Serialization Quick Reference

Programming Fundamentals with C#

SPRING SEMESTER 2018

horizontal line

## Employee Class

Below is the class that we will be saving and loading in the following code examples.

**[Serializable]**

**public class Employee {**

**public int EmployeeNumber { get; set; }**

**public string Name { get; set; }**

**public double Salary { get; set; }**

**}**

## Binary Files

Below are methods for saving and loading binary files.

public static bool SaveBinaryFile(string path, List<Employee> employees) {

using (var stream = new FileStream(path, FileMode.Create, FileAccess.Write)) {

**var f = new BinaryFormatter();**

**f.Serialize(stream, employees);**

**stream.Flush();**

return true;

}

}

public static List<Employee> LoadBinaryFile(string path) {

using (var stream = new FileStream(path, FileMode.Open, FileAccess.Read)) {

**var f = new BinaryFormatter();**

**List<Employee> employees = (List<Employee>)f.Deserialize(stream);**

**return employees;**

}

}

## JSON Files

Below are methods for saving and loading JSON files. ***(Remember to add a reference to the System.Web.Extensions assembly.)***

public static bool SaveJsonFile(string path, List<Employee> employees) {

using (var stream = new FileStream(path, FileMode.Create, FileAccess.Write)) {

using (var writer = new StreamWriter(stream)) {

**var s = new JavaScriptSerializer();**

**string json = ;**

**writer.WriteLine(json);**

**writer.Flush();**

return true;

}

}

}

public static List<Employee> LoadJsonFile(string path) {

using (var stream = new FileStream(path, FileMode.Open, FileAccess.Read)) {

using (var reader = new StreamReader(stream)) {

**var s = new JavaScriptSerializer();**

**string json = reader.ReadToEnd();**

**List<Employee> employees = s.Deserialize<List<Employee>>(json);**

return employees;

}

}

}

## 

## 

## CSV Files

Below is a method for saving a CSV file.

public static bool SaveCsvFile(string path, List<Employee> employees) {

**StringBuilder sb = new StringBuilder();**

using (var stream = new FileStream(path, FileMode.Create, FileAccess.Write)) {

using (var writer = new StreamWriter(stream)) {

**foreach (Employee emp in employees) {**

**sb.Clear();**

**sb.Append(emp.EmployeeNumber);**

**sb.Append(CSV\_DELIM);**

**sb.Append(emp.Name);**

**sb.Append(CSV\_DELIM);**

**sb.Append(emp.Salary.ToString("0.00"));**

**writer.WriteLine(sb.ToString());**

**}**

writer.Flush();

return true;

}

}

}

## CSV Files

Below is a method for loading a CSV file.

public static List<Employee> LoadCsvFile(string path) {

**List<Employee> employees = new List<Employee>();**

using (var stream = new FileStream(path, FileMode.Open, FileAccess.Read)) {

using (var reader = new StreamReader(stream)) {

**string line;**

**while ((line = reader.ReadLine()) != null) {**

**string[] fields = line.Split(CSV\_DELIM);**

**Employee emp = new Employee();**

**emp.EmployeeNumber = int.Parse(fields[0]);**

**emp.Name = fields[1];**

**emp.Salary = double.Parse(fields[2]);**

**employees.Add(emp);**

**}**

}

}

return employees;

}