[1315] Last Week's Assignment with LINQ

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;

namespace AccessLogAnalyzer
{

    class LogItem
    {
        public string IP { get; private set; }
        public string Path { get; private set; }
        public string Status { get; private set; }

        public LogItem(string ip, string path, string status)
        {
            this.IP = ip;
            this.Path = path;
            this.Status = status;
        }

    }

    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Access Log Analyzer (LINQ)");
            Console.WriteLine("==========================\n");

            var logData = new List<LogItem>();

            // Read the Access Log File
            StreamReader sr = new StreamReader("access-log.txt");
            while (!sr.EndOfStream)
            {
                string line = sr.ReadLine();

                line = line.Replace(" HTTP/1.0", "");
                line = line.Replace(" HTTP/1.1", "");
                line = line.Replace("\"", "");

                string[] tokens = line.Split(' ');

                string ip = tokens[0];
                string url = tokens[6];
                string path = url.Split('?')[0];
                string status = tokens[7];
                LogItem logitem = new LogItem(ip, path, status);
                logData.Add(logitem);

            }

            // Display Summary Stats

            Console.WriteLine("\nStatus Frequencies:");
            Console.WriteLine("===================");
            var statusCounts = from s in logData
                               group s by s.Status into summary
                               orderby summary.Count() descending
                               select new
                               {
                                   Status = summary.Key,
                                   StatusCount = summary.Count()
                               };

            foreach (var item in statusCounts)
                Console.WriteLine(item.Status + " - " + item.StatusCount);

            Console.WriteLine("\nIP Frequencies:");
            Console.WriteLine("===============");
            var ipCounts = from s in logData
                           group s by s.IP into summary
                           where summary.Count() > 9
                           orderby summary.Count() descending
                           select new
                           {
                               IP = summary.Key,
                               IpCount = summary.Count()
                           };

            foreach (var item in ipCounts)
                Console.WriteLine(item.IP + "\t- " + item.IpCount);


            Console.WriteLine("\nPath Frequencies:");
            Console.WriteLine("================");
            var pathCounts = from s in logData
                            group s by s.Path into summary
                            where summary.Count() > 9
                            orderby summary.Count() descending
                            select new
                            {
                                Path = summary.Key,
                                PathCount = summary.Count()
                            };

            foreach (var item in pathCounts)
                Console.WriteLine(item.PathCount + "\t" + item.Path);

        }

    }

}
```