UnityEngine.Sys API Documentation v1.0.4b

Logging/Debugging Methods

---

public static void Log(string message)
-Sends a debug message to the Sys.Log that can be exported to a text file when 'SaveLog()' is called.
PARAMETERS:
message - the debug message to be displayed in the exported Sys.Log file.
USEAGE:
Sys.Log("This is a test!");

---

public static void Log(string message, bool showInConsole)
-Sends a debug message to the Sys.Log that can be exported to a text file when 'SaveLog()' is called.
-When showInConsole is true the debug message will be appear debug to console (via 'Debug.Log' method).
PARAMETERS:
message - the debug message to be displayed in the exported Sys.Log file.
showInConsole - if true, the Sys.Log message will also be send to the console via Debug.Log method.
USEAGE:
Sys.Log("This is a test!" , true);

---

public static void ClearLog()
-Clears the Sys.Log cache.
USEAGE:
Sys.ClearLog();

---

public static void SaveLog()
-Saves Sys.Log data to default location: ' Application.persistentDataPath + "/" + Application.productName + "/Logs/SysLog.txt" '.
USEAGE:
Sys.SaveLog();

---

public static void SaveLog(string path)
-Saves Sys.Log data to a specified path.

PARAMETERS:

path - desired path to save the Sys.Log as a text file.

USEAGE:

`Sys.SaveLog("Path_To_Log_File/SysLog.txt");`

---

public static void SaveLog(string path, bool openDirectory)

-Saves Sys.Log data to a specified path then opens file directory upon save.

PARAMETERS:

path - desired path to save the Sys.Log as a text file.

openDirectory - if true, File Explorer will open the containing folder of your saved
file (Desktop Only).

USEAGE:

`Sys.SaveLog("Path_To_Log_File/SysLog.txt", true);`

Data Saving Methods

---

public static void SaveDataToFile(string path, string[] data)

-Saves string array to a specified path. (Path must include filename and extension.)

PARAMETERS:

path - desired path to save the text file.

data - the string array to be saved to text file.

USEAGE:

```
public string[] data = new string[]{"a","b","c"};
        Sys.SaveDataToFile("Path_To_Text_File/SomeFile.txt", data);
```

---

public static void SaveDataToFile(string path, string[] data, bool openDirectory)

-Saves string array to a specified path then opens file directory upon save. (Path must include filename and extension.)

PARAMETERS:

path - desired path to save the text file.

data - the string array to be saved to text file.

openDirectory - if true, File Explorer will open the containing folder of your saved file (Desktop Only).

USEAGE:

```
public string[] data = new string[]{"a","b","c"};
        Sys.SaveDataToFile("Path_To_Text_File/SomeFile.txt", data, true);
```

---

public static string[] LoadDataFromFile (string path)

-Loads data from a specified path and returns it as a string array. (Path must include filename and extension.)

PARAMETERS:

path - path containing text file to be read and cast to an array of strings.

USEAGE:

```
public string[] data;
        data = Sys.LoadDataFromFile("Path_To_Text_File/SomeFile.txt");
```

---

public static List<string> LoadDataFromFile (string path)

-Loads data from a specified path and returns it as a list of strings. (Path must include filename and extension.)

PARAMETERS:

path - path containing text file to be read and cast to list of strings.

USEAGE:

```
public List<string> data;
        data = Sys.LoadDataFromFile("Path_To_Text_File/SomeFile.txt");
```

Screen Capturing Methods

---

public static void CaptureScreenshot (MonoBehaviour instance)

      -Saves a screenshot to default location: Application.persistentDataPath + "/" + Application.productName+ "/Screenshots/" + System.DateTime.Now.ToString("MMddyyyy - hhmmss") + ".png"

      PARAMETERS:

            instance - the instance calling the CaptureScreenshot method. Use 'this' by default.

      USEAGE:

            Sys.CaptureScreenshot(this);

---

public static void CaptureScreenshot (MonoBehaviour instance, string path,)

      -Saves a screenshot to a specified path. (Path must include filename and '.png' extension.)

      PARAMETERS:

            instance - the instance calling the CaptureScreenshot method. Use 'this' by default.

            path - the path to save screenshot as png file.

      USEAGE:

            Sys.CaptureScreenshot(this, "Path_To_Screenshots/SomeScreenshot.png");

---

public static void CaptureScreenshot (MonoBehaviour instance, string path, bool openDirectory)

      -Saves a screenshot to a specified path then opens the png image upon save. (Path must include filename and '.png' extension.)

      PARAMETERS:

            instance - the instance calling the CaptureScreenshot method. Use 'this' by default.

            path - the path to save screenshot as png file.

            openDirectory - if true, the saved png will open with the systems default image viewer (Desktop Only).

      USEAGE:

            Sys.CaptureScreenshot(this, "Path_To_Screenshots/SomeScreenshot.png", true);

---

public static Texture2D ScreenToTexture2D ()

      -Captures and returns screenshot as Texture2D. (Not Recommended - Bugged)

      USEAGE:

            public Texture2D tex;

```
tex = Sys.ScreenToTexture2D();
```

---

public static Texture2D LoadImageAtPath(string path)

-Loads an image from a specified path and returns it as Texture2D.

PARAMETERS:

path - the path containing the desired png image.

USEAGE:

```
public Texture2D tex;

tex = Sys.LoadImageAtPath("Path_To_Image/SomeImage.png");
```

Basic Arithmetic Methods

---

public static int Add(int a, int b, int c, int d)

     -Returns sum of any given multiple values.

     -Will add anywhere from two to four seperate values.

          PARAMETERS:

               a - value to be added.

               b - value to be added.

               c - value to be added.

               d - value to be added.

          USEAGE:

               public int sum;

                    sum = Sys.Add(1,2,3,4);

---

public static float Add(float a, float b, float c, float d)

     -Returns sum of any given multiple values.

     -Will add anywhere from two to four separate values.

          PARAMETERS:

               a - value to be added.

               b - value to be added.

               c - value to be added.

               d - value to be added.

          USEAGE:

               public float sum;

                    sum = Sys.Add(1.2f,2.2f,3.4f,4.5f);

---

public static int Subtract(int a, int b, int c, int d)

     -Returns sum of any given multiple values.

     -Will subtract anywhere from two to four seperate values.

          PARAMETERS:

               a - value to be subtracted.

               b - value to be subtracted.

               c - value to be subtracted.

               d - value to be subtracted.

          USEAGE:

               public int sum;

                    sum = Sys.Subtract(1,2,3,4);

public static float Subtract(float a, float b, float c, float d)

    -Returns sum of any given multiple values.

    -Will subtract anywhere from two to four separate values.

        PARAMETERS:

            a - value to be subtracted.

            b - value to be subtracted.

            c - value to be subtracted.

            d - value to be subtracted.

        USEAGE:

            `public float sum;`

                `sum = Sys.Subtract(1.2f,2.2f,3.4f,4.5f);`

---

public static int Multiply(int a, int b, int c)

    -Returns sum of any given multiple values.

    -Will multiply anywhere from two to four separate values.

        PARAMETERS:

            a - value to be multiplied.

            b - value to be multiplied.

            c - value to be multiplied.

            d - value to be multiplied.

        USEAGE:

            `public int sum;`

                `sum = Sys.Multiply(1,2,3,4);`

---

public static float Multiply(float a, float b, float c)

    -Returns sum of any given multiple values.

    -Will multiply anywhere from two to four seperate values.

        PARAMETERS:

            a - value to be multiplied.

            b - value to be multiplied.

            c - value to be multiplied.

            d - value to be multiplied.

        USEAGE:

            `public float sum;`

                `sum = Sys.Multiply(1.2f,2.2f,3.4f,4.5f);`

---

public static float Divide(float a, float b)

    -This will return the quotient of the given values.

    -Can return only float values.

        PARAMETERS:

            a - value to be divided.

USEAGE:

```
public int sum;
        sum = Sys.Divide(1.2f,2.2f);
```

Unique Identifier Generation Methods

public static string GenerateUniqueID(int length)

-Returns a randomly generated identification code of any given length.

PARAMETERS:

length - desired length of the returned unique identifier.

USEAGE:

```
public string uniqueID;
        uniqueID = Sys.GenerateUniqueID(21);
```

'SystemInfo' Methods

---

public static void SaveSystemInfo (string path)

-Saves UnityEngine.SystemInfo data to a neatly formatted text file at a specified path. (Path must include filename and extension.)

PARAMETERS:

path - the path to save UnityEngine.SystemInfo as a formatted text file.

USEAGE:

Sys.SaveSystemInfo("Path_To_Save/SystemInfo.txt");

---

public static void SaveSystemInfo (string path, bool openDirectory)

-Saves UnityEngine.SystemInfo data to a neatly formatted text file at a specified path then opens file directory upon save. (Path must include filename and extension.)

PARAMETERS:

path - the path to save UnityEngine.SystemInfo as a formatted text file.

openDirectory - if true, File Explorer will open the containing folder of your saved file (Desktop Only).

USEAGE:

Sys.SaveSystemInfo("Path_To_Save/SystemInfo.txt", true);

---

public static List<string> GetSystemInfo ()

-Returns formatted UnityEngine.SystemInfo to a list of strings.

USEAGE:

public List<string> systemInfo;

systemInfo = Sys.GetSystemInfo();

NOTES:

Always make sure you're importing the 'UnityEngine.Sys' namespace if using javascript.
At the top of your file paste: import UnityEngine.Sys;
When programming in C#, it is not necessary to import the 'UnityEngine.Sys' as c# files in unity must already import the 'UnityEngine' namespace.