

Name: _____

Date: _____

Lab 3 – Serialization

Objectives

Part 1 – Functional Requirements

Reflection

Background/Scenario

In Lab 2, you had created a system where you could view a list of fonts installed on your system, mark as favorite, and leave comments. But, if you closed and reopened the application, none of the data was saved. The application did not have any functionality to persist its state.

In this lab, you will be using serialization to save the application's state so next time you close and reopen the application, all your favorites and comments remain.

The .NET framework has several built-in classes that let you easily serialize and deserialize objects with different formatters. You will explore serialization by updating the existing code you wrote from Lab 2 in order to save the application's state.

Required Resources

- Visual Studio
- Lab 2

Part 1: Functional Requirements

Summary

The goal of this lab is to extend the functionality of Lab 2 by allowing serialization of the application data so the data is persisted across subsequent sessions. The application will allow the user to save and load their data which should restore the application's state to how it was before the application was closed.

Technical Requirements

- The serialization format is of your choice to implement. (e.g. Binary, XML, etc...)
- The application must implement serialization and deserialization for two purposes:

Application Preferences

- The following properties must be serialized:
 - Window location
 - Window size
 - Path to last saved font information
- Preferences must be loaded on application launch if exists otherwise use default values.
- If path to last saved font information exists, load the data.
- Preferences must be saved on application exit.
- Preferences file must be saved in application directory.

Font Information

- The following properties must be serialized:
 - Font name
 - Font favorite status
 - Font comments
- Only fonts that are favorites and/or have comments must be serialized. Do not serialize any fonts that are not favorites or do not have comments.

User Interface Requirements

- The application UI must provide a way to:
 - Save the current font information to a file.
 - Load font information from a file overwriting the current font information context.
 - Clear the current font information
- An open and save file dialog box must be used to select the file.
- The file dialog boxes must have the extension filter set.

Optional Extra Credit

- Implement both Save and Save As for the font information.
- A method to allow the user to export a list of ALL the fonts including font name, favorite, and comment information as a comma separated value (csv file) to be able to be viewed in Excel. Make sure the csv has headers.

Reflection

1. Which serialization format did you select for this lab and why did you choose it over the other formats?
