



## Instructions

You can take part in this challenge as a team or alone, you decide. You will gain points for each achievement and lose points for each hint bought.

You have 2 hours to create a Python script that downloads the HTML from <http://www.mcast.edu.mt> and identifies all the links in the page.

A link that fails to open for some reason is considered an error.

A link in the HTML can be found as follows: `xxxxxxxhref="LINK_IS_HERE"xxxxxxx`

Examples:

- `<link type="text/css" href="/Portals/_default/Skins/MCAST/skin.css">`
- `<link rel='SHORTCUT ICON' href='/Portals/0/favicon.ico' type='image/x-icon' />`
- `<a href="http://www.google.com">Google</a>`

## Scoring System

Below are the details of the points gained by completing each achievement.

Points	Achievement
100	Minimum Submission: <ul style="list-style-type: none"> <li>• Parse all the links in 1 HTML page, without crashing</li> <li>• Display all the working links in the console</li> <li>• Display number of working links and errors as a summary</li> </ul>
50	Parse all the links 2 levels deep
30	Parse more than 2 levels deep
30	Skip links that start with 'javascript:' and 'mailto:'
50	Expand relative paths, e.g.: /Portals/_default/default.css?cdv=50 becomes <a href="http://www.mcast.edu.mt/Portals/_default/default.css?cdv=50">http://www.mcast.edu.mt/Portals/_default/default.css?cdv=50</a>
30	Only investigate links that are under the domain <a href="http://www.mcast.edu.mt">www.mcast.edu.mt</a>
30	Only count unique links, no duplicates
20	Save output to a text file (as well as printing it to console)
10	Meaningful variable and function names, commented code

## Hints System

You can buy the hints below by 'paying' the respective amount of points.

Points	Hint
-20	How to get the HTML for a particular URL
-20	How to get all the links from an HTML string
-20	How to handle Errors (crashes)
-10	How to write to a text file