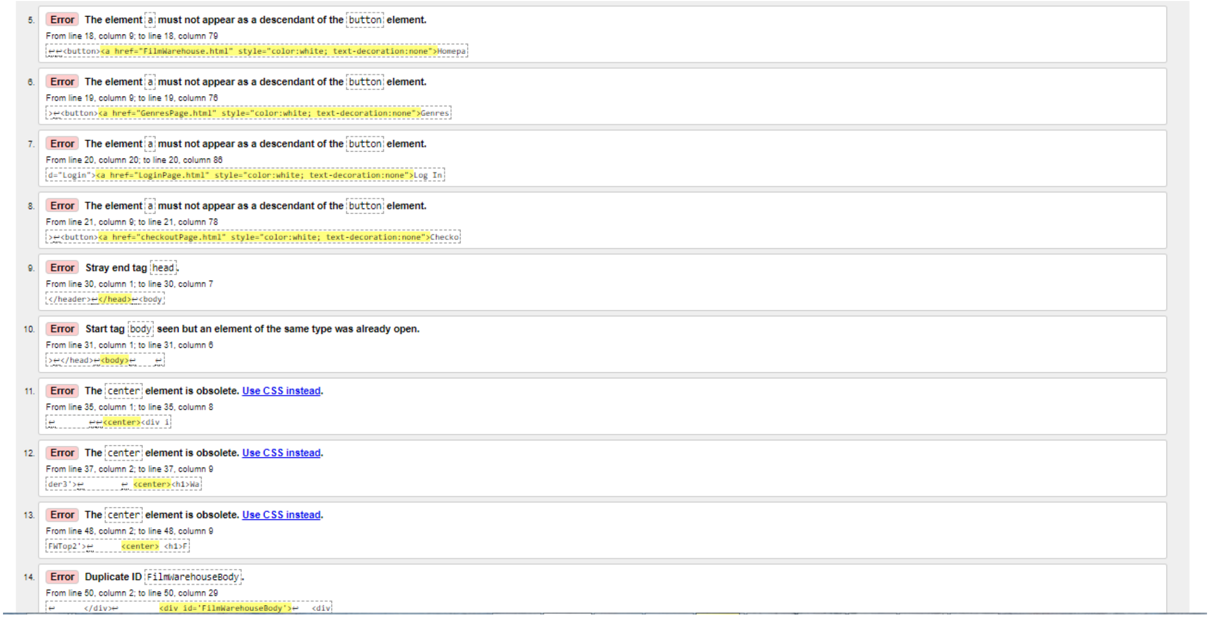
Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

[Check]

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

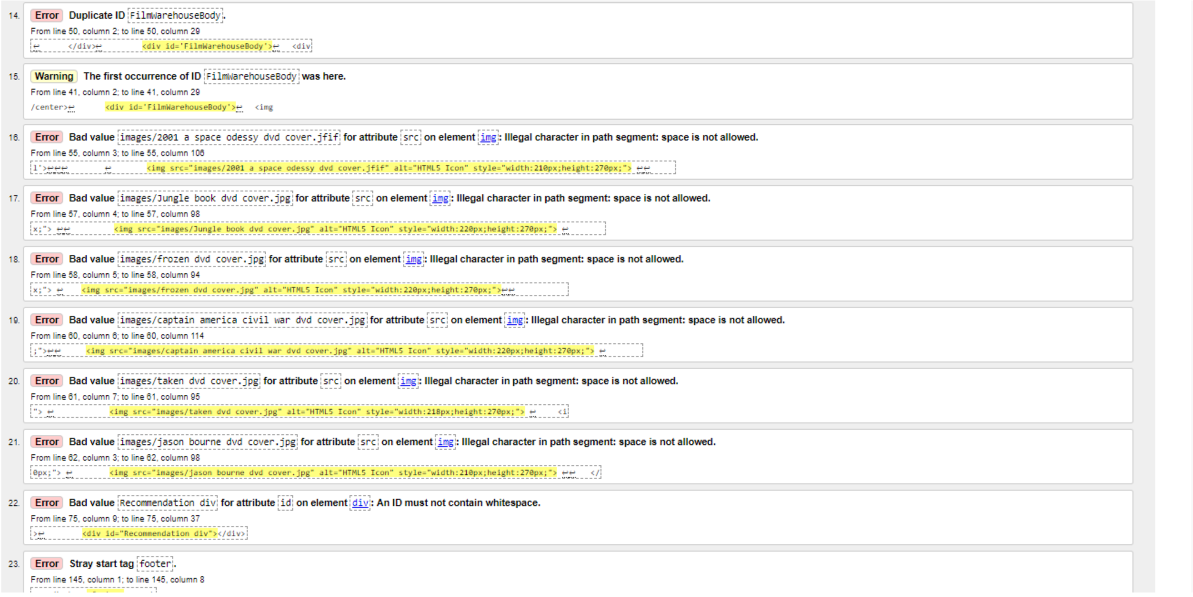
[Message Filtering]

1. **Warning** **Consider adding a `lang` attribute to the `html` start tag to declare the language of this document.**
   From line 1, column 16; to line 2, column 6
   `TYPE html>↩<html>↩<head`
   For further guidance, consult Declaring the overall language of a page and Choosing language tags.
   If the HTML checker has misidentified the language of this document, please file an issue report or send e-mail to report the problem.
2. **Error** **The character encoding was not declared. Proceeding using `windows-1252`.**
3. **Error** **The `center` element is obsolete. Use CSS instead.**
   From line 9, column 10; to line 9, column 17
   `↩<header> <center>     <h1>`
4. **Error** **The `center` element is obsolete. Use CSS instead.**
   From line 16, column 1; to line 16, column 8
   ` ↩↩<nav>↩↩<center>↩↩<but`
5. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 18, column 9; to line 18, column 79
   `↩↩<button><a href="FilmWarehouse.html" style="color:white; text-decoration:none">Homepa`
6. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 19, column 9; to line 19, column 76



5. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 18, column 9; to line 18, column 79
   `↩↩<button><a href="FilmWarehouse.html" style="color:white; text-decoration:none">Homepa`
6. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 19, column 9; to line 19, column 76

   `>↩<button><a href="GenresPage.html" style="color:white; text-decoration:none">Genres`
7. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 20, column 20; to line 20, column 88
   `d="Login"><a href="LoginPage.html" style="color:white; text-decoration:none">Log In`
8. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 21, column 9; to line 21, column 78
   `>↩<button><a href="checkoutPage.html" style="color:white; text-decoration:none">Checko`
9. **Error** **Stray end tag `head`.**
   From line 30, column 1; to line 30, column 7
   `</header>↩</head>↩<body`
10. **Error** **Start tag `body` seen but an element of the same type was already open.**
    From line 31, column 1; to line 31, column 6
    `>↩</head>↩<body>↩     ↩`
11. **Error** **The `center` element is obsolete. Use CSS instead.**
    From line 35, column 1; to line 35, column 8
    `↩       ↩↩<center><div i`
12. **Error** **The `center` element is obsolete. Use CSS instead.**
    From line 37, column 2; to line 37, column 9
    `der3'>↩        ↩ <center><h1>Wa`
13. **Error** **The `center` element is obsolete. Use CSS instead.**
    From line 48, column 2; to line 48, column 9
    `FWTop2'>↩      <center> <h1>F`
14. **Error** **Duplicate ID `FilmWarehouseBody`.**
    From line 50, column 2; to line 50, column 29
    `↩      </div>↩         <div id='FilmWarehouseBody'>↩   <div`



14. Error Duplicate ID FilmWarehouseBody.

From line 50, column 2; to line 50, column 29

```
↩      </div>↩          <div id='FilmWarehouseBody'>↩   <div
```

15. Warning The first occurrence of ID FilmWarehouseBody was here.

From line 41, column 2; to line 41, column 29

```
/center>↩        <div id='FilmWarehouseBody'>↩   <img
```

16. Error Bad value images/2001 a space odessy dvd cover.jfif for attribute src on element img: Illegal character in path segment: space is not allowed.

From line 55, column 3; to line 55, column 106

```
1'>↩↩↩          ↩        <img src="images/2001 a space odessy dvd cover.jfif" alt="HTML5 Icon" style="width:210px;height:270px;"> ↩↩
```

17. Error Bad value images/Jungle book dvd cover.jpg for attribute src on element img: Illegal character in path segment: space is not allowed.

From line 57, column 4; to line 57, column 98

```
x;"> ↩↩          <img src="images/Jungle book dvd cover.jpg" alt="HTML5 Icon" style="width:220px;height:270px;"> ↩
```

18. Error Bad value images/frozen dvd cover.jpg for attribute src on element img: Illegal character in path segment: space is not allowed.

From line 58, column 5; to line 58, column 94

```
x;"> ↩    <img src="images/frozen dvd cover.jpg" alt="HTML5 Icon" style="width:220px;height:270px;">↩↩
```

19. Error Bad value images/captain america civil war dvd cover.jpg for attribute src on element img: Illegal character in path segment: space is not allowed.

From line 60, column 6; to line 60, column 114

```
;">↩↩     <img src="images/captain america civil war dvd cover.jpg" alt="HTML5 Icon" style="width:220px;height:270px;"> ↩
```

20. Error Bad value images/taken dvd cover.jpg for attribute src on element img: Illegal character in path segment: space is not allowed.

From line 61, column 7; to line 61, column 95

```
"> ↩           <img src="images/taken dvd cover.jpg" alt="HTML5 Icon" style="width:218px;height:270px;"> ↩    <i
```

21. Error Bad value images/jason bourne dvd cover.jpg for attribute src on element img: Illegal character in path segment: space is not allowed.

From line 62, column 3; to line 62, column 98

```
0px;"> ↩        <img src="images/jason bourne dvd cover.jpg" alt="HTML5 Icon" style="width:210px;height:270px;"> ↩↩   </
```

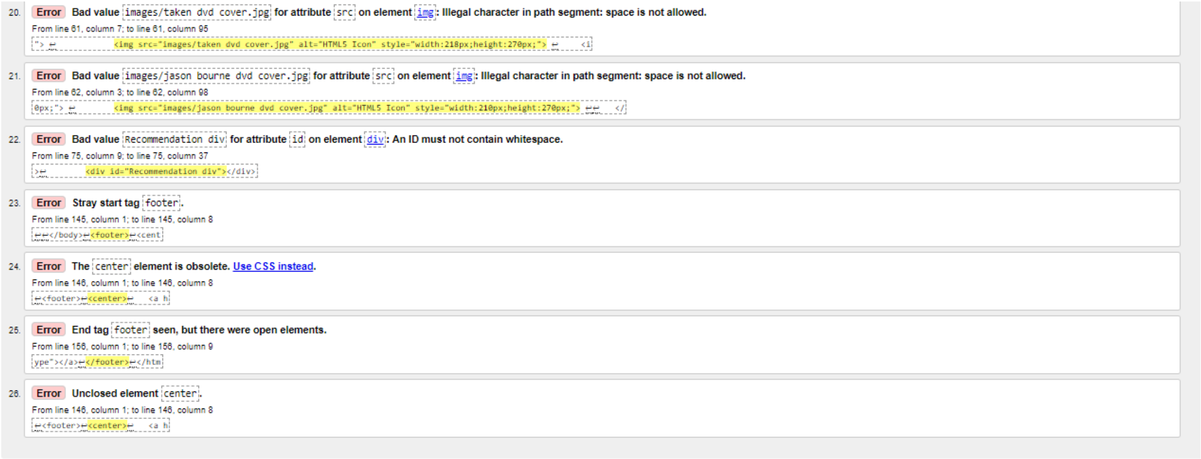
22. Error Bad value Recommendation div for attribute id on element div: An ID must not contain whitespace.

From line 75, column 9; to line 75, column 37

```
>↩        <div id="Recommendation div"></div>
```

23. Error Stray start tag footer.

From line 145, column 1; to line 145, column 8



20. Error Bad value images/taken dvd cover.jpg for attribute src on element img: Illegal character in path segment: space is not allowed.

From line 61, column 7; to line 61, column 95

```
"> ↩           <img src="images/taken dvd cover.jpg" alt="HTML5 Icon" style="width:218px;height:270px;"> ↩    <i
```

21. Error Bad value images/jason bourne dvd cover.jpg for attribute src on element img: Illegal character in path segment: space is not allowed.

From line 62, column 3; to line 62, column 98

```
0px;"> ↩        <img src="images/jason bourne dvd cover.jpg" alt="HTML5 Icon" style="width:210px;height:270px;"> ↩↩   </
```

22. Error Bad value Recommendation div for attribute id on element div: An ID must not contain whitespace.

From line 75, column 9; to line 75, column 37

```
>↩        <div id="Recommendation div"></div>
```

23. Error Stray start tag footer.

From line 145, column 1; to line 145, column 8

```
↩↩</body>↩<footer>↩<cent
```

24. Error The center element is obsolete. Use CSS instead.

From line 146, column 1; to line 146, column 8

```
↩<footer>↩<center>↩   <a h
```

25. Error End tag footer seen, but there were open elements.

From line 156, column 1; to line 156, column 9

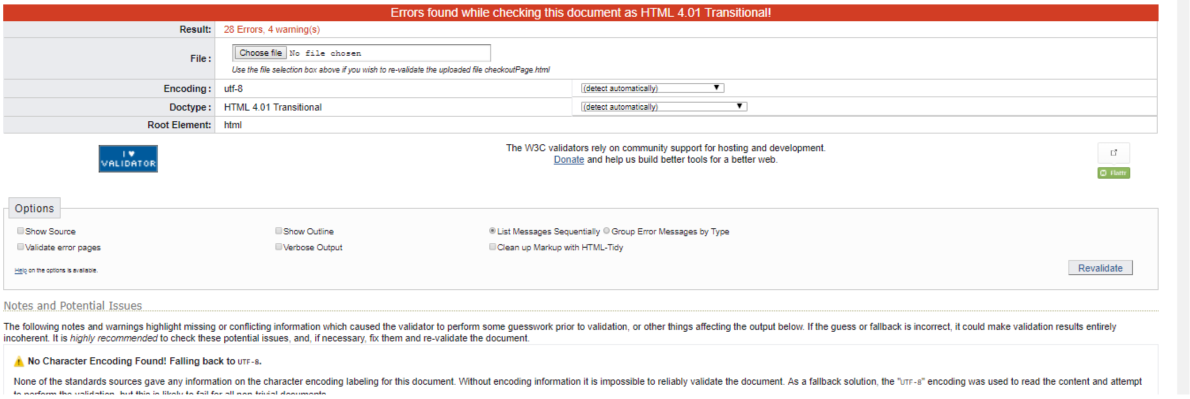
```
ype"></a>↩</footer>↩</htm
```

26. Error Unclosed element center.

From line 146, column 1; to line 146, column 8

```
↩<footer>↩<center>↩   <a h
```

Check out page:



Errors found while checking this document as HTML 4.01 Transitional!

Result:	28 Errors, 4 warning(s)	
File :	Choose file No file chosen *Use the file selection box above if you wish to re-validate the uploaded file checkoutPage.html*	
Encoding :	utf-8	(detect automatically) ▼
Doctype :	HTML 4.01 Transitional	(detect automatically) ▼
Root Element:	html	

The W3C validators rely on community support for hosting and development.
Donate and help us build better tools for a better web.

Options

□ Show Source
□ Validate error pages
□ Show Outline
□ Verbose Output
● List Messages Sequentially ○ Group Error Messages by Type
□ Clean up Markup with HTML-Tidy

Help on the options is available.

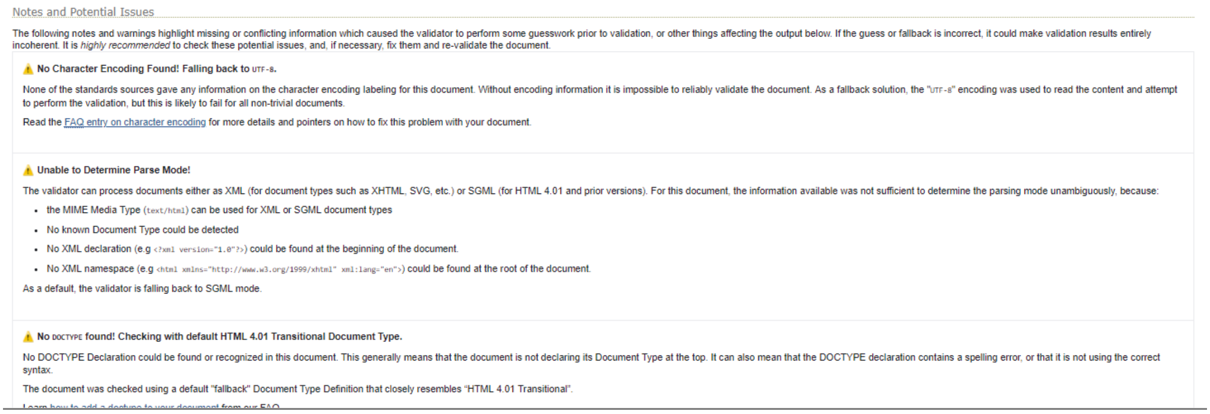
Revalidate

Notes and Potential Issues

The following notes and warnings highlight missing or conflicting information which caused the validator to perform some guesswork prior to validation, or other things affecting the output below. If the guess or fallback is incorrect, it could make validation results entirely incoherent. It is *highly recommended* to check these potential issues, and, if necessary, fix them and re-validate the document.

⚠ **No Character Encoding Found! Falling back to UTF-8.**

None of the standards sources gave any information on the character encoding labeling for this document. Without encoding information it is impossible to reliably validate the document. As a fallback solution, the "UTF-8" encoding was used to read the content and attempt to perform the validation, but this is likely to fail for all non-trivial documents.



Notes and Potential Issues

The following notes and warnings highlight missing or conflicting information which caused the validator to perform some guesswork prior to validation, or other things affecting the output below. If the guess or fallback is incorrect, it could make validation results entirely incoherent. It is *highly recommended* to check these potential issues, and, if necessary, fix them and re-validate the document.

⚠ **No Character Encoding Found! Falling back to `UTF-8`.**

None of the standards sources gave any information on the character encoding labeling for this document. Without encoding information it is impossible to reliably validate the document. As a fallback solution, the "`UTF-8`" encoding was used to read the content and attempt to perform the validation, but this is likely to fail for all non-trivial documents.

Read the FAQ entry on character encoding for more details and pointers on how to fix this problem with your document.

⚠ **Unable to Determine Parse Mode!**

The validator can process documents either as XML (for document types such as XHTML, SVG, etc.) or SGML (for HTML 4.01 and prior versions). For this document, the information available was not sufficient to determine the parsing mode unambiguously, because:

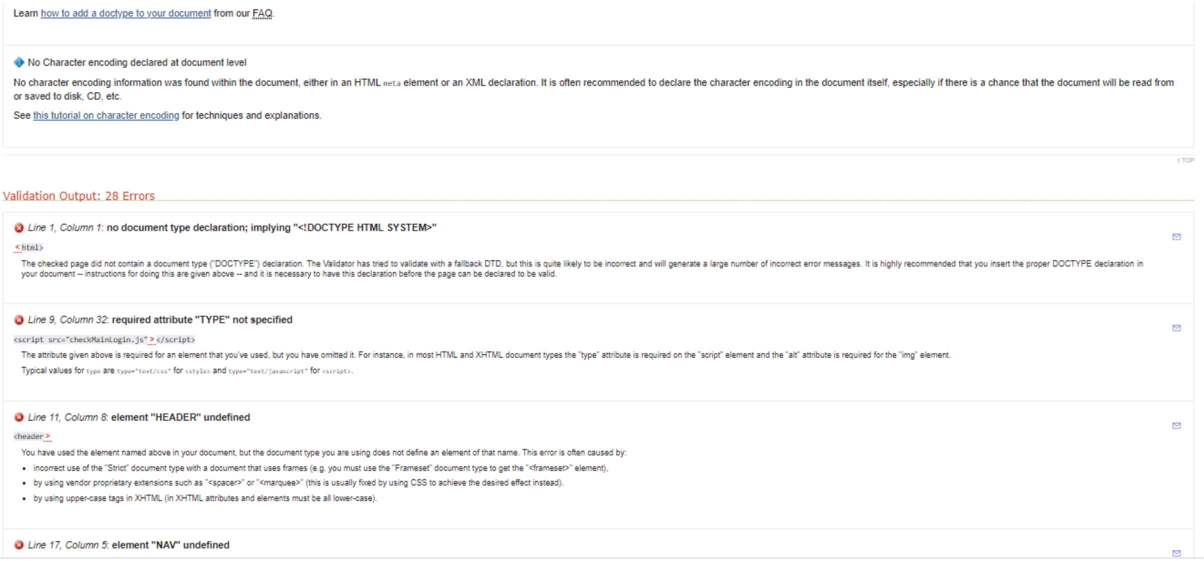
- the MIME Media Type (`text/html`) can be used for XML or SGML document types
- No known Document Type could be detected
- No XML declaration (e.g `<?xml version="1.0"?>`) could be found at the beginning of the document.
- No XML namespace (e.g `<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">`) could be found at the root of the document.

As a default, the validator is falling back to SGML mode.

⚠ **No `DOCTYPE` found! Checking with default HTML 4.01 Transitional Document Type.**

No DOCTYPE Declaration could be found or recognized in this document. This generally means that the document is not declaring its Document Type at the top. It can also mean that the DOCTYPE declaration contains a spelling error, or that it is not using the correct syntax.

The document was checked using a default "fallback" Document Type Definition that closely resembles "HTML 4.01 Transitional".



Learn how to add a doctype to your document from our FAQ.

🔷 No Character encoding declared at document level

No character encoding information was found within the document, either in an HTML `meta` element or an XML declaration. It is often recommended to declare the character encoding in the document itself, especially if there is a chance that the document will be read from or saved to disk, CD, etc.

See this tutorial on character encoding for techniques and explanations.

↑ TOP

Validation Output: 28 Errors

❌ *Line 1, Column 1*: **no document type declaration; implying "<!DOCTYPE HTML SYSTEM>"**

```
<html>
```

The checked page did not contain a document type ("DOCTYPE") declaration. The Validator has tried to validate with a fallback DTD, but this is quite likely to be incorrect and will generate a large number of incorrect error messages. It is highly recommended that you insert the proper DOCTYPE declaration in your document -- instructions for doing this are given above -- and it is necessary to have this declaration before the page can be declared to be valid.

❌ *Line 9, Column 32*: **required attribute "TYPE" not specified**

```
<script src="checkMainLogin.js"></script>
```

The attribute given above is required for an element that you've used, but you have omitted it. For instance, in most HTML and XHTML document types the "type" attribute is required on the "script" element and the "alt" attribute is required for the "img" element.

Typical values for `type` are `type="text/css"` for `<style>` and `type="text/javascript"` for `<script>`.

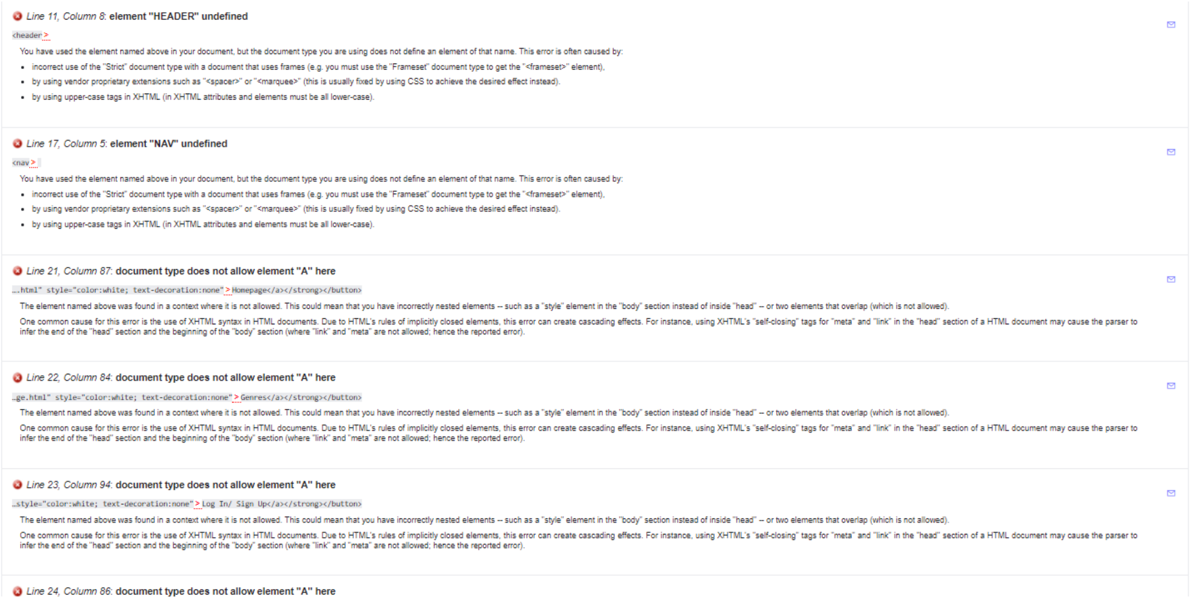
❌ *Line 11, Column 8*: **element "HEADER" undefined**

```
<header>
```

You have used the element named above in your document, but the document type you are using does not define an element of that name. This error is often caused by:

- incorrect use of the "Strict" document type with a document that uses frames (e.g. you must use the "Frameset" document type to get the "<frameset>" element),
- by using vendor proprietary extensions such as "<spacer>" or "<marquee>" (this is usually fixed by using CSS to achieve the desired effect instead).
- by using upper-case tags in XHTML (in XHTML attributes and elements must be all lower-case).

❌ *Line 17, Column 5*: **element "NAV" undefined**



❌ *Line 11, Column 8*: **element "HEADER" undefined**

```
<header>
```

You have used the element named above in your document, but the document type you are using does not define an element of that name. This error is often caused by:

- incorrect use of the "Strict" document type with a document that uses frames (e.g. you must use the "Frameset" document type to get the "<frameset>" element),
- by using vendor proprietary extensions such as "<spacer>" or "<marquee>" (this is usually fixed by using CSS to achieve the desired effect instead).
- by using upper-case tags in XHTML (in XHTML attributes and elements must be all lower-case).

❌ *Line 17, Column 5*: **element "NAV" undefined**

```
<nav>
```

You have used the element named above in your document, but the document type you are using does not define an element of that name. This error is often caused by:

- incorrect use of the "Strict" document type with a document that uses frames (e.g. you must use the "Frameset" document type to get the "<frameset>" element),
- by using vendor proprietary extensions such as "<spacer>" or "<marquee>" (this is usually fixed by using CSS to achieve the desired effect instead).
- by using upper-case tags in XHTML (in XHTML attributes and elements must be all lower-case).

❌ *Line 21, Column 87*: **document type does not allow element "A" here**

```
…html" style="color:white; text-decoration:none">Homepage</a></strong></button>
```

The element named above was found in a context where it is not allowed. This could mean that you have incorrectly nested elements -- such as a "style" element in the "body" section instead of inside "head" -- or two elements that overlap (which is not allowed).

One common cause for this error is the use of XHTML syntax in HTML documents. Due to HTML's rules of implicitly closed elements, this error can create cascading effects. For instance, using XHTML's "self-closing" tags for "meta" and "link" in the "head" section of a HTML document may cause the parser to infer the end of the "head" section and the beginning of the "body" section (where "link" and "meta" are not allowed; hence the reported error).

❌ *Line 22, Column 84*: **document type does not allow element "A" here**

```
…ge.html" style="color:white; text-decoration:none">Genres</a></strong></button>
```

The element named above was found in a context where it is not allowed. This could mean that you have incorrectly nested elements -- such as a "style" element in the "body" section instead of inside "head" -- or two elements that overlap (which is not allowed).

One common cause for this error is the use of XHTML syntax in HTML documents. Due to HTML's rules of implicitly closed elements, this error can create cascading effects. For instance, using XHTML's "self-closing" tags for "meta" and "link" in the "head" section of a HTML document may cause the parser to infer the end of the "head" section and the beginning of the "body" section (where "link" and "meta" are not allowed; hence the reported error).

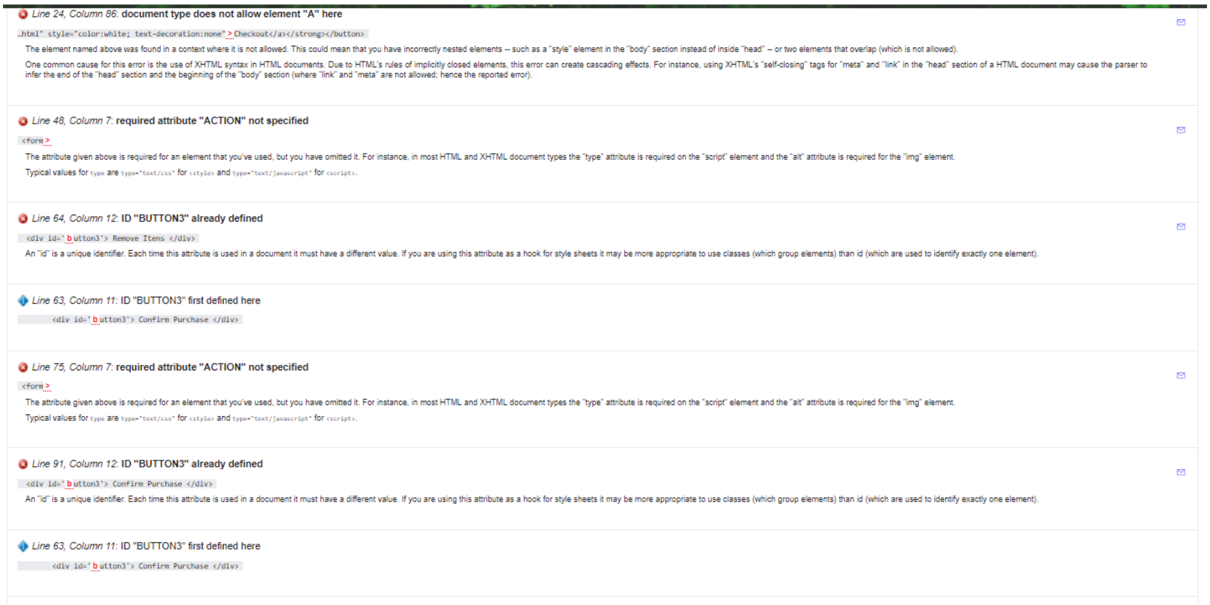
❌ *Line 23, Column 94*: **document type does not allow element "A" here**

```
…style="color:white; text-decoration:none">Log In/ Sign Up</a></strong></button>
```

The element named above was found in a context where it is not allowed. This could mean that you have incorrectly nested elements -- such as a "style" element in the "body" section instead of inside "head" -- or two elements that overlap (which is not allowed).

One common cause for this error is the use of XHTML syntax in HTML documents. Due to HTML's rules of implicitly closed elements, this error can create cascading effects. For instance, using XHTML's "self-closing" tags for "meta" and "link" in the "head" section of a HTML document may cause the parser to infer the end of the "head" section and the beginning of the "body" section (where "link" and "meta" are not allowed; hence the reported error).

❌ *Line 24, Column 86*: **document type does not allow element "A" here**



⊗ *Line 24, Column 86:* **document type does not allow element "A" here**

```
…html" style="color:white; text-decoration:none">Checkout</a></strong></button>
```

The element named above was found in a context where it is not allowed. This could mean that you have incorrectly nested elements -- such as a "style" element in the "body" section instead of inside "head" -- or two elements that overlap (which is not allowed).

One common cause for this error is the use of XHTML syntax in HTML documents. Due to HTML's rules of implicitly closed elements, this error can create cascading effects. For instance, using XHTML's "self-closing" tags for "meta" and "link" in the "head" section of a HTML document may cause the parser to infer the end of the "head" section and the beginning of the "body" section (where "link" and "meta" are not allowed; hence the reported error).

⊗ *Line 48, Column 7:* **required attribute "ACTION" not specified**

```
<form>
```

The attribute given above is required for an element that you've used, but you have omitted it. For instance, in most HTML and XHTML document types the "type" attribute is required on the "script" element and the "alt" attribute is required for the "img" element.

Typical values for `type` are `type="text/css"` for `<style>` and `type="text/javascript"` for `<script>`.

⊗ *Line 64, Column 12:* **ID "BUTTON3" already defined**

```
  <div id=' button3'> Remove Itens </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

◆ *Line 63, Column 11:* ID "BUTTON3" first defined here

```
        <div id=' button3'> Confirm Purchase </div>
```

⊗ *Line 75, Column 7:* **required attribute "ACTION" not specified**

```
<form>
```

The attribute given above is required for an element that you've used, but you have omitted it. For instance, in most HTML and XHTML document types the "type" attribute is required on the "script" element and the "alt" attribute is required for the "img" element.

Typical values for `type` are `type="text/css"` for `<style>` and `type="text/javascript"` for `<script>`.

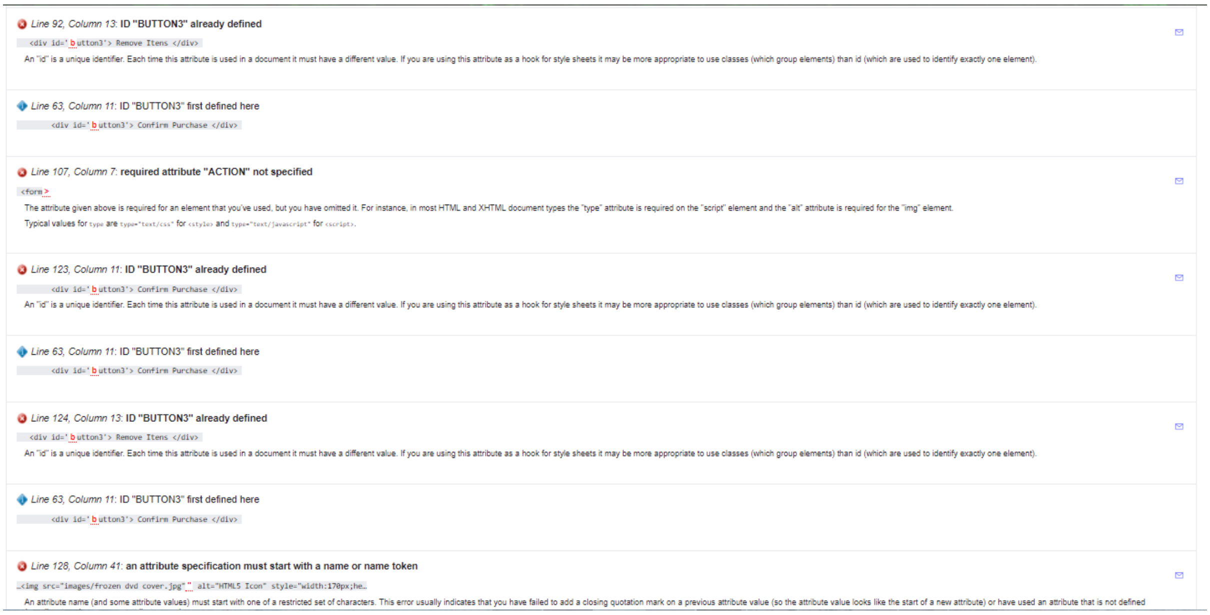
⊗ *Line 91, Column 12:* **ID "BUTTON3" already defined**

```
  <div id=' button3'> Confirm Purchase </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

◆ *Line 63, Column 11:* ID "BUTTON3" first defined here

```
        <div id=' button3'> Confirm Purchase </div>
```



⊗ *Line 92, Column 13:* **ID "BUTTON3" already defined**

```
   <div id=' button3'> Remove Itens </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

◆ *Line 63, Column 11:* ID "BUTTON3" first defined here

```
        <div id=' button3'> Confirm Purchase </div>
```

⊗ *Line 107, Column 7:* **required attribute "ACTION" not specified**

```
<form>
```

The attribute given above is required for an element that you've used, but you have omitted it. For instance, in most HTML and XHTML document types the "type" attribute is required on the "script" element and the "alt" attribute is required for the "img" element.

Typical values for `type` are `type="text/css"` for `<style>` and `type="text/javascript"` for `<script>`.

⊗ *Line 123, Column 11:* **ID "BUTTON3" already defined**

```
        <div id=' button3'> Confirm Purchase </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

◆ *Line 63, Column 11:* ID "BUTTON3" first defined here

```
        <div id=' button3'> Confirm Purchase </div>
```

⊗ *Line 124, Column 13:* **ID "BUTTON3" already defined**

```
   <div id=' button3'> Remove Itens </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

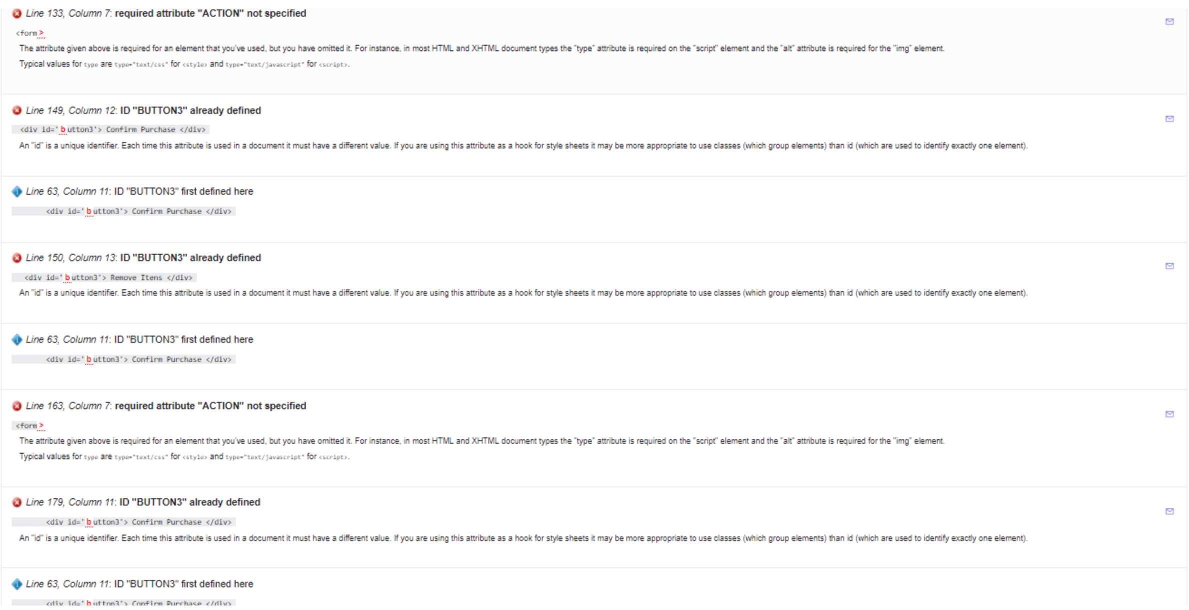
◆ *Line 63, Column 11:* ID "BUTTON3" first defined here

```
        <div id=' button3'> Confirm Purchase </div>
```

⊗ *Line 128, Column 41:* **an attribute specification must start with a name or name token**

```
…<img src="images/frozen dvd cover.jpg"" alt="HTML5 Icon" style="width:170px;he…
```

An attribute name (and some attribute values) must start with one of a restricted set of characters. This error usually indicates that you have failed to add a closing quotation mark on a previous attribute value (so the attribute value looks like the start of a new attribute) or have used an attribute that is not defined



⊗ *Line 133, Column 7:* **required attribute "ACTION" not specified**

```
<form>
```

The attribute given above is required for an element that you've used, but you have omitted it. For instance, in most HTML and XHTML document types the "type" attribute is required on the "script" element and the "alt" attribute is required for the "img" element.

Typical values for `type` are `type="text/css"` for `<style>` and `type="text/javascript"` for `<script>`.

⊗ *Line 149, Column 12:* **ID "BUTTON3" already defined**

```
  <div id=' button3'> Confirm Purchase </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

◆ *Line 63, Column 11:* ID "BUTTON3" first defined here

```
        <div id=' button3'> Confirm Purchase </div>
```

⊗ *Line 150, Column 13:* **ID "BUTTON3" already defined**

```
   <div id=' button3'> Remove Itens </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

◆ *Line 63, Column 11:* ID "BUTTON3" first defined here

```
        <div id=' button3'> Confirm Purchase </div>
```

⊗ *Line 163, Column 7:* **required attribute "ACTION" not specified**

```
<form>
```

The attribute given above is required for an element that you've used, but you have omitted it. For instance, in most HTML and XHTML document types the "type" attribute is required on the "script" element and the "alt" attribute is required for the "img" element.

Typical values for `type` are `type="text/css"` for `<style>` and `type="text/javascript"` for `<script>`.

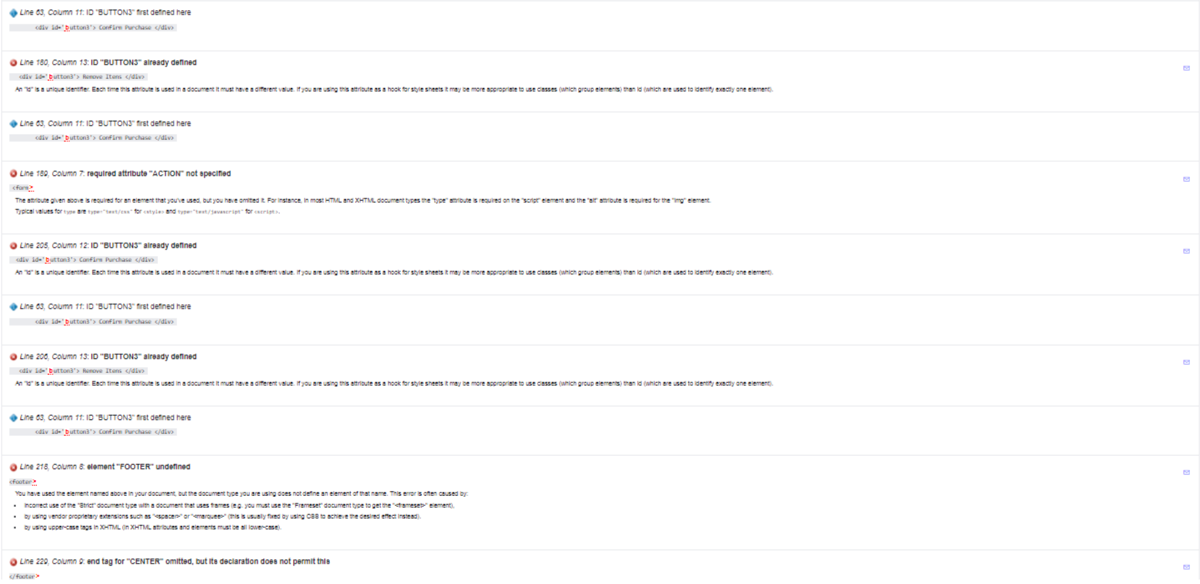
⊗ *Line 179, Column 11:* **ID "BUTTON3" already defined**

```
        <div id=' button3'> Confirm Purchase </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

◆ *Line 63, Column 11:* ID "BUTTON3" first defined here

```
        <div id=' button3'> Confirm Purchase </div>
```



Line 63, Column 11: ID "BUTTON3" first defined here

```
        <div id='button3'> Confirm Purchase </div>
```

Line 180, Column 13: **ID "BUTTON3" already defined**

```
  <div id='button3'> Remove Items </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

Line 63, Column 11: ID "BUTTON3" first defined here

```
        <div id='button3'> Confirm Purchase </div>
```

Line 189, Column 7: **required attribute "ACTION" not specified**

```
<form>
```

The attribute given above is required for an element that you've used, but you have omitted it. For instance, in most HTML and XHTML document types the "type" attribute is required on the "script" element and the "alt" attribute is required for the "img" element.

Typical values for `type` are `type="text/css"` for `<style>` and `type="text/javascript"` for `<script>`.

Line 205, Column 12: **ID "BUTTON3" already defined**

```
 <div id='button3'> Confirm Purchase </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

Line 63, Column 11: ID "BUTTON3" first defined here

```
        <div id='button3'> Confirm Purchase </div>
```

Line 206, Column 13: **ID "BUTTON3" already defined**

```
  <div id='button3'> Remove Items </div>
```

An "id" is a unique identifier. Each time this attribute is used in a document it must have a different value. If you are using this attribute as a hook for style sheets it may be more appropriate to use classes (which group elements) than id (which are used to identify exactly one element).

Line 63, Column 11: ID "BUTTON3" first defined here

```
        <div id='button3'> Confirm Purchase </div>
```

Line 218, Column 8: **element "FOOTER" undefined**

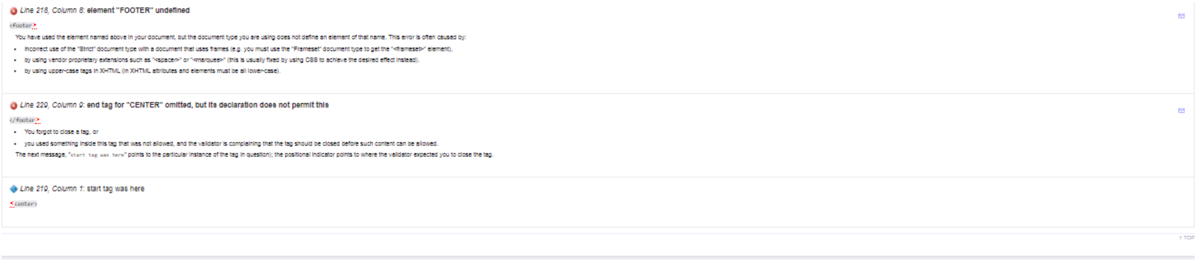
```
<footer>
```

You have used the element named above in your document, but the document type you are using does not define an element of that name. This error is often caused by:

- incorrect use of the "Strict" document type with a document that uses frames (e.g. you must use the "Frameset" document type to get the "<frameset>" element),
- by using vendor proprietary extensions such as "<spacer>" or "<marquee>" (this is usually fixed by using CSS to achieve the desired effect instead).
- by using upper-case tags in XHTML (in XHTML attributes and elements must be all lower-case).

Line 229, Column 9: **end tag for "CENTER" omitted, but its declaration does not permit this**

```
</footer>
```



Line 218, Column 8: **element "FOOTER" undefined**

```
<footer>
```

You have used the element named above in your document, but the document type you are using does not define an element of that name. This error is often caused by:

- incorrect use of the "Strict" document type with a document that uses frames (e.g. you must use the "Frameset" document type to get the "<frameset>" element),
- by using vendor proprietary extensions such as "<spacer>" or "<marquee>" (this is usually fixed by using CSS to achieve the desired effect instead).
- by using upper-case tags in XHTML (in XHTML attributes and elements must be all lower-case).

Line 229, Column 9: **end tag for "CENTER" omitted, but its declaration does not permit this**

```
</footer>
```

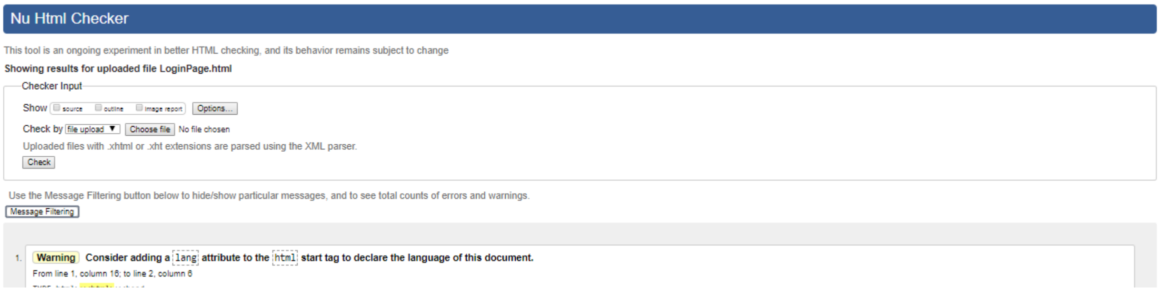
- You forgot to close a tag, or
- you used something inside this tag that was not allowed, and the validator is complaining that the tag should be closed before such content can be allowed.

The next message, "`start tag was here`" points to the particular instance of the tag in question); the positional indicator points to where the validator expected you to close the tag.

Line 219, Column 1: start tag was here

```
<center>
```

Login page:



Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for uploaded file LoginPage.html

Checker Input

Show □ source □ outline □ image report Options...

Check by file upload ▼ Choose file No file chosen

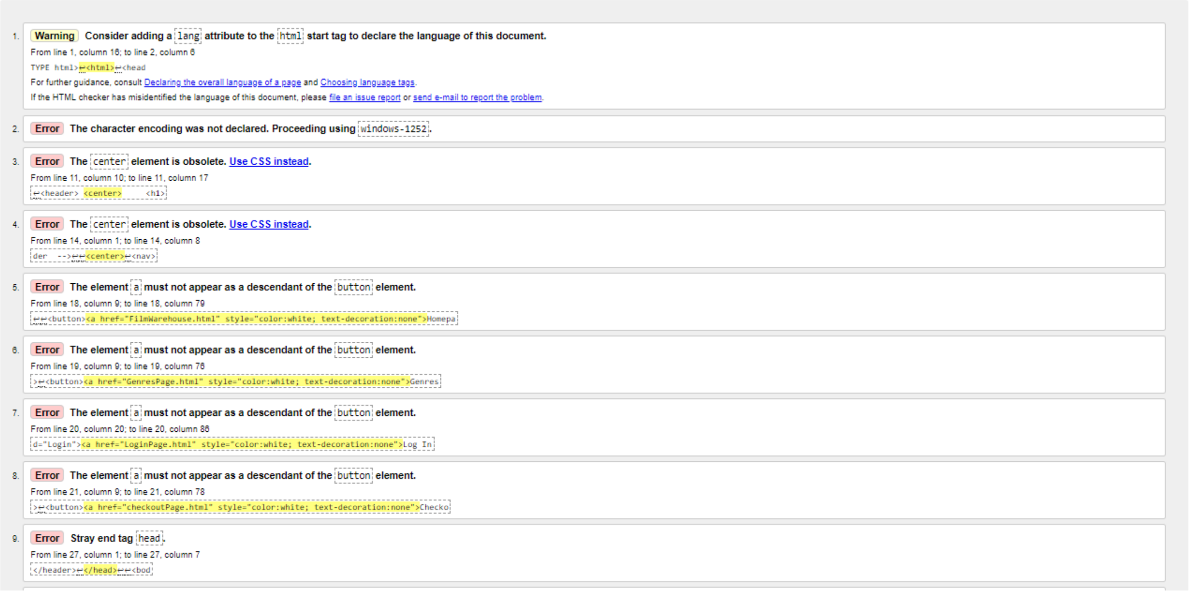
Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

Check

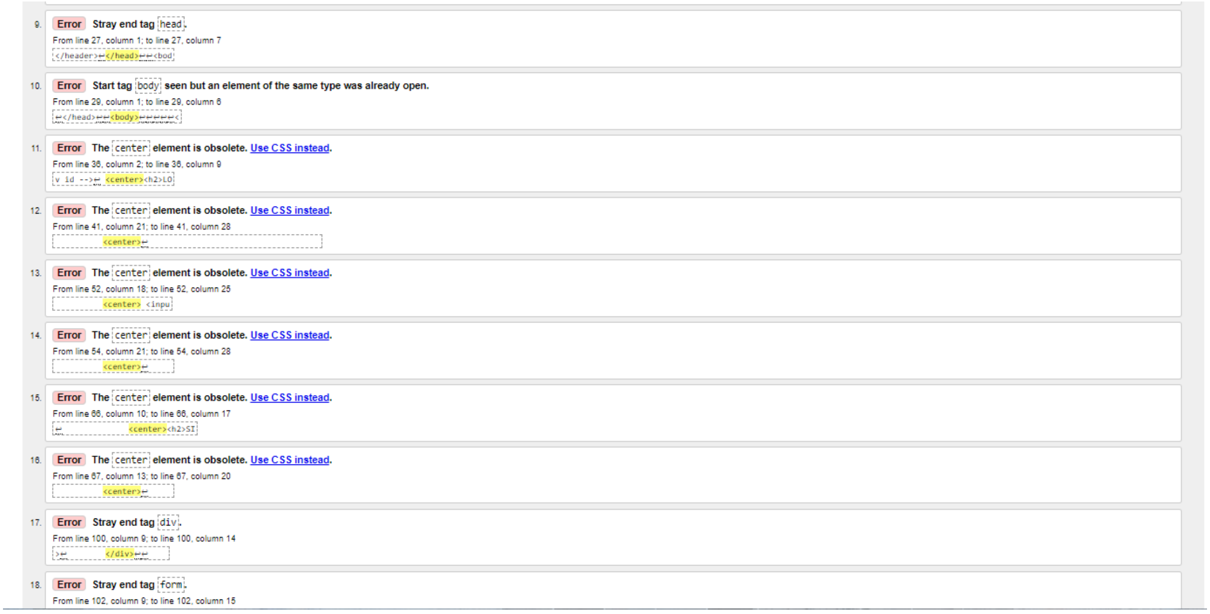
Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

1. **Warning Consider adding a `lang` attribute to the `html` start tag to declare the language of this document.**
   From line 1, column 16; to line 2, column 6



1. **Warning Consider adding a `lang` attribute to the `html` start tag to declare the language of this document.**
   From line 1, column 16; to line 2, column 6
   `TYPE html>↩<html>↩<head`
   For further guidance, consult Declaring the overall language of a page and Choosing language tags.
   If the HTML checker has misidentified the language of this document, please file an issue report or send e-mail to report the problem.
2. **Error The character encoding was not declared. Proceeding using `windows-1252`.**
3. **Error The `center` element is obsolete. Use CSS instead.**
   From line 11, column 10; to line 11, column 17
   `↩<header> <center>     <h1>`
4. **Error The `center` element is obsolete. Use CSS instead.**
   From line 14, column 1; to line 14, column 8
   `der  -->↩↩<center>↩<nav>`
5. **Error The element `a` must not appear as a descendant of the `button` element.**
   From line 18, column 9; to line 18, column 79
   `↩↩<button><a href="FilmWarehouse.html" style="color:white; text-decoration:none">Homepa`
6. **Error The element `a` must not appear as a descendant of the `button` element.**
   From line 19, column 9; to line 19, column 76
   `>↩<button><a href="GenresPage.html" style="color:white; text-decoration:none">Genres`
7. **Error The element `a` must not appear as a descendant of the `button` element.**
   From line 20, column 20; to line 20, column 86
   `d="Login"><a href="LoginPage.html" style="color:white; text-decoration:none">Log In`
8. **Error The element `a` must not appear as a descendant of the `button` element.**
   From line 21, column 9; to line 21, column 78
   `>↩<button><a href="checkoutPage.html" style="color:white; text-decoration:none">Checko`
9. **Error Stray end tag `head`.**
   From line 27, column 1; to line 27, column 7
   `</header>↩</head>↩↩<bod`



9. Error Stray end tag `head`.
From line 27, column 1; to line 27, column 7
`</header>↩</head>↩↩<bod`

10. Error Start tag `body` seen but an element of the same type was already open.
From line 29, column 1; to line 29, column 6
`↩</head>↩↩<body>↩↩↩↩↩<`

11. Error The `center` element is obsolete. Use CSS instead.
From line 36, column 2; to line 36, column 9
`v id -->↩ <center><h2>LO`

12. Error The `center` element is obsolete. Use CSS instead.
From line 41, column 21; to line 41, column 28
`            <center>↩`

13. Error The `center` element is obsolete. Use CSS instead.
From line 52, column 18; to line 52, column 25
`            <center> <inpu`

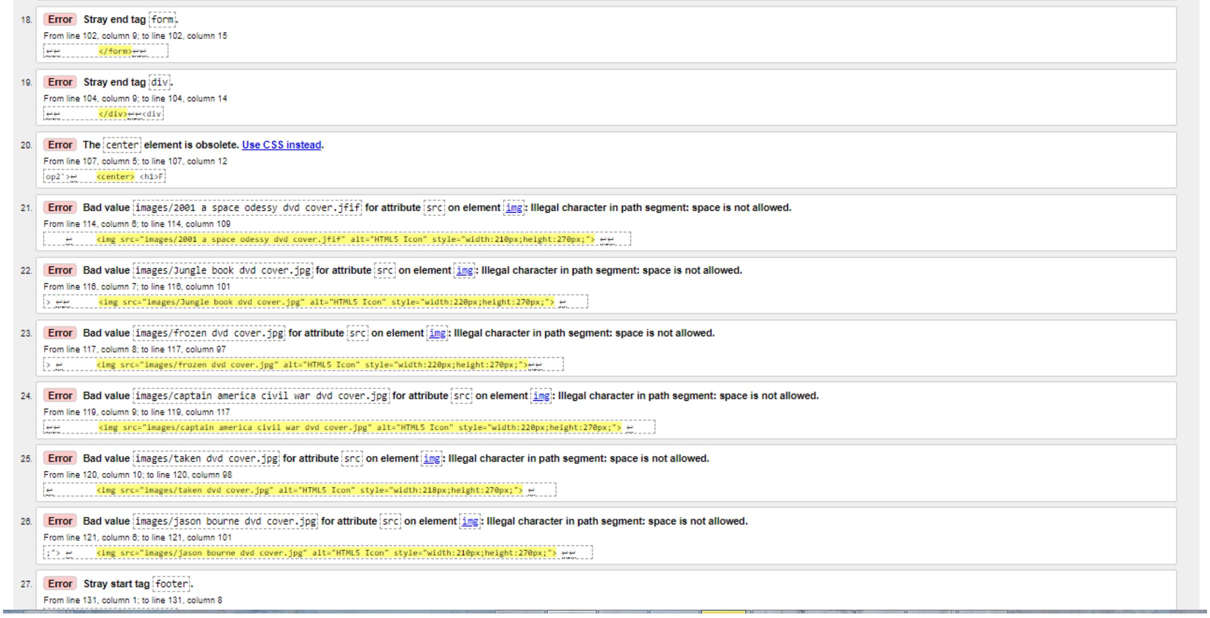
14. Error The `center` element is obsolete. Use CSS instead.
From line 54, column 21; to line 54, column 28
`            <center>↩`

15. Error The `center` element is obsolete. Use CSS instead.
From line 66, column 10; to line 66, column 17
`↩                  <center><h2>SI`

16. Error The `center` element is obsolete. Use CSS instead.
From line 67, column 13; to line 67, column 20
`            <center>↩`

17. Error Stray end tag `div`.
From line 100, column 9; to line 100, column 14
`>↩          </div>↩↩`

18. Error Stray end tag `form`.
From line 102, column 9; to line 102, column 15



18. Error Stray end tag `form`.
From line 102, column 9; to line 102, column 15
`↩↩         </form>↩↩`

19. Error Stray end tag `div`.
From line 104, column 9; to line 104, column 14
`↩↩         </div>↩↩<div`

20. Error The `center` element is obsolete. Use CSS instead.
From line 107, column 5; to line 107, column 12
`op2'>↩     <center> <h1>F`

21. Error Bad value `images/2001 a space odessy dvd cover.jfif` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.
From line 114, column 6; to line 114, column 109
`    ↩     <img src="images/2001 a space odessy dvd cover.jfif" alt="HTML5 Icon" style="width:210px;height:270px;"> ↩↩`

22. Error Bad value `images/Jungle book dvd cover.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.
From line 116, column 7; to line 116, column 101
`> ↩↩      <img src="images/Jungle book dvd cover.jpg" alt="HTML5 Icon" style="width:220px;height:270px;"> ↩`

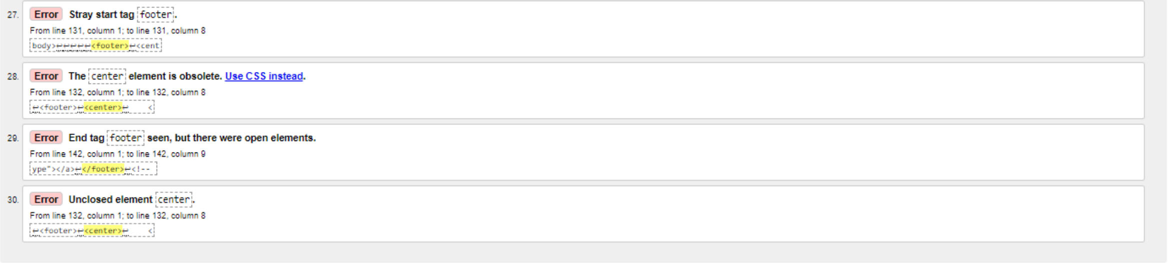
23. Error Bad value `images/frozen dvd cover.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.
From line 117, column 8; to line 117, column 97
`> ↩      <img src="images/frozen dvd cover.jpg" alt="HTML5 Icon" style="width:220px;height:270px;">↩↩`

24. Error Bad value `images/captain america civil war dvd cover.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.
From line 119, column 9; to line 119, column 117
`↩↩       <img src="images/captain america civil war dvd cover.jpg" alt="HTML5 Icon" style="width:220px;height:270px;"> ↩`

25. Error Bad value `images/taken dvd cover.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.
From line 120, column 10; to line 120, column 98
`↩        <img src="images/taken dvd cover.jpg" alt="HTML5 Icon" style="width:218px;height:270px;"> ↩`

26. Error Bad value `images/jason bourne dvd cover.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.
From line 121, column 6; to line 121, column 101
`;"> ↩     <img src="images/jason bourne dvd cover.jpg" alt="HTML5 Icon" style="width:210px;height:270px;"> ↩↩`

27. Error Stray start tag `footer`.
From line 131, column 1; to line 131, column 8



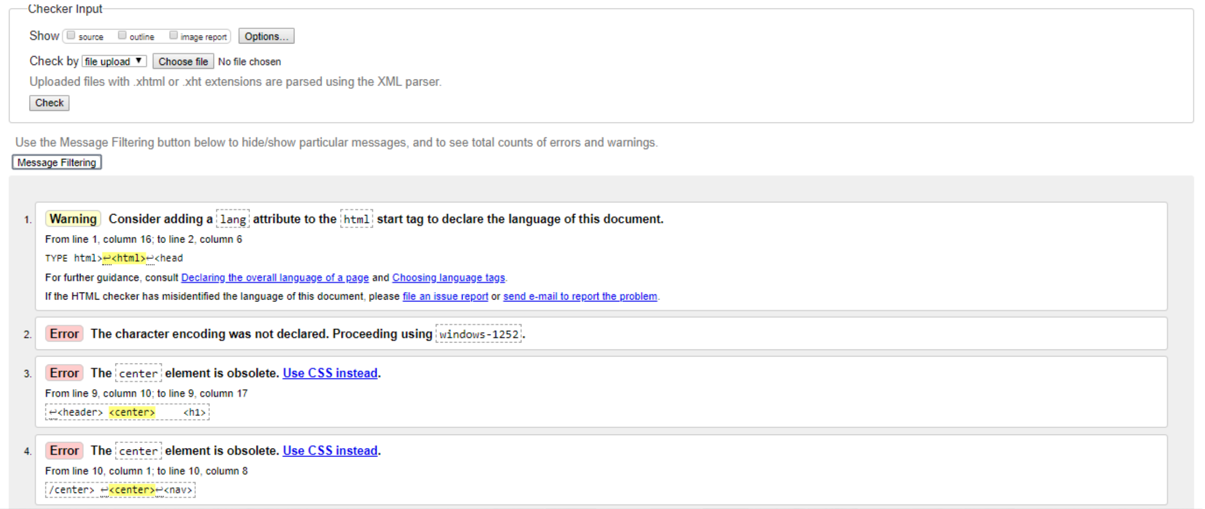
27. Error Stray start tag `footer`.
From line 131, column 1; to line 131, column 8
`body>↩↩↩↩↩<footer>↩<cent`

28. Error The `center` element is obsolete. Use CSS instead.
From line 132, column 1; to line 132, column 8
`↩<footer>↩<center>↩    <`

29. Error End tag `footer` seen, but there were open elements.
From line 142, column 1; to line 142, column 9
`ype"></a>↩</footer>↩<!--`

30. Error Unclosed element `center`.
From line 132, column 1; to line 132, column 8
`↩<footer>↩<center>↩    <`

Genres page:



Checker Input

Show □ source □ outline □ image report [Options...]

Check by [file upload ▼] [Choose file] No file chosen

Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

[Check]

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

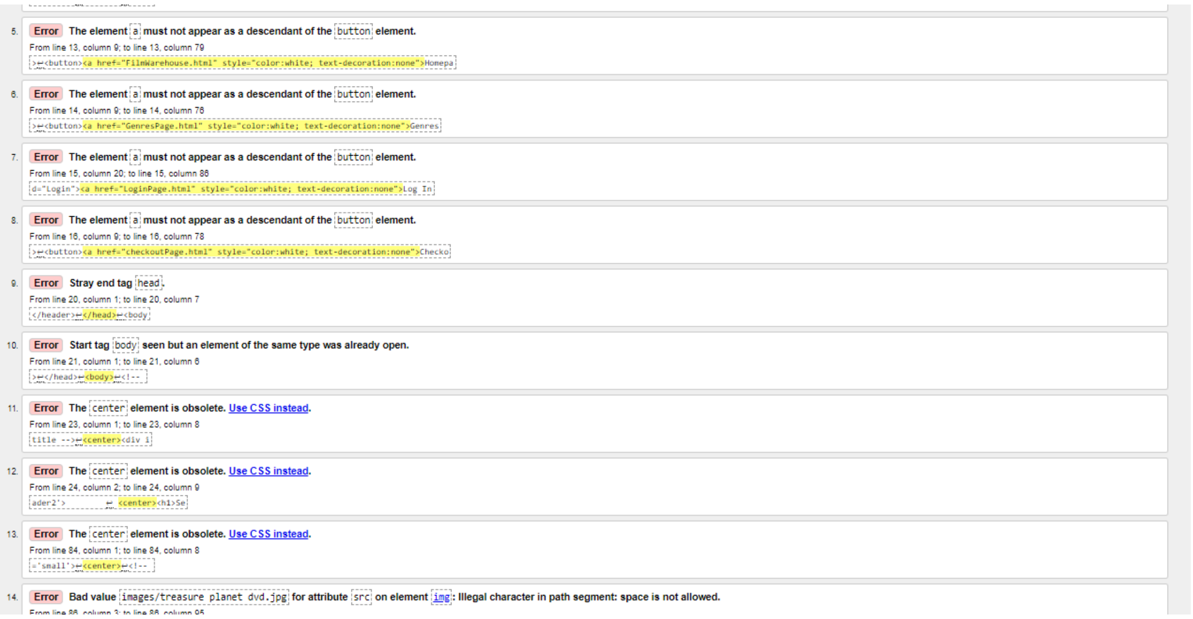
[Message Filtering]

1. **Warning** **Consider adding a `lang` attribute to the `html` start tag to declare the language of this document.**
   From line 1, column 16; to line 2, column 6
   `TYPE html>↵<html>↵<head`
   For further guidance, consult Declaring the overall language of a page and Choosing language tags.
   If the HTML checker has misidentified the language of this document, please file an issue report or send e-mail to report the problem.

2. **Error** **The character encoding was not declared. Proceeding using `windows-1252`.**

3. **Error** **The `center` element is obsolete. Use CSS instead.**
   From line 9, column 10; to line 9, column 17
   `↵<header> <center>     <h1>`

4. **Error** **The `center` element is obsolete. Use CSS instead.**
   From line 10, column 1; to line 10, column 8
   `/center> ↵<center>↵<nav>`



5. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 13, column 9; to line 13, column 79
   `>↵<button><a href="FilmWarehouse.html" style="color:white; text-decoration:none">Homepa`

6. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 14, column 9; to line 14, column 76
   `>↵<button><a href="GenresPage.html" style="color:white; text-decoration:none">Genres`

7. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 15, column 20; to line 15, column 86
   `d="Login"><a href="LoginPage.html" style="color:white; text-decoration:none">Log In`

8. **Error** **The element `a` must not appear as a descendant of the `button` element.**
   From line 16, column 9; to line 16, column 78
   `>↵<button><a href="checkoutPage.html" style="color:white; text-decoration:none">Checko`

9. **Error** **Stray end tag `head`.**
   From line 20, column 1; to line 20, column 7
   `</header>↵</head>↵<body`

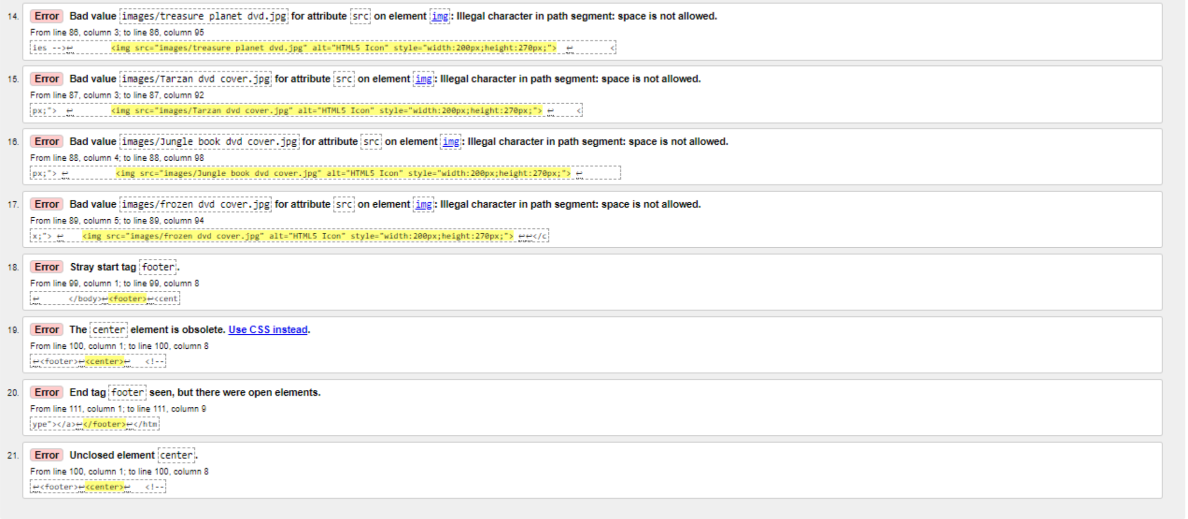
10. **Error** **Start tag `body` seen but an element of the same type was already open.**
    From line 21, column 1; to line 21, column 6
    `>↵</head>↵<body>↵<!--`

11. **Error** **The `center` element is obsolete. Use CSS instead.**
    From line 23, column 1; to line 23, column 8
    `title -->↵<center><div i`

12. **Error** **The `center` element is obsolete. Use CSS instead.**
    From line 24, column 2; to line 24, column 9
    `ader2'>         ↵ <center><h1>Se`

13. **Error** **The `center` element is obsolete. Use CSS instead.**
    From line 84, column 1; to line 84, column 8
    `='small'>↵<center>↵<!--`

14. **Error** **Bad value `images/treasure planet dvd.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.**
    From line 86, column 3; to line 86, column 95



14. **Error** **Bad value `images/treasure planet dvd.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.**
    From line 86, column 3; to line 86, column 95
    `ies -->↵        <img src="images/treasure planet dvd.jpg" alt="HTML5 Icon" style="width:200px;height:270px;">  ↵        <`

15. **Error** **Bad value `images/Tarzan dvd cover.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.**
    From line 87, column 3; to line 87, column 92
    `px;">  ↵        <img src="images/Tarzan dvd cover.jpg" alt="HTML5 Icon" style="width:200px;height:270px;"> ↵      <`

16. **Error** **Bad value `images/Jungle book dvd cover.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.**
    From line 88, column 4; to line 88, column 98
    `px;"> ↵         <img src="images/Jungle book dvd cover.jpg" alt="HTML5 Icon" style="width:200px;height:270px;"> ↵`

17. **Error** **Bad value `images/frozen dvd cover.jpg` for attribute `src` on element `img`: Illegal character in path segment: space is not allowed.**
    From line 89, column 5; to line 89, column 94
    `x;"> ↵    <img src="images/frozen dvd cover.jpg" alt="HTML5 Icon" style="width:200px;height:270px;"> ↵↵</c`

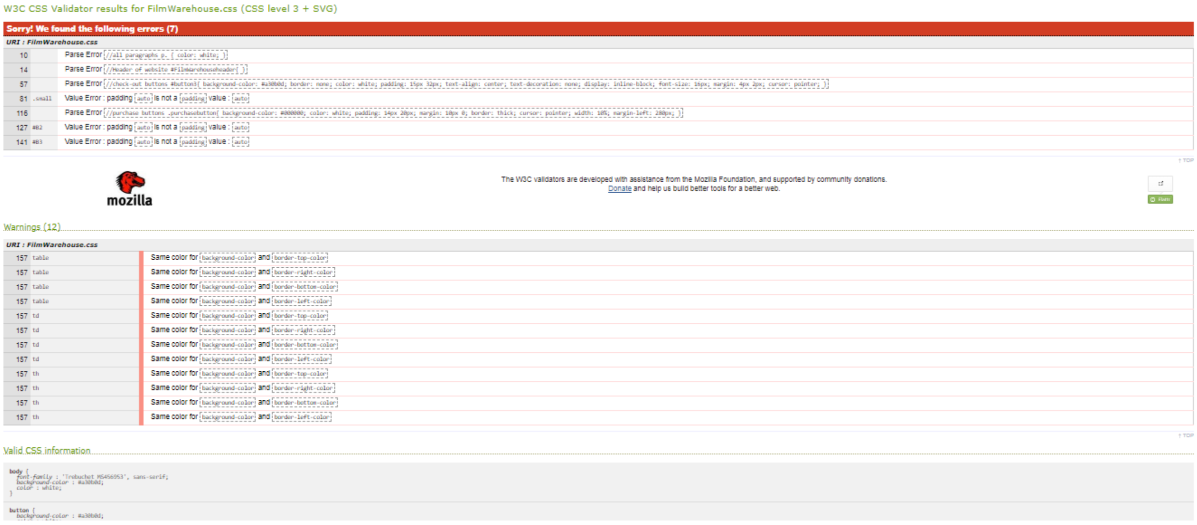
18. **Error** **Stray start tag `footer`.**
    From line 99, column 1; to line 99, column 8
    `↵       </body>↵<footer>↵<cent`

19. **Error** **The `center` element is obsolete. Use CSS instead.**
    From line 100, column 1; to line 100, column 8
    `↵<footer>↵<center>↵   <!--`

20. **Error** **End tag `footer` seen, but there were open elements.**
    From line 111, column 1; to line 111, column 9
    `ype"></a>↵</footer>↵</htm`

21. **Error** **Unclosed element `center`.**
    From line 100, column 1; to line 100, column 8
    `↵<footer>↵<center>↵   <!--`

Film Warehouse CSS:



W3C CSS Validator results for FilmWarehouse.css (CSS level 3 + SVG)

Sorry! We found the following errors (7)

```
URI : FilmWarehouse.css
10          Parse Error //all paragraphs p. { color: white; }
14          Parse Error //Header of website #FilmWarehouseheader{ }
57          Parse Error //check-out buttons #button3{ background-color: #a30b0d; border: none; color: white; padding: 15px 32px; text-align: center; text-decoration: none; display: inline-block; font-size: 16px; margin: 4px 2px; cursor: pointer; }
81  .small  Value Error : padding auto is not a padding value : auto
116         Parse Error //purchase buttons .purchasebutton{ background-color: #000000; color: white; padding: 14px 20px; margin: 10px 0; border: thick; cursor: pointer; width: 10%; margin-left: 280px; }
127 #B2    Value Error : padding auto is not a padding value : auto
141 #B3    Value Error : padding auto is not a padding value : auto
```

↑ TOP

The W3C validators are developed with assistance from the Mozilla Foundation, and supported by community donations.
Donate and help us build better tools for a better web.

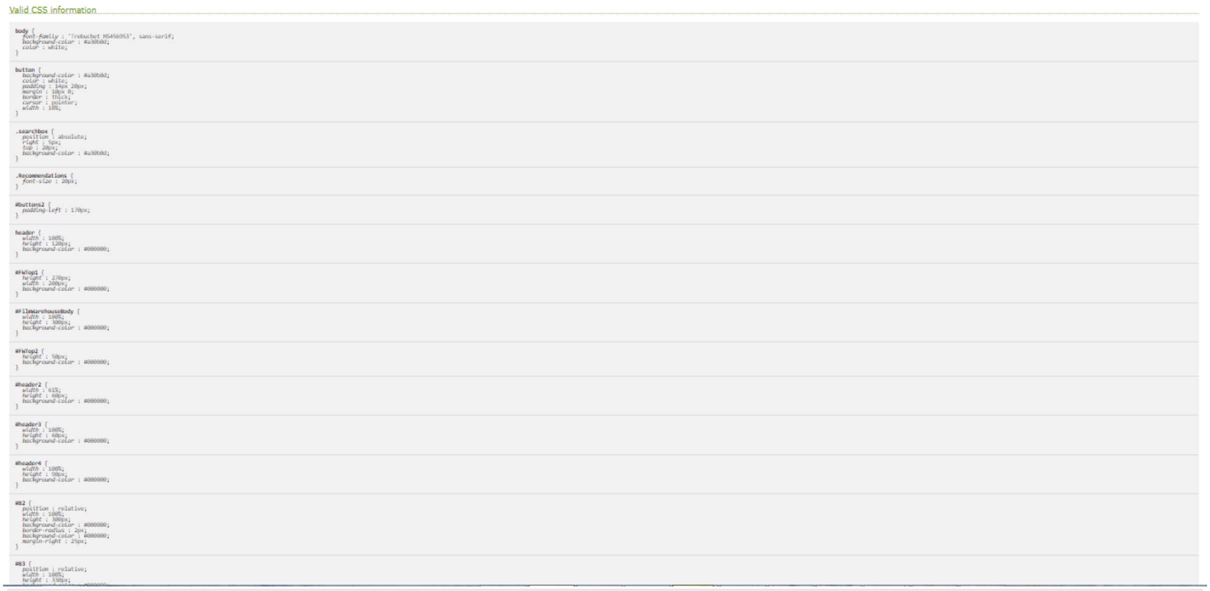
Warnings (12)

```
URI : FilmWarehouse.css
157 table  Same color for background-color and border-top-color
157 table  Same color for background-color and border-right-color
157 table  Same color for background-color and border-bottom-color
157 table  Same color for background-color and border-left-color
157 td     Same color for background-color and border-top-color
157 td     Same color for background-color and border-right-color
157 td     Same color for background-color and border-bottom-color
157 td     Same color for background-color and border-left-color
157 th     Same color for background-color and border-top-color
157 th     Same color for background-color and border-right-color
157 th     Same color for background-color and border-bottom-color
157 th     Same color for background-color and border-left-color
```

↑ TOP

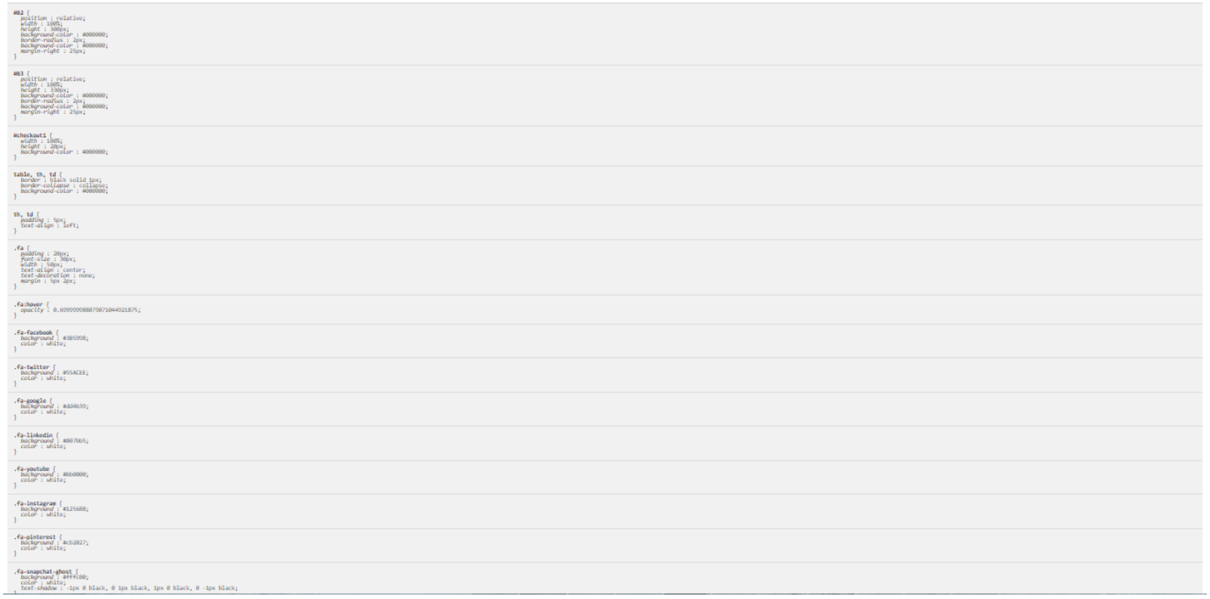
Valid CSS information

```
body {
  font-family : 'Trebuchet MS456953', sans-serif;
  background-color : #a30b0d;
  color : white;
}

button {
  background-color : #a30b0d;
```

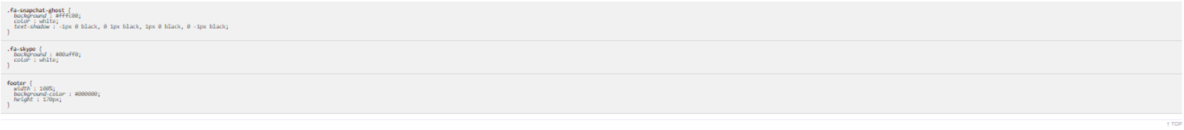


Valid CSS information

```
body {
  font-family : 'Trebuchet MS456953', sans-serif;
  background-color : #a30b0d;
  color : white;
}

button {
  background-color : #a30b0d;
  color : white;
  padding : 14px 20px;
  margin : 10px 0;
  border : thick;
  cursor : pointer;
  width : 10%;
}

.searchbox {
  position : absolute;
  right : 5px;
  top : 20px;
  background-color : #a30b0d;
}

.Recommendations {
  font-size : 20px;
}

#buttons2 {
  padding-left : 170px;
}

header {
  width : 100%;
  height : 120px;
  background-color : #000000;
}

#FWTop1 {
  height : 270px;
  width : 200px;
  background-color : #000000;
}

#FilmWarehouseBody {
  width : 100%;
  height : 300px;
  background-color : #000000;
}

#FWTop2 {
  height : 50px;
  background-color : #000000;
}

#header2 {
  width : 61%;
  height : 60px;
  background-color : #000000;
}

#header3 {
  width : 100%;
  height : 60px;
  background-color : #000000;
}

#header4 {
  width : 100%;
  height : 90px;
  background-color : #000000;
}

#B2 {
  position : relative;
  width : 100%;
  height : 300px;
  background-color : #000000;
  border-radius : 2px;
  background-color : #000000;
  margin-right : 25px;
}

#B3 {
  position : relative;
  width : 100%;
  height : 330px;
```



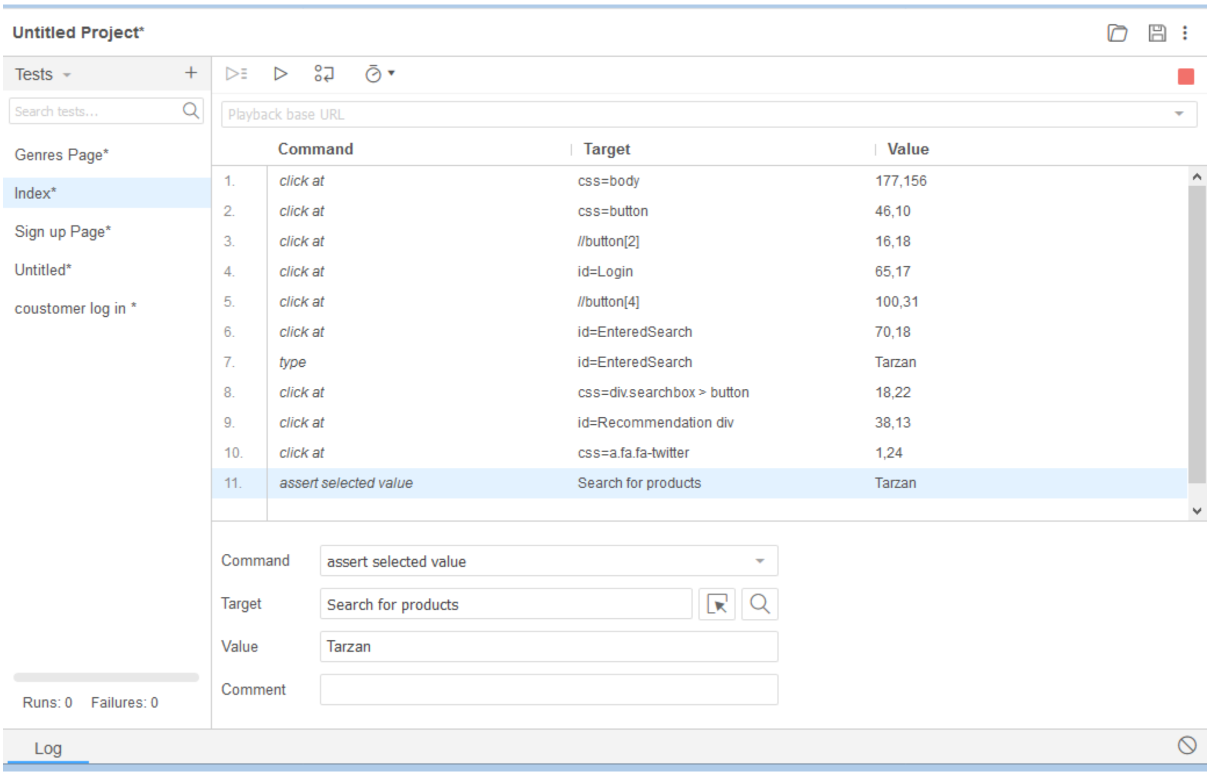
```
#B2 {
  position : relative;
  width : 100%;
  height : 300px;
  background-color : #000000;
  border-radius : 2px;
  background-color : #000000;
  margin-right : 25px;
}

#B3 {
  position : relative;
  width : 100%;
  height : 330px;
  background-color : #000000;
  border-radius : 2px;
  background-color : #000000;
  margin-right : 25px;
}

#checkout1 {
  width : 100%;
  height : 20px;
  background-color : #000000;
}

table, th, td {
  border : black solid 1px;
  border-collapse : collapse;
  background-color : #000000;
}

th, td {
  padding : 5px;
  text-align : left;
}

.fa {
  padding : 20px;
  font-size : 30px;
  width : 50px;
  text-align : center;
  text-decoration : none;
  margin : 5px 2px;
}

.fa:hover {
  opacity : 0.699999988079071044921875;
}

.fa-facebook {
  background : #3B5998;
  color : white;
}

.fa-twitter {
  background : #55ACEE;
  color : white;
}

.fa-google {
  background : #dd4b39;
  color : white;
}

.fa-linkedin {
  background : #007bb5;
  color : white;
}

.fa-youtube {
  background : #bb0000;
  color : white;
}

.fa-instagram {
  background : #125688;
  color : white;
}

.fa-pinterest {
  background : #cb2027;
  color : white;
}

.fa-snapchat-ghost {
  background : #fffc00;
  color : white;
  text-shadow : -1px 0 black, 0 1px black, 1px 0 black, 0 -1px black;
}
```



```
.fa-snapchat-ghost {
  background : #fffc00;
  color : white;
  text-shadow : -1px 0 black, 0 1px black, 1px 0 black, 0 -1px black;
}

.fa-skype {
  background : #00aff0;
  color : white;
}

footer {
  width : 100%;
  background-color : #000000;
  height : 170px;
}
```

Front end Testing

Index Page (go to the Search for products section and look for Tarzan + Press nav bar buttons):



Untitled Project*

	Command	Target	Value
1.	*click at*	css=body	177,156
2.	*click at*	css=button	46,10
3.	*click at*	//button[2]	16,18
4.	*click at*	id=Login	65,17
5.	*click at*	//button[4]	100,31
6.	*click at*	id=EnteredSearch	70,18
7.	*type*	id=EnteredSearch	Tarzan
8.	*click at*	css=div.searchbox > button	18,22
9.	*click at*	id=Recommendation div	38,13
10.	*click at*	css=a.fa.fa-twitter	1,24
11.	*assert selected value*	Search for products	Tarzan

Command: assert selected value

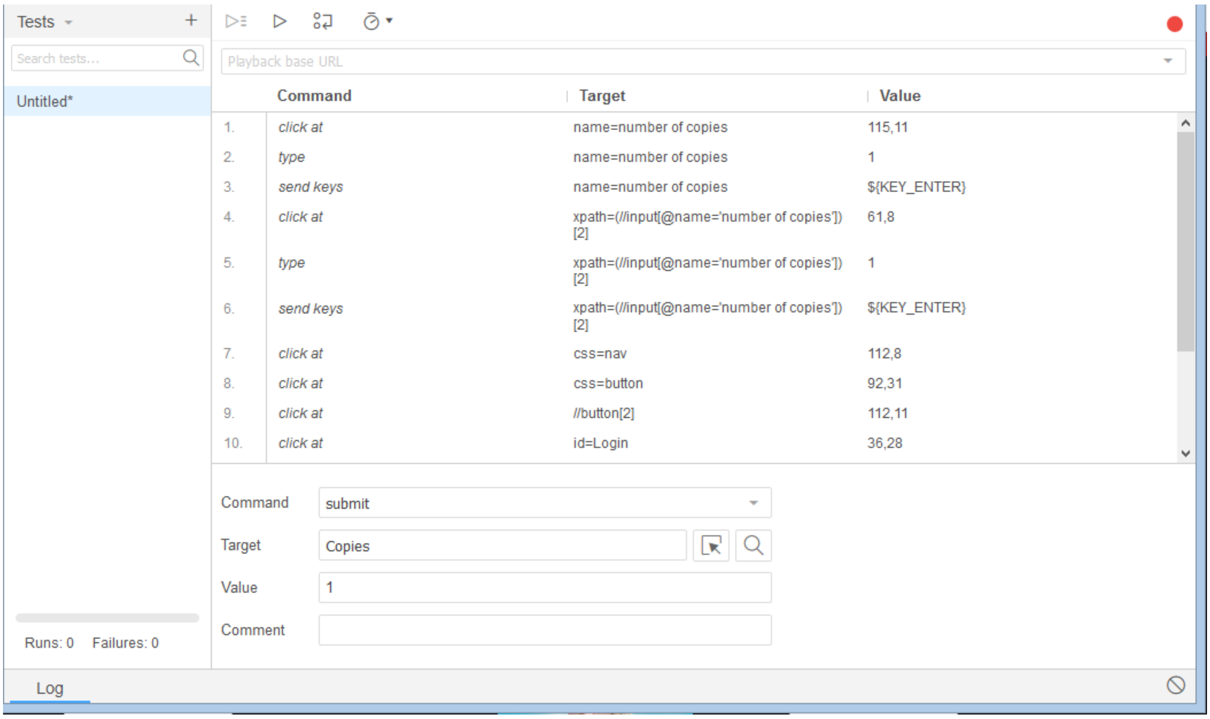
Target: Search for products

Value: Tarzan

Comment:

Runs: 0 Failures: 0

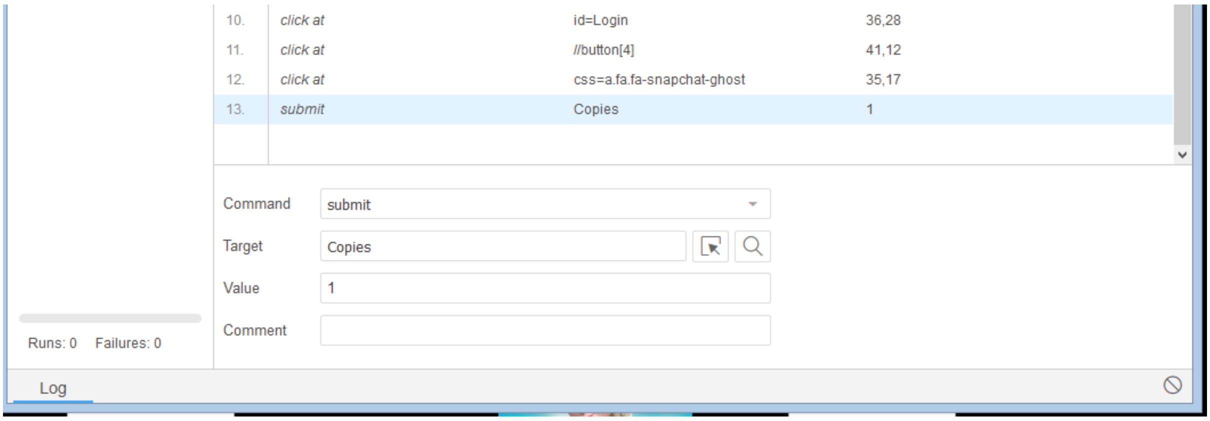
Checkout Page (Enter 1 copy for the “Copies” section + pressed nav bar buttons)



	Command	Target	Value
1.	*click at*	name=number of copies	115,11
2.	*type*	name=number of copies	1
3.	*send keys*	name=number of copies	${KEY_ENTER}
4.	*click at*	xpath=(//input[@name='number of copies'])[2]	61,8
5.	*type*	xpath=(//input[@name='number of copies'])[2]	1
6.	*send keys*	xpath=(//input[@name='number of copies'])[2]	${KEY_ENTER}
7.	*click at*	css=nav	112,8
8.	*click at*	css=button	92,31
9.	*click at*	//button[2]	112,11
10.	*click at*	id=Login	36,28

Command	submit
Target	Copies
Value	1
Comment	

Runs: 0 Failures: 0

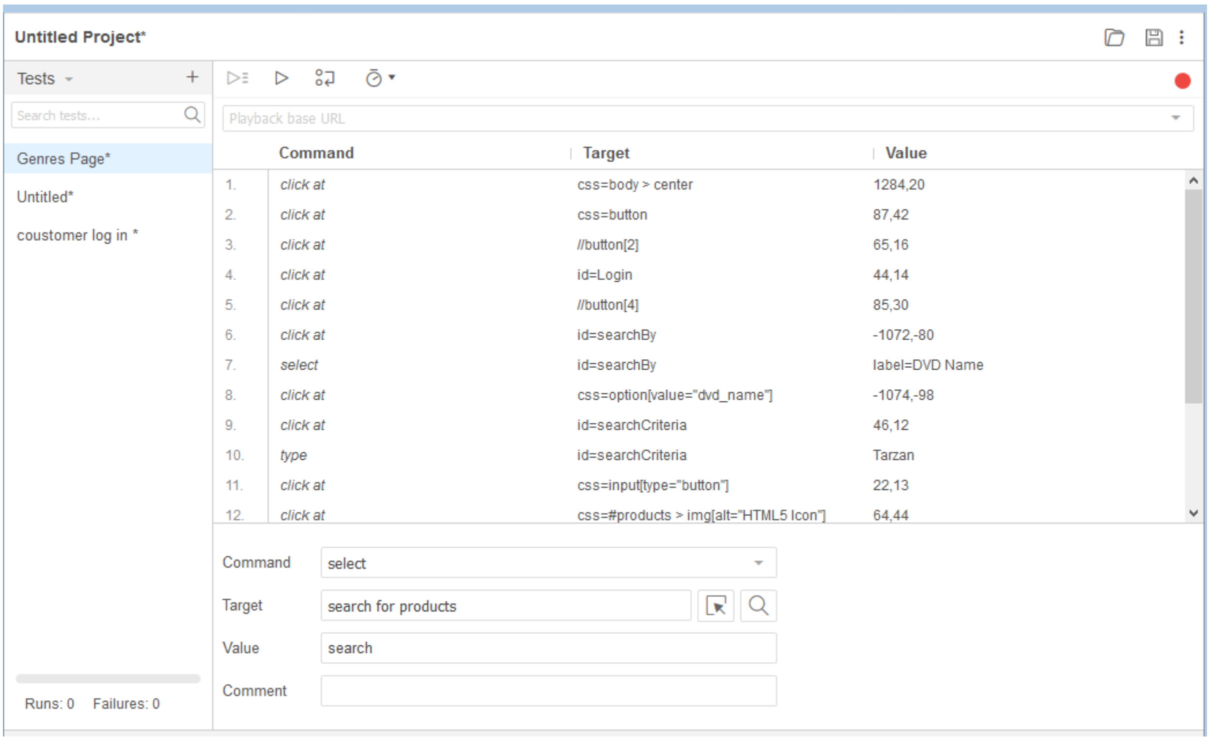


10.	*click at*	id=Login	36,28
11.	*click at*	//button[4]	41,12
12.	*click at*	css=a.fa.fa-snapchat-ghost	35,17
13.	*submit*	Copies	1

Command	submit
Target	Copies
Value	1
Comment	

Runs: 0 Failures: 0

Genres Page (Go on the Search for products page and find Tarzan + Press nav bar buttons):



	Command	Target	Value
1.	*click at*	css=body > center	1284,20
2.	*click at*	css=button	87,42
3.	*click at*	//button[2]	65,16
4.	*click at*	id=Login	44,14
5.	*click at*	//button[4]	85,30
6.	*click at*	id=searchBy	-1072,-80
7.	*select*	id=searchBy	label=DVD Name
8.	*click at*	css=option[value="dvd_name"]	-1074,-98
9.	*click at*	id=searchCriteria	46,12
10.	*type*	id=searchCriteria	Tarzan
11.	*click at*	css=input[type="button"]	22,13
12.	*click at*	css=#products > img[alt="HTML5 Icon"]	64,44

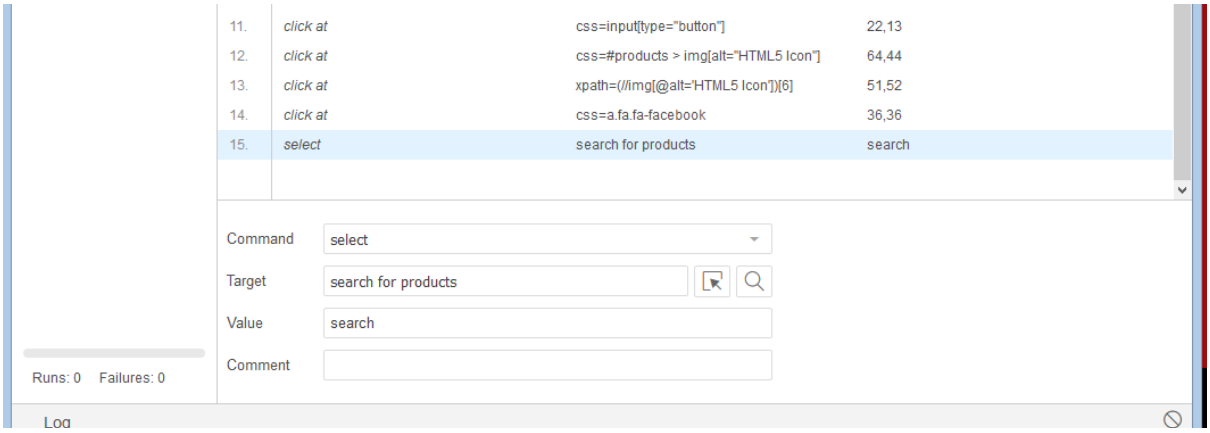
Command: select

Target: search for products

Value: search

Comment:

Runs: 0 Failures: 0



11.	*click at*	css=input[type="button"]	22,13
12.	*click at*	css=#products > img[alt="HTML5 Icon"]	64,44
13.	*click at*	xpath=(//img[@alt='HTML5 Icon'])[6]	51,52
14.	*click at*	css=a.fa.fa-facebook	36,36
15.	*select*	search for products	search

Command: select

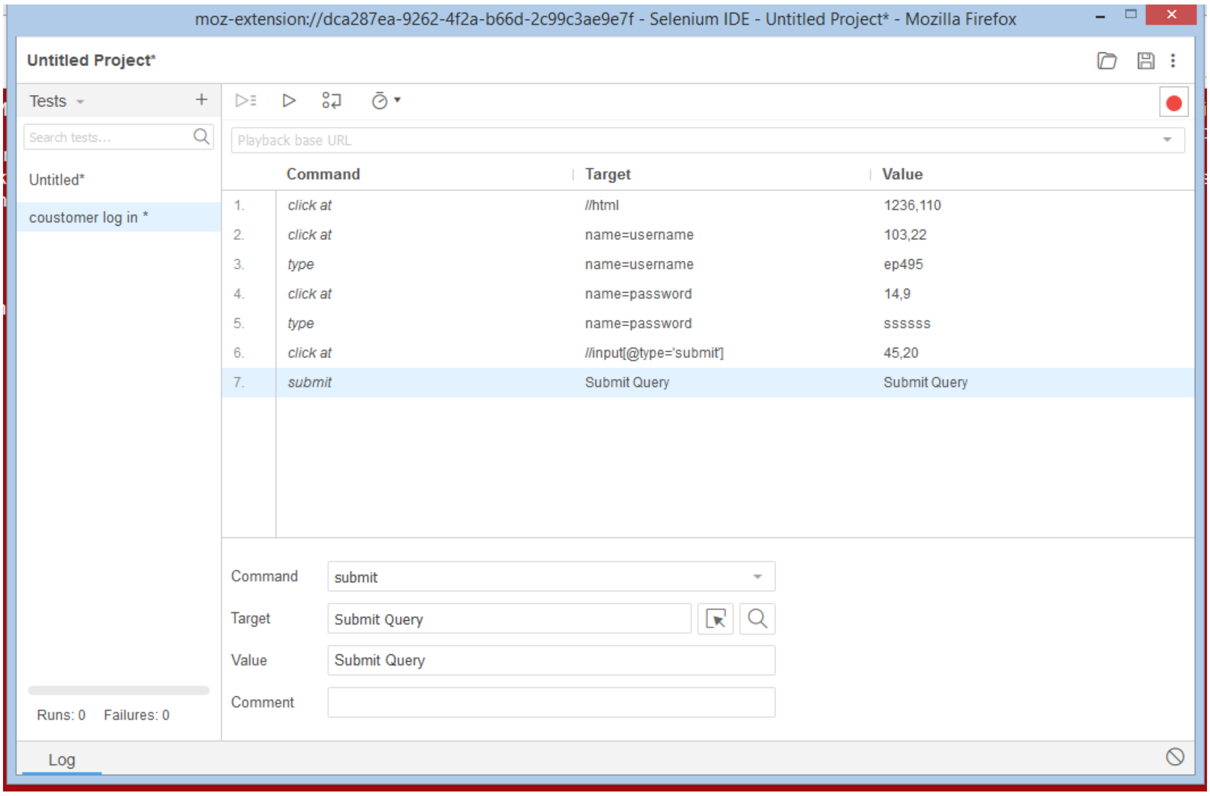
Target: search for products

Value: search

Comment:

Runs: 0 Failures: 0

Customer Login (Try to log in):



moz-extension://dca287ea-9262-4f2a-b66d-2c99c3ae9e7f - Selenium IDE - Untitled Project* - Mozilla Firefox

Untitled Project*

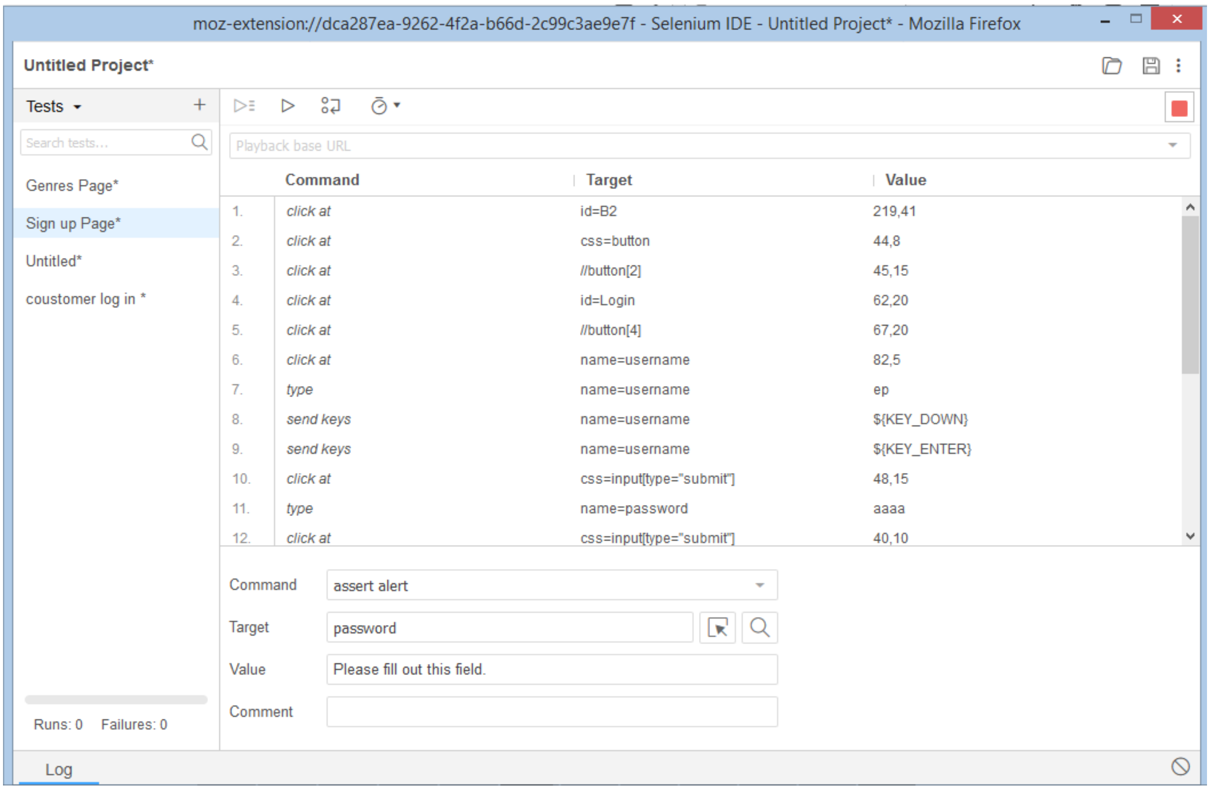
Untitled*
coustomer log in *

	Command	Target	Value
1.	*click at*	//html	1236,110
2.	*click at*	name=username	103,22
3.	*type*	name=username	ep495
4.	*click at*	name=password	14,9
5.	*type*	name=password	ssssss
6.	*click at*	//input[@type='submit']	45,20
7.	*submit*	Submit Query	Submit Query

Command: submit
Target: Submit Query
Value: Submit Query
Comment:

Runs: 0 Failures: 0

Sign Up Page (Try to sign up and log in while checking if the site sends out an alert for not filling out password section + Press nav bar buttons):



moz-extension://dca287ea-9262-4f2a-b66d-2c99c3ae9e7f - Selenium IDE - Untitled Project* - Mozilla Firefox

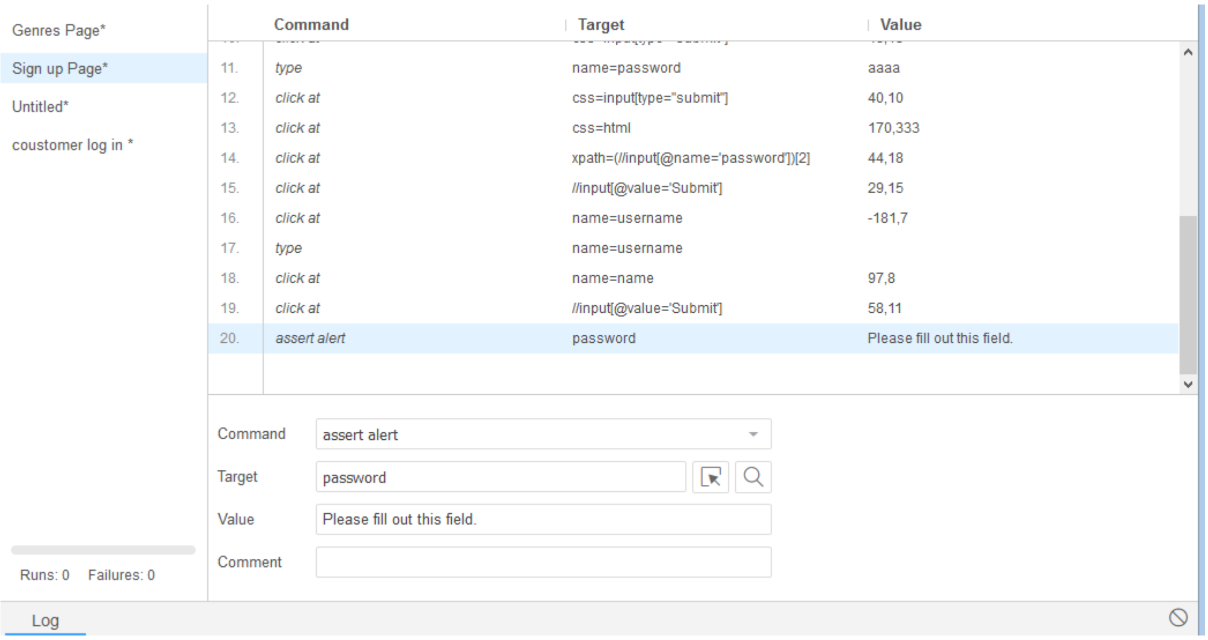
Untitled Project*

Genres Page*
Sign up Page*
Untitled*
coustomer log in *

	Command	Target	Value
1.	*click at*	id=B2	219,41
2.	*click at*	css=button	44,8
3.	*click at*	//button[2]	45,15
4.	*click at*	id=Login	62,20
5.	*click at*	//button[4]	67,20
6.	*click at*	name=username	82,5
7.	*type*	name=username	ep
8.	*send keys*	name=username	${KEY_DOWN}
9.	*send keys*	name=username	${KEY_ENTER}
10.	*click at*	css=input[type="submit"]	48,15
11.	*type*	name=password	aaaa
12.	*click at*	css=input[type="submit"]	40,10

Command: assert alert
Target: password
Value: Please fill out this field.
Comment:

Runs: 0 Failures: 0



Genres Page*
Sign up Page*
Untitled*
coustomer log in *

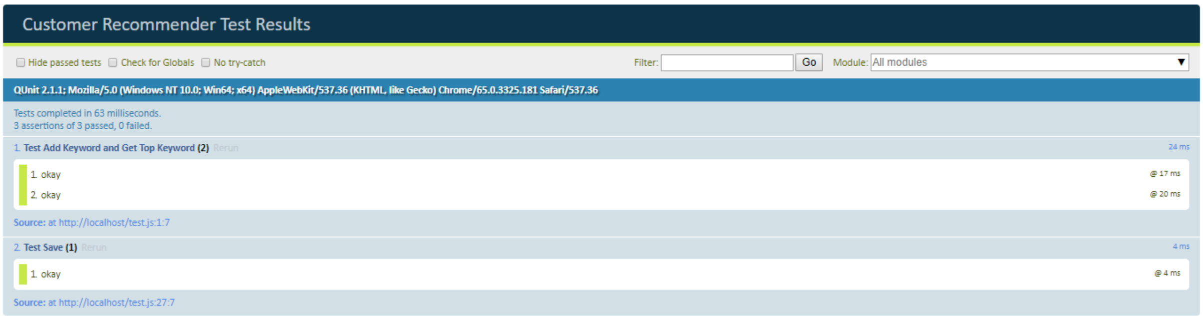
	Command	Target	Value
11.	type	name=password	aaaa
12.	click at	css=input[type="submit"]	40,10
13.	click at	css=html	170,333
14.	click at	xpath=(//input[@name='password'])[2]	44,18
15.	click at	//input[@value='Submit']	29,15
16.	click at	name=username	-181,7
17.	type	name=username	
18.	click at	name=name	97,8
19.	click at	//input[@value='Submit']	58,11
20.	assert alert	password	Please fill out this field.

Command: assert alert
Target: password
Value: Please fill out this field.
Comment:

Runs: 0 Failures: 0

JavaScript Testing

- “Customer Recommender test” and “Save test” (code available in test.js)



Customer Recommender Test Results

☐ Hide passed tests ☐ Check for Globals ☐ No try-catch

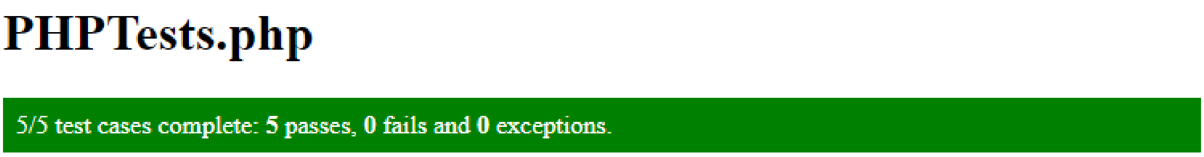
QUnit 2.1.1; Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/65.0.3325.181 Safari/537.36

Tests completed in 63 milliseconds.
3 assertions of 3 passed, 0 failed.

1. Test Add Keyword and Get Top Keyword (2) Rerun 24 ms
1. okay @ 17 ms
2. okay @ 20 ms
Source: at http://localhost/test.js:1:7

2. Test Save (1) Rerun 4 ms
1. okay @ 4 ms
Source: at http://localhost/test.js:27:7

PHP Unit Testing



PHPTests.php

5/5 test cases complete: 5 passes, 0 fails and 0 exceptions.

The test files are available in the folder. “PHPTests.php” is the main file used for the tests. “db_tools.php” contains the utility functions used in the tests, and “db_interface” contains the re-written code from the website, put into functions and run.

Five tests were written (in the file PHPTests.php) and passed, as seen in the screenshot above:

- To add a customer
- To add a product
- The staff login
- Updating of a customer

- Updating of a product

Security, privacy and legal issues

When it comes to commercial websites there are a variety of security threats which can occur. The reason for these threats is that there is a lot of data about customers such as personal information. Another reason could be political controversy. There are a few people/groups who can pose a threat and those are governments, criminals, activists and so on.

Criminals can pose a threat because they could get a customer's information from a website such as credit card details, bank details, address etc. Another way criminals could be a threat is that they could use a person's email account to send spam to others. Also, criminals can extort a person by ransomware and this works by locking a person's personal data. In order to unlock this data they will have to pay a specific amount of money which will go to that criminal and it doesn't necessarily mean that they will get their personal data back.

Governments can also pose a threat because they could be snooping on their own citizen's personal data or another countries citizens. Another reason they could pose a threat is because they could decide to put legislations in place which could result in weakened security. The reason for this is so that they can facilitate their own access meaning they could get their hands on a lot of personal information.

There are also activists who could pose a threat. These could be 'terrorists' or 'freedom fighters' and they could perform DoS (Denial-of-service) attack, backdoor access, firewalls and so on. The reason they may attack a website is so that they can deface it to raise awareness of political issues. Another reason could be that they want to spy on businesses (tech companies in particular) and media companies.

There are different types of attacks which can be used against commercial websites. For example, physical attacks, denial of service attack, compromising computers and so on. A physical attack may be the easiest one to pull off but it is also a risky one. This is because a person has to walk into the company's headquarters and plug into the network or an employee's computer. Once this is done they simply have to walk out. The risk to this is that they need to get through security, into the network/computer and leave without being seen. There are a few ways in which physical attacks can be prevented for example, having various physical access points so that it is harder to tailgate personnel. Another way to prevent physical attacks is to make sure information about employees which can be accessed outside the company is kept to the minimal so that it's harder to create a fake ID.

Another form of attack is a denial of service attack. This attack is used to block network communications by crashing/flooding servers. This is done by sending a malformed packet to the server which will then result in it crashing. To prevent this type of attack happening a user or the company should ensure that the latest operating system is installed and ensure that the latest patches are installed.

If a person wanted to perform another type of attack it could be compromising a computer. They could do this using a backdoor. A backdoor is essentially used to install malicious software onto a computer which can be done by the government, attackers, viruses and so on. This may be called a rootkit sometimes and rootkits can be hidden within the operating system which means it can be hard to remove or even detect. The reason this is a problem is because it can result in ransomware, store illegal files without the person knowing, record keystrokes (passwords) and more. A way to prevent a backdoor being used on a system is to have a firewall in place because it will block entry points apart from ones which are from authorised users. It can also prevent malware communication meaning a malware will not be able to communicate online e.g. sending keystrokes to another person.

There are people who may be concerned about their privacy and others not so much. The best way a person can ensure their privacy is to not share their data online, don't use social media or make any accounts on websites. To be anonymous online people can use Tor. Tor is a P2P software which routes HTTP requests through a network of machines which allows the user to access websites without them worried about being tracked. It is used by many people such as governments, journalists, criminals and more. However, the downside to Tor is that if it is used to create an account with personal information then there will be no privacy protection.

There is also the matter of customer data stored on websites when it comes to privacy. For example, if a website gets hacked and it has a lot of information about their customer then it will cause a huge problem because it can result in customers losing faith in this website and the website losing money. Also, the customer data could be used to commit fraud. Therefore it is a good idea to store as minimal data about customers as possible because it will increase their privacy and have minimal damage if a hack occurs. Another way could be to ensure that there is some sort of feature which temporarily locks the account when the password is entered incorrectly a certain amount of times so that someone cannot try multiple times and manage to get into the account and obtain personal details.

There can be certain legal issues when it comes to online websites. Some of them are theft of customer data from the websites database, illegal content could be uploaded by customers, data could be shared with the government by a cloud provider and more. Due to these legal issues which could occur it is important that legal liability is clear as there is a high chance these things could happen.

Legal issues mostly occur when it concerns personal data. This is because if personal data gets stolen then it can cause a lot of problems such as identity theft, loss of privacy, impact on the business and so on. If data loss occurs then it could result in individuals being compensated for harm and distress and the business could lose its customers due to the fact that they may not trust it anymore which essentially means the company has lost its reputation. Also, the business may incur penalties from the information commissioner.

Another legal issue could be concerning cloud services. This is because a user doesn't entirely have control over data management as the cloud provider has some responsibility. Due to this the government could request cloud providers to give data and the user will most likely not be informed about it and they will not be asked for permission.

Security, privacy and legal Issues which affect the e-commerce website

There are security, privacy and legal issues concerning the e-commerce website my group has created. The website is one which sells DVDs and to make sales customers need to enter their personal details such as name, address, credit card information and so on. As there is personal information certain security and privacy need to be considered.

Security is a major part when it comes to the e-commerce website. Since it's a website which sells DVDs meaning there is customer information could result in it being a target for hackers. Hackers will want to get into the website because they could steal credit card details, email address and more. If they get the email address they could use that email account to send spam. If they get other information such as credit card details then they could make unauthorised purchases and this could mean loss of money and company reputation resulting in users not using the e-commerce website.

Another security issue could be government putting legislations in place to weaken the e-commerce websites security. This could result in it being easier for hackers to get into the website or the government themselves. Once they do this the government could snoop on the activity of customers on the website or hackers could get their hands on personal details (as mentioned above).

A security issue for the e-commerce website could be activists performing a Denial of service attack on the website. Activists may do this to deface the website for their own gain which could be raising awareness of political issues or something else. If they do this they could spam the website, spy on the users of the website even find out their location which is a huge security risk.

In order to battle these security issues we should ensure that the computers which the websites runs off have firewall enabled and the operating system is up to date along with the latest security updates installed. If these actions are taken then the risk of being attacked/hacked is minimal and no customer data will be stolen.

There are also privacy issues concerning the e-commerce website. For example, a lot of people don’t want their data shared with other companies while there are other people who have no issue whatsoever having their data shared. In terms of the e-commerce website uses will be sharing their personal data to make a purchase.

When a credit card is used to make a payment the location of the purchase and be tracked along with what the purchase was. If a security breach occurs on the e-commerce website and credit card details are stolen then it could lead to a lot of money being loss, email address, home address and so on being stolen. On the other hand, if there is a cash security breach then it would only result in a finite amount of money being lost however, since it is an e-commerce website credit card has to be used. If something like this does happen then the e-commerce website will lose its reputation resulting in customer loss and money loss. To ensure the privacy of customers the security measures which are mentioned above can be taken so that no problems will occur which violates customers privacy.

Also, there are legal issues concerning the e-commerce website. As customer data is stored in the database it could be a target for hackers as they can steal customer data. Due to this the e-commerce website could be liable if they do not have the correct security measures in place ensuring the security of the website. However, legal liability needs to be made clear to the users before anything like this happens to protect the website.

Another issue could be a cloud provider sharing customer details with the government without us knowing or without customer's knowledge. They may do this because they must respond to a subpoena or for another reason. To make sure no issues occur between the customers and the e-commerce website it should be clearly stated how information will be shared via a contract with the service provider.