```
using System;
using System.Collections.ObjectModel;
using System.Net;
using OpenQA.Selenium;
using OpenQA.Selenium.Chrome;
using System.Text;
using OpenQA.Selenium.Support.UI;

namespace selenium
{
    class Program
    {
        //Yu Wen
        static void Main(string[] args)
        {
            IWebDriver driver = new ChromeDriver();

            //question3internalLinks(driver);

            //question6TradeMeInternalLinks(driver);

        }

        static void question3internalLinks(IWebDriver driver)
        {
            //opens website
            driver.Navigate().GoToUrl("http://automationpractice.com");
            //find links with tag a
            List<IWebElement> allLink = driver.FindElements(By.TagName("a")).ToList();
            //List<IWebElement> imgLink = driver.FindElements(By.TagName("img")).ToList();
            //allLink.AddRange(imgLink);
            String homepage = "http://automationpractice.com";

            for (int i = 0; i < allLink.Count; i++)
            {
                string url = allLink[i].GetAttribute("href");//get links

                if (url == string.Empty || url == null)
                {
                    //when it is empty (broken)
                    Console.Write((i + 1) + " ");
                    Console.WriteLine("Empty :" + allLink[i].Text);
                }
                else if (url.StartsWith("http://") || url.StartsWith("https://"))
                {
                    if (!url.StartsWith(homepage))
                    {
                        //external links
                        Console.Write((i + 1) + " ");
                        Console.WriteLine("external : " + url);
                    }
                    else
                    {
                        //valid internal links
                        Console.Write((i + 1) + " ");
                        Console.WriteLine("Valid : " + url);
                    }
                }
                else
                {
                    //for broken links
                    Uri uriResult;
                    bool result = Uri.TryCreate(url, UriKind.Absolute, out uriResult) && (uriResult.Scheme == Uri.UriSchemeHttp || uriResult.Scheme == Uri.UriSchemeHttps);
                    if (result != true)
                    {
                        Console.Write((i + 1) + " ");
                        Console.WriteLine("Broken link : " + url);
                    }
                }
            }

            Console.ReadKey();
            driver.Quit();
        }


        static void question6TradeMeInternalLinks(IWebDriver driver)
        {
            //open the webpage
            driver.Navigate().GoToUrl("https://www.trademe.co.nz/a/");

            //find links with tag a
            List<IWebElement> allLink = driver.FindElements(By.TagName("a")).ToList();

            //find links with tag img(images)
            //List<IWebElement> imgLink = driver.FindElements(By.TagName("img")).ToList();
            //allLink.AddRange(imgLink);
            string homepage = "https://www.trademe.co.nz/";

            for (int i = 0; i < allLink.Count; i++)
            {
                string link = allLink[i].GetAttribute("href");

                if (link == null || link == string.Empty)
                {
                    //when the url is not abiliable or empty
                    //string.empty is leading to somewhere in memory, null is not
                    continue;
                }
                else
                {
                    if (link.StartsWith(homepage))//from internal
                    {
                        //find different links
                        if (link.Contains("property"))
                        {
                            Console.WriteLine("property link " + link);
                        }
                        else if (link.Contains("service"))
                        {
                            Console.WriteLine("service link " + link);
                        }
                    }
                }
            }

            //output
            Console.ReadKey();
            driver.Quit();
        }

    }
}
```