

# Creating Accessible Websites

---

## Table of Contents

Why make websites accessible?.....	3
Legislation .....	3
Authoring Tools .....	3
Please note.....	4
Appropriate Use of Colour.....	4
Colours and Cascading Style Sheets (C.S.S.).....	4
Colour Contrast.....	5
Colour Contrast Checkers .....	5
Formatting Text .....	6
Semantic structure.....	6
Fonts.....	6
Headings .....	7
HTML Code.....	7
Using WebPublish.....	7
Navigation and Hyperlinks.....	8
Navigation.....	8
Hyperlinks .....	8
Using Images .....	9
Alternative Text.....	9
HTML Code.....	9
Using WebPublish.....	9
How to Create Good Alternate Text.....	9
Complex Images.....	10
Charts or Graphs.....	10

Infographics .....	12
Flowcharts.....	12
Math Equations.....	12
Using Images and Alt tags .....	12
MathML .....	12
Tables .....	13
Best Practices for using Tables .....	13
Inserting a Table using WebPublish .....	15
Forms.....	15
Labels.....	15
HTML code .....	16
Tab order.....	16
HTML Code.....	16
Video Captions and Audio Transcripts .....	16
Accessibility Tips: for Video Captioning and Audio Transcripts .....	16
Video Captions .....	16
Audio Transcripts.....	17
Tips.....	17
Examples.....	17
Text transcript of audio file .....	18
Accessibility Checkers.....	18
Resources .....	18
External Resources .....	18

## Why make websites accessible?

Making accessible websites ensures that they are usable by the widest range of users, but also ensures your site is easier to read and navigate. It is important to make these changes to your websites to accommodate a variety of disabilities. For example, many people with visual disabilities use screen readers which read aloud information on the screen such as text or image descriptions provided through alternative text (Alt Text).

If you plan, format, and structure your website correctly in the beginning, it will ensure the site is not only accessible but can also be viewed and the content is usable for everyone.

[The Web Content Accessibility Guidelines \(WCAG\)](#) explains how to make web content accessible to people with disabilities. There are four guiding principles (P.O.U.R.):

1. **Perceivable** - Information and user interface must be presented to users in ways they can perceive.
2. **Operable** - Users must be able to use the interface. It cannot require interaction that a user cannot perform.
3. **Understandable** - Users must be able to understand the information as well as how to use the interface.
4. **Robust** – Users must be able to access the content as technologies advance. In other words, as technologies and user agents evolve, the content should remain accessible.

## Legislation

[Ontario Regulation 191/11, section 14:](#)

- By January 1, 2014, new internet websites and web content on those sites must conform with WCAG 2.0 Level A.
- As of January 1, 2021, all internet websites and web content must conform with WCAG 2.0 Level AA, other than, success criteria 1.2.4 Captions (Live), and success criteria 1.2.5 Audio Descriptions (Pre-recorded).

## Authoring Tools

There are many tools used to author html code for websites pages. Queen's University uses the web publishing tool WebPublish. Even though several web publishing tools and WYSIWYG webpage editors exist, it is important to note that using these tools

alone does not guarantee accessibility. There are many principles and practices, listed in this guide, to employ to ensure proper website accessibility.

More information is available for [WebPublish](#) or [The Web Standards and Accessibility Development Guide](#).

## **Please note**

For this tutorial we are focusing on webpage accessibility principles and practices using html code and WebPublish. If your website contains or is the following:

- Documents (e.g. PDF, MS Word) please see "[Accessible Documents](#)" tutorial.
- Video please refer to "[Video Accessibility](#)" how-to
- Social Media refer to "[Social Media Accessibility](#)" how-to.
- An online course, refer to "[onQ Accessibility](#)" how-to.

## **Appropriate Use of Colour**

When using colour, you must make sure that any information conveyed with colour is also conveyed by another means. For example, including text with colour-coded icons will help to clarify the function of the icon. Marking required form fields in another way (either with an \* before the field, using the word "required" after the form-field label) or using appropriate headers and contextual navigation in each distinct section of the web site will help to alleviate these colour-reliant issues.

Consider a form that indicates "all required fields are marked in red," a page that states "click the green button to continue," or a web-based report that shows status icons for transactions that differ only in their colour. In each of these cases, someone that may not be able to distinctly recognize certain colours because of their monitor settings, because of colorblindness, or because he/she is using a device that doesn't display colours would experience difficulty getting the information or completing the task at hand.

## **Colours and Cascading Style Sheets (C.S.S.)**

Using style sheets for presentation is not without pitfalls. Because there are so many interacting style rules that are required to create a full web site, colours must be paired when they are declared in a style sheet. This ensures that there are always contrasting colours when the style sheet is applied to a document and, when the style sheet is not there or is overridden, that both foreground and background are affected. Design errors occur when text colour is determined by the style sheet while the background colour is

declared directly in the HTML. As an example, when Cascading Style Sheets are turned off (via the user's browser settings) or not supported (alternative browser), this could result in default blue-coloured links laid out on a dark-coloured background, making the links impossible to read.

Thus, all colours should be declared in pairs in style sheets. Such a declaration may look like this:

```
a {color: red; background-color: transparent;}
```

## Colour Contrast

You must provide high colour contrast to the text in your document. A good example of high colour contrast is black and white; while an example of poor colour contrast is light yellow and white.

Avoid blinking or scrolling text.

[WCAG 2.0 Guideline 1.4.3](#): The visual presentation of text and images of text has a contrast ratio of at least 4.5:1, except for the following: (Level AA)

- **Large Text:** Large-scale text and images of large-scale text have a contrast ratio of at least 3:1;
- **Incidental:** Text or images of text that are part of an inactive user interface component, that are pure decoration, that are not visible to anyone, or that are part of a picture that contains significant other visual content, have no contrast requirement.
- **Logotypes:** Text that is part of a logo or brand name has no minimum contrast requirement.

## Colour Contrast Checkers

- [WebAim - Color Contrast Checker](#)
- [WCAG Contrast checker](#) (Firefox plug-in)
- [The Colour Contrast Analyser for Web pages](#)
- [Colour Contrast Check](#)

## Formatting Text

### Semantic structure

Preparing content for the web is much more than just copying text from your favourite word processor and pasting it into the framework of a web page. In order to ensure the greatest accessibility and widespread availability of web content, it is important to consider that there is more to the web page than the words and pictures that end up in a person's browser.

Web pages are built with HTML, a structural language that is used to denote the different structural parts of the document. HTML employs an extensive series of opening and closing tags to mark parts of the document as, for example, headings (`<h1></h1>`), unordered lists (`<ul></ul>`), ordered lists (`<ol></ol>`), and paragraphs (`<p></p>`). These tags don't appear in the final view of the web page, but are used by the various web software applications to interpret the intended format of the text for output to the end user.

To help clarify the term semantic further, consider that some HTML tags (such as the `<strong>` for bold text) are not considered semantic elements but are presentational elements, i.e. while they do alter the visual presentation of the text, they don't define the actual structure of the text.

Using semantically structured HTML and then changing the way that the HTML looks with Cascading Style Sheets results in the final version of the web page that looks aesthetically pleasing to people viewing the web page in a browser, and is logically structured, providing meaning to those that need it. Pages built this way are not only accessible to the largest possible audience, but are also well set for future revision of both style and content, and can be easily adapted to accommodate emerging technology and advancing web development standards.

### Fonts

- The need to separate content from design is the underlying foundation for accessibility, scalability, and future compatibility of all of your web content. For this reason, size declarations should be recorded in a site's style sheet, and not in the content page.
- Try to use fonts that are legible and easier to read such as Verdana, Tahoma, Lucida Grande, Arial, or Georgia.
- Avoid font sizes smaller than 9pt as they are difficult to read.
- Avoid italics and underlining text other than hyperlinks.

- Use <strong></strong> tags for bolded text.

## Headings

Using Headings makes it easier for various adaptive technologies to navigate a document. Many people do not create Headings correctly, either making font sizes bigger or in bold rather than using the formats already provided by WebPublish. Headings need to be marked up in the html code. Heading levels should have a meaningful hierarchy and it is recommended to reserve H1 for the page title, H2 for major headings and H3 for major sub headings.

## HTML Code

```
<h1>Queen's University</h1>
```

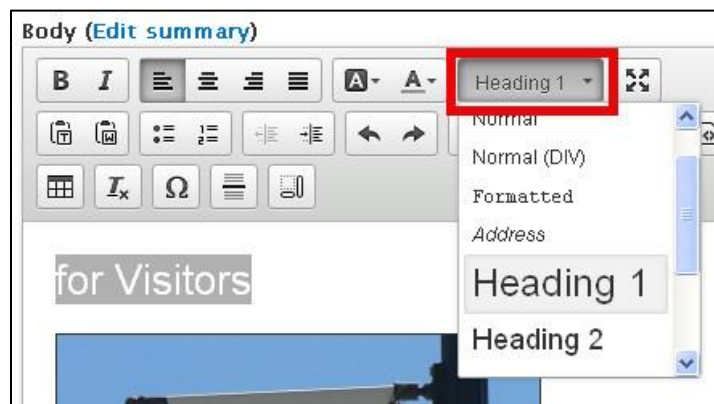
```
<p>Located in Kingston, Ontario...</p>
```

```
<h2>Weather</h2>
```

## Using WebPublish

While you're editing a draft:

- 1) Select the text you would like to make into a heading.
- 2) Select the desired format from the format menu drop-down list.



## Navigation and Hyperlinks

### Navigation

Navigation and links are the features that make the web such a pervasive medium. Almost any web site will employ the use of navigational links - either within the same web document, within the same web site, or to another web site.

- It is important to all users that links be clearly marked and distinguishable from the rest of the items on the page.
- Do not underline text for emphasis when that text is not hyperlinked. To web users, underlined text signifies a link. When required, use **bolding** or *italics* to emphasize text.
- Placing navigational links in an unordered list provides structure to the document and "groups" the links together, establishing a relationship between the items. This can be achieved within the Cascading Style Sheet.
- Avoid using image rollovers for navigation with images of text.

### Hyperlinks

To link your document to a website or another document, you may use hyperlinks. Below are some tips that web pages at Queen's University should follow:

- **Ensure that the hyperlink has context and describes where it leads.** It should not just read "click here", and should make it clear what the destination of the link is (example, the web link [www.queensu.ca](http://www.queensu.ca) should be written as "[Queen's University](http://www.queensu.ca)").
- **Avoid language that requires spatial or visual references.** One of the greatest powers of the Internet is the ability to cross link related documents, allowing the reader to explore and learn in a non-linear fashion. Content authors must remember, however, that not all users will be experiencing their web pages the same way that the author may be. Instructing readers to click on a 'link on the right,' for example, does not take into consideration the fact that visually impaired users do not see left or right. The same holds true for references to colours or shapes (i.e. click on the 'round' button, or the 'green' text).
- **When linking to a file, indicate the format and document size** for example: Accessible Documents (PDF 75KB).
- **Hyperlinks should not open in a new browser window.** Programming a link to open in a new browser window can cause difficulties with novice users for a variety of reasons (i.e. the new window does not retain the current window's



browser history and disables the 'back' button function, or the launching of multiple pop-up windows may be blocked by a third-party application), and can disorient people who are using screen-reading or screen-magnification software.

## Using Images

### Alternative Text

Alternative text must give an accurate description of what the item is, so that if a screen reader or refreshable braille display is used, the “Alt tag” will convey the text alternative of the image.


To add alternative text:

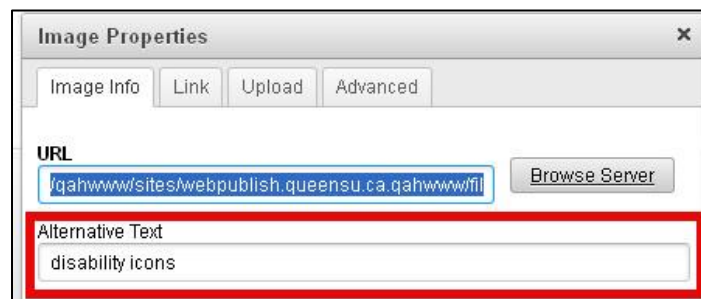
### HTML Code

```

```

### Using WebPublish

1. First to add an image to the page select the “Image” button 
2. A new window open called “Image Properties”. To add an image from the server select the “Browse Server” button to select or upload an image for use.
3. Once an image is selected enter the alternate text in the “Alternate Text field of the “Image Properties dialogue box.



### How to Create Good Alternate Text

- Consider the content and function of your image.

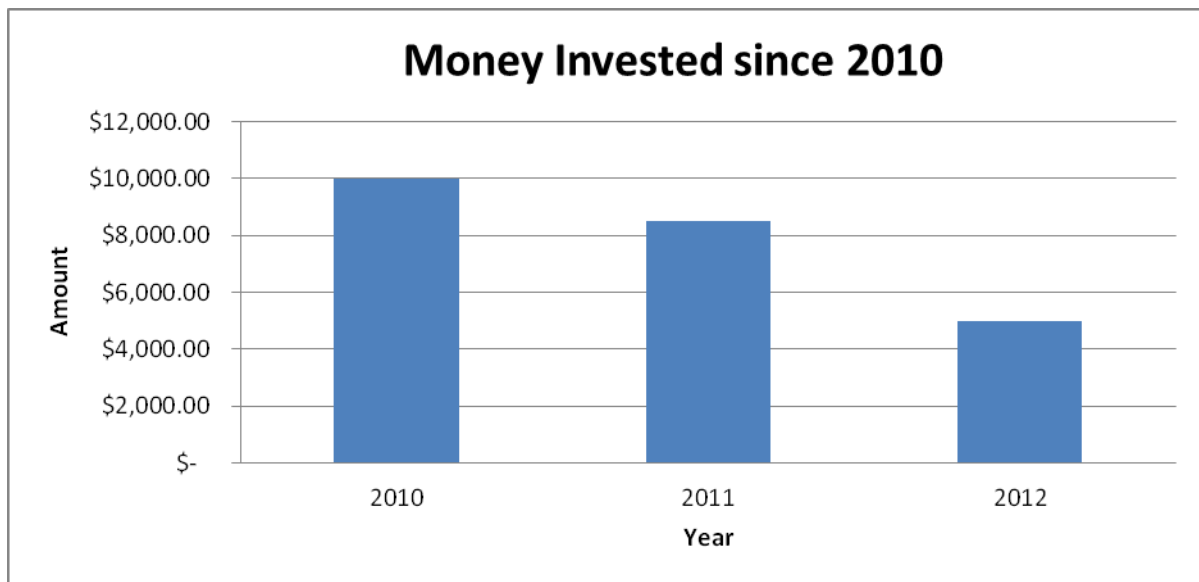
- If it provides content to your document, make sure that the information the image provides is described in the alt text.
- If your image only provides a function (for example, providing a portrait of a historical figure described in the text) you need only describe the image. In the case that the image is of a historical figure, write his/her name as the alt text.
- Try not to use “Image of...” or “Graphic of...” as alt text. That is usually evident to the person reading the alt text.
- Do not repeat the information which is contained in the document itself into the alt text. If it's already in the document, that should be enough.
- Images used for a bullet or spacer should have an empty Alt tag i.e. alt=""

## Complex Images

### Charts or Graphs

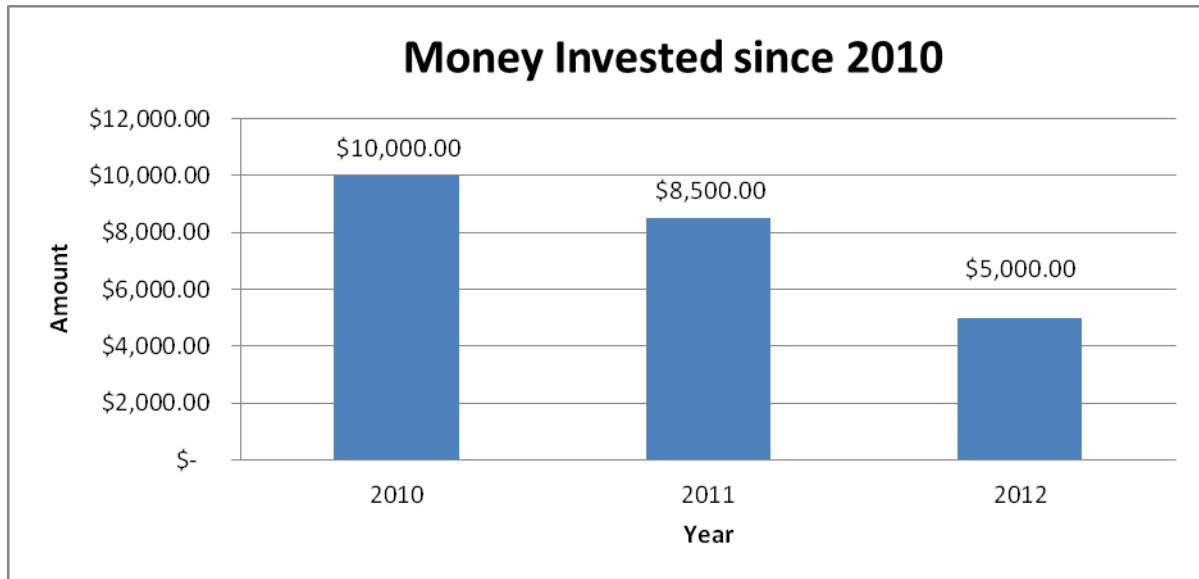
Usually a graph on a webpage is an image file placed using html code. As such, an appropriate Alt tag should be included. However, on its own, an alt tag doesn't do a graph justice.

The following bar graph with the alt tag alt="graph of money invested since 2010" does not convey all the information represented by the graph.



In order to serve a wider audience and improve accessibility of a graph it is wise to include a text summary (perhaps describing trends) and a properly coded table ([see Table section](#)) of the data placed near the graph.

The improved graph has a title, axis labels, data labels, alt text, a summary, and table of data.



In html code `<img src=graph.jpg" alt="bar graph of money invested in 2010, 2011, and 2012" width="300" height="175" />`

Summary:

The bar graph shows that every year between and including 2010 and 2012, the money that has been invested has decreased 50%.

Data Table

Amount	Year
\$10,000.00	2010
\$ 8,500.00	2011
\$5,000.00	2012

## Infographics

These images can hold massive quantities of information in a visual way. For a person who is blind or with low vision these images are troublesome. The information contained within an infographic is too large for an Alt tag. Therefore, a text equivalent is required either below the image or on a separate web page.

[ITServices has a page that demonstrates this technique.](#) The images are present to be viewed as thumbnails or PDF files but a link containing a text version is also present.

## Flowcharts

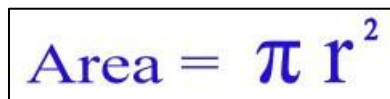
Another type of graphic is a flowchart or concept map that shows relationships between ideas or concepts. Providing a text based outline of the diagram assists with accessibility.

## Math Equations

Equations and scientific notations can be handled by using images with accompanied Alt tags for simple less complex equations to using MathML for the complex. MathML is an XML markup language designed to display complex mathematical expressions (e.g. fractions, square roots, matrices, bounded integrals). The theoretical advantage of MathML is that equations are represented as text instead of images.

## Using Images and Alt tags

If your website used an image for the formula to calculate the area of a circle it might look like the following:


$$\text{Area} = \pi r^2$$

For someone using a screen reader they have no idea whether the image contains a picture or a formula. Since the formula is simple in nature an Alt tag would probably suffice. The alternative text for a formula can be written in [Nemeth MathSpeak](#) for anyone who has learned that system.

Alt="formula for area of a circle. Area equals pi dot radius sup 2"

## MathML

To create MathML code it is best to use to use an equation editor such as:

- MathType
- MathEQ Expression Editor

- MathMagic Personal Edition
- Scientific NoteBook Scientific
- Fire Math

One of the more common is MathType which can integrate with MS Office. Please see [“External Resources”](#) for more information and step-by-step instructions on using MathType.

## Tables

In websites, tables should be used to hold data and NOT used for design layout. Layout should be determined by a Cascading Style Sheet (css) with use of the <div> tag.

### Best Practices for using Tables

Simple data tables can easily be read by screen readers if they incorporate the <th> tag and “scope” attribute to identify which cells are row and column headers. More complex tables can incorporate the <caption> and <summary> tags.

Consider the following table:

Contact Information	
Person	Number
John Doe	(555) 555-5555

It is important to remember that screen reader read tables linearly. This table looks simple but is a little more complex when looking at the html code.

```
<table border="1" cellpadding="0" cellspacing="0" style="width: 500px;">
  <thead>
    <tr>
      <th colspan="2" scope="col">Contact Information</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>Person</th>
      <th>Number</th>
    </tr>
```

```

        <tr>
            <td>John Doe</td>
            <td>(555) 555-5555</td>
        </tr>
    </tbody>
</table>

```

The table uses the <th> tags to identify the headers content. Most browsers will render any text in the <th> tag as bold and centred. We can improve this table's usability by adding a <caption> and <summary>. The <caption> tag is used to give a table a title. The <summary> is only read by screen readers and not displayed visually and they provide brief summaries of complex data and don't need to be used for every table.

Now we add a caption and summary to our example:

Contact Information	
Person	Number
John Doe	(555) 555-5555

The html code:


```

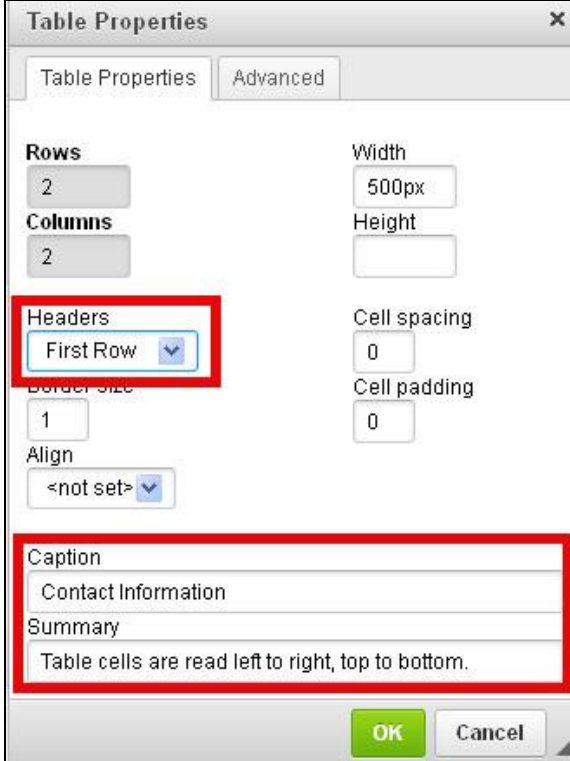
<table border="1" cellpadding="0" cellspacing="0" style="width: 500px;"
summary="Table cells are read left to right, top to bottom.">
    <caption>Contact Information</caption>
    <thead>
    </thead>
    <tbody>
        <tr>
            <th>Person</th>
            <th>Number</th>
        </tr>
        <tr>
            <td>John Doe</td>
            <td>(555) 555-5555</td>
        </tr>
    </tbody>
</table>

```

## Inserting a Table using WebPublish

In “Edit Draft” mode:

1. Place your cursor where you wish to insert a table.
2. The “Table” button from the menu 
3. This opens the “Table Properties” dialogue box where you can select which cells are headers, enter a caption, and a summary for complex tables.



**Table Properties**

Table Properties | Advanced

**Rows**  
2

**Columns**  
2

**Width**  
500px

**Height**

**Headers**  
First Row

**Cell spacing**  
0

**Cell padding**  
0

**Border size**  
1

**Align**  
<not set>

**Caption**  
Contact Information

**Summary**  
Table cells are read left to right, top to bottom.

OK | Cancel

## Forms

Online forms can be handy way to get assistance, search, request, or submit information to be sent somewhere else. They can also be accessibility nightmares if done incorrectly.

## Labels

Labels should be named and placed in a logical order. For example a form that asks you to input your name should have the <label> to the immediate left of the text-box. Labels should also be to the right of radio buttons and checkboxes.

A rectangular box containing the text "Name" on the left and a small rectangular text input field on the right.

### HTML code

```
<p><label for="name">Name</label>
```

```
<input type="text" name="name" id="name" /></p>
```

### Tab order

When navigating web forms anyone can use the “Tab” key to move from field to field. Someone using a screen reader will make good use of the Tab key. Most of the time, the “Tab” will move focus in the logical order of the form (top to bottom) but not always. To ensure a logical movement, you can set the “tab index” attribute for each form element to set the order.

### HTML Code

```
<p><label for="name">Name</label>
```

```
<input type="text" name="name" id="name" tabindex="1" /></p>
```

Any field that is required to be filled out should be labelled using proper symbols and text such as (\* required). If required fields are indication by a bold font or the colour red, a screen reader won’t be able to identify those fields as required.

If a “Submit” button is an image, ensure it has an Alt tag saying “Submit”.

## Video Captions and Audio Transcripts

### Accessibility Tips: for Video Captioning and Audio Transcripts

1. Avoid having an audio or video file play automatically on a Web page. Such a feature is potentially distracting for some users, and could interfere with some adaptive technology.
2. Visually impaired users may need additional information about images in a video.

### Video Captions

When video files are used in your website, **captions or a synchronized text transcript** should be provided. Captions not only provide good [Universal Instructional Design](#) but



are also useful for non-native speakers (for example, ESL), when the video has poor audio quality and when users view the video with audio disabled.

[WCAG 2.0 Guideline 1.2.2](#) — "Captions are provided for all prerecorded audio content in synchronized media, except when the media is a media alternative for text and is clearly labeled as such."

Refer to "[Video Accessibility](#)" for detailed instructions on captioning videos.

## Audio Transcripts

When you use audio files on your Web page, **a text transcript or other text-based material** should be provided.

[WCAG 2.0 Guideline 1.2.1](#) — "An alternative for time-based media is provided that presents equivalent information for prerecorded audio-only content."

## Tips

- If you have a script for an audio or video production, it can be the basis for a text transcript. Otherwise you may need to manually transcribe the text (i.e. play/pause/type into MS Word).
- Using speech recognition software can automate some transcription, but should be reviewed for errors and corrected. Example of speech recognition software are [Dragon Naturally Speaking \(Win/Mac\)](#), [Speech Recognition built-in to Windows](#) and [Dictation built-in to Mac OSX](#).
- Video files should be embedded or displayed in a player that can be accessed by a screen reader **by using keyboard commands**. Some accessible players include QuickTime, RealPlayer, YouTube and JW Player (version 6 full-screen mode).

[WCAG 2.0 Guideline 2.1](#) — "Make all functionality available from a keyboard."

- Avoid having an audio or video file play automatically on a Web page. Such a feature is potentially distracting for some users, and could interfere with speech recognition software.
- Visually impaired users may need additional information about images in a video.

## Examples

1. [Closed Captioning Videos - Select CC to turn on](#)

2. [Open Captioning - captions are already on when video plays](#)

### **Text transcript of audio file**

1. “My name is Melissa Vassallo and I'm a Disability Advocate. I'm a proud alumni of Queen's University.

Accessibility is a hard word for some people because they don't really know what it means. People are afraid of the word disability. Do I say I'm disabled? Do they bring it up? Do they ask me if I need help?

I just find that people are really unsure about what the world of disability and accessibility is all about and it's one that I welcome with open arms. I want the dialogue to be free-flowing and everyone feel that they have a space where they can feel comfortable talking about disability...”

2. [Audio Transcript of Koerner Artist in Residence Program, 2003-2009 Video \(PDF 32kb\)](#) - Queen's University BFA Programme

## **Accessibility Checkers**

- [Queen's ITS Web Content Accessibility Review](#)
- [WAVE Web Accessibility Tool](#)
- [Web Accessibility Checker](#)
- [The Colour Contrast Analyser for Web pages](#)
- [Web Accessibility Toolbar for IE](#)

## **Resources**

1. [Queen's ITS Accessibility](#)
2. [Queen's ITS Web Content Accessibility Review](#)

## **External Resources**

1. [WebAim](#) (website)
2. [Penn State AccessAbility – Website Accessibility](#)
3. [MathType with MS Word \(step-by-step guide\)](#)

4. [Accessible Math with Word 2010 and MathType](#) (YouTube- video is closed captioned)

---

**Queen's is committed to an inclusive campus community with accessible goods, services, and facilities that respect the dignity and independence of persons with disabilities. This document is available in an accessible format or with appropriate communication supports upon request.**

**Please contact the Accessibility Coordinator, Andrew Ashby, in one of the following ways:**

**Email: [accessibility.hub@queensu.ca](mailto:accessibility.hub@queensu.ca)**

**Phone: (613) 533-6000 ext. 75734**

**In person: Adaptive Technology Centre, Stauffer Library, Room 120E**