```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using JerryWebMVC.Classes;
using System.IO;
using System.Text;
using System.Web.Mvc;

namespace JerryWebMVC.Models
{
    public class WaterModel
    {
        public WaterModel()
        {
            Mode = "EMPTY";
            Err = "";
            InitLists();
        }

        public void AutoWaterTransit(int numrows, int numcols, bool allintegers)
        {
            WaterTransit wt = new WaterTransit();
            wt.AutoMatrix(numrows, numcols, allintegers);

            Matrix = wt.Matrix;
            TransitResult = wt.ProcessRoutes() ? "Yes" : "No";
            RowRoute = wt.RowRoute();
            TotalWeight = wt.Weight();
            LeastRoute = getLeastRoute(wt);
            Mode = "Matrix";
        }

        private string getLeastRoute(WaterTransit wt)
        {
            StringBuilder sb = new StringBuilder();
            foreach (var step in wt.LeastResitant().Steps)
            {

                sb.Append("rc-" + step.row.ToString() +step.col.ToString()+";");

            }
            return sb.ToString();

        }

        private int CountLine(string line)
        {

            int count = 0;
            char[] ch = new char[2] {',','\n'};
            int pos = line.IndexOfAny(ch);

            while (! string.IsNullOrEmpty(line))
            {
                count++;
                line = pos==-1 ? string.Empty : line.Substring(pos+1);

                pos = line.IndexOfAny(ch);
            }

            return count;

        }

        public bool CheckRunFile(string file)
        {
            FileStream stream = new FileStream(file, FileMode.Open);
            StreamReader reader = new StreamReader(stream);

            try
            {
                string line= reader.ReadLine();
                int maxCol = CountLine(line);
                int maxRow = 1;
                while (!reader.EndOfStream)
                {
                    maxRow++;

                    line = reader.ReadLine();
                    if (CountLine(line) != maxCol)
                    {
                        throw new Exception("File has vary column numbers");
                    }
                }

                if ((maxCol >100) || (maxRow >10 ))
                {
                    throw new Exception("File exceeds maximum columns or rows");
                }
                reader.Close();
               RunFile(file,maxRow,maxCol);
                return true;
            } catch (Exception E)
            {
                Mode = "Error";
                Err = E.Message;
                reader.Close();
                return false;
            }
        }

        public void RunFile(string file, int rownum, int colnum)
        {
            WaterTransit wt = new WaterTransit();
            wt.LoadMatrix(file,rownum,colnum);

            Matrix = wt.Matrix;
            TransitResult = wt.ProcessRoutes() ? "Yes" : "No";
            RowRoute = wt.RowRoute();
            TotalWeight = wt.Weight();
            LeastRoute = getLeastRoute(wt);
            Mode = "Matrix";

        }

        public void InitLists()
        {
            RowList = new List<SelectListItem>();

            for (int i = 1; i < 11; i++)
            {
                RowList.Add(new SelectListItem() { Text = i.ToString(), Value = i.ToString↙
    () });

            }


            ColList = new List<SelectListItem>();

            for (int i = 1; i < 101; i++)
            {
                  ColList.Add(new SelectListItem() { Text = i.ToString(), Value = i.      ↙
    ToString() });
```

```
            }



        }

        public string TransitResult { get; set; }
        public string Mode { get; set; }
        public string Err {get; set;}
        public string TotalWeight { get; set; }
        public string RowRoute { get; set; }
        public string LeastRoute { get; set; }
        public int[,] Matrix { get; set;  }
        public List<SelectListItem> RowList { get; set; }
        public List<SelectListItem> ColList { get; set; }
        public string RowNum { get; set; }
        public string ColNum { get; set; }
    }
}
```