AA Save and Load System

DOCUMENTATION

1	What is AA Save and Load System?	**3**
2	How to install?	**3**
3	What is the “Save System” component?	**3**
4	How to change the save file location?	**4**
5	How to change the save file extension?	**4**
6	How to change the encryption key?	**5**
7	How to save the game data?	**5**
8	How to load the game data?	**9**
9	How to delete the game data?	**13**
10	DoesDataExists method.	**14**
11	GetDataType method.	**14**
12	Developer contact.	**14**

1) What is AA Save and Load System?

AA Save and Load System is a Unity Asset developed by Ahmet ALP. With this asset, game developers can save and manage the game data in the player's device.

---

2) How to install?

Step 1 After purchasing the asset, in the Unity Editor window, go to **Window > Package Manager.**

Step 2 In the Package Manager window, on the top left corner, select **Packages : My Assets.**

Step 3 Locate **AA Save and Load System** on the **list.**

Step 4 On the bottom right corner, press **Download** and wait for the Package Manager to download the asset.

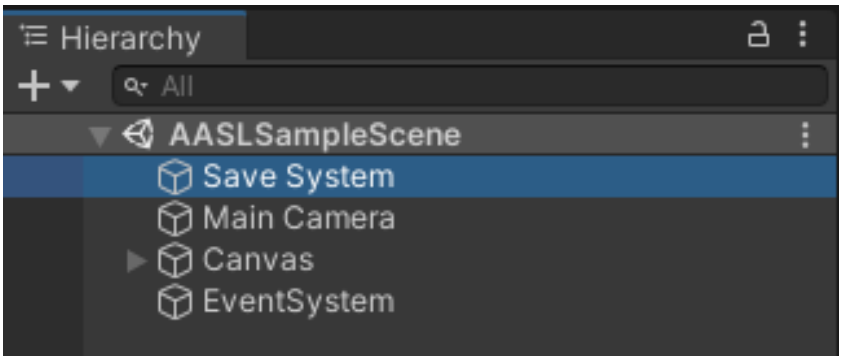
Step 5 Press **Import.**

Step 6 On the **Import Unity Package** window, press **Import** and wait for Unity to install the asset to your current project.

Step 7 Exit the **Package Manager.**

Step 8 Right click on the **Hierarchy** and select **AA Save and Load System > Create Save System.**

This will create a GameObject in your Hierarchy. This GameObject is named "**Save System**" and has two components on it: Transform and **Save System.**



✔ **Installation has been completed.**

---

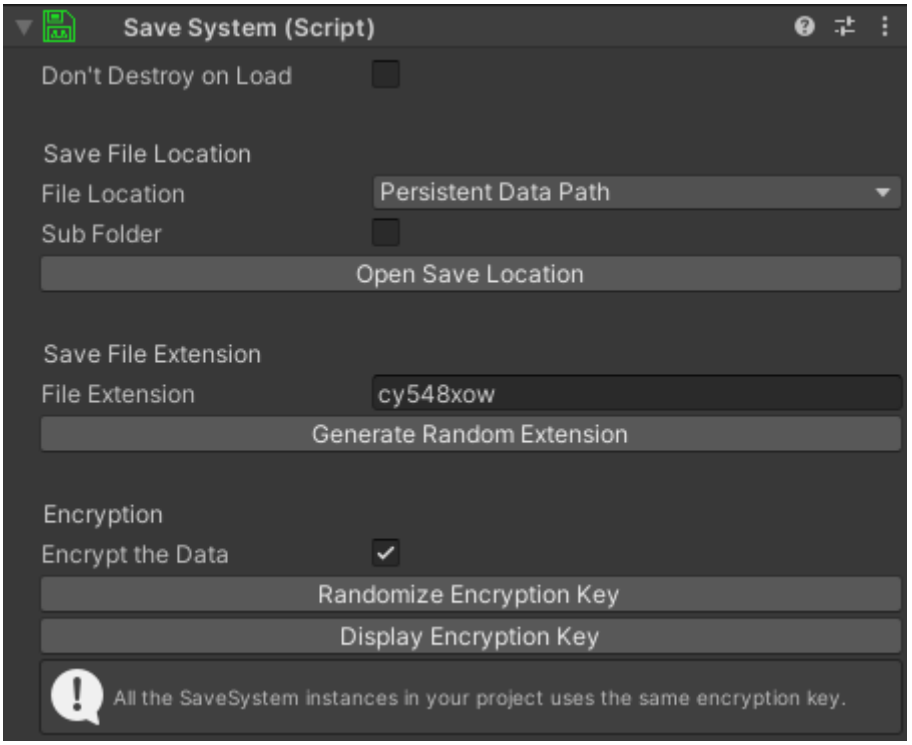
3) What is the "Save System" component?

The Save System component is the **operator** of the entire save system. You will be using this Save System component to manage the game data.

✔ **By following the step 8 in the installation process, you can have a Save System component on every scene in your project. You can even have multiple Save System components in one scene.**

✦ Save files created by this asset are not included within the build. In other words, the save files created during the production time will not be passed on to the player's device.

A view of the Save System component is given below. In the Unity Editor, you can hover your cursor on the fields to get the descriptions of them.



✦ All the properties on the Save System component are also accessible with the Get and Set methods.

---

4) How to change the save file location?

In the Save System inspector, there is a group of fields located under the **Save File Location** title. In this section, locate the **File Location** dropdown to change the save file location. You can also press the **Open Save Location** button to view the saving location. All the save file locations are listed below.

FILE LOCATION	DESCRIPTION
Persistent Data Path **(Recommended)**	Path to the persistent data directory. The data stored in here will be kept in between the runs. Files in this location are not deleted by the updates.
Application Data Path	Path to the game data folder on the target device.
Temporary Cache Path	Path to the temporary file directory.
Streaming Assets Path	Path to the StreamingAssets folder. This folder is used to store assets.
Custom File Path	A custom file path that can be created by you.

---

5) How to change the save file extension?

In the Save System inspector, there is a field named **File Extension.** You can use this field to customize your save file extension. Don't use spaces or special characters. Also, you can use the **Generate Random Extension** button to create a random save file extension.

6) How to change the encryption key?

This asset uses an *(optional)* encryption algorithm to encrypt your game data before saving it to the player's device. You can change the encryption key by pressing the **Randomize Encryption Key** button in the Save System inspector. You can also use the **Display Encryption Key** button to view the key in your code editor.

---

7) How to save the game data?

Save System component contains **Save** methods. You can use these methods to save the game data in the game mode. You can call these methods from anywhere in your project. In order to access the **Save** methods, first you need to access the **Save System** component. Examples for saving the game data for some of the popular data types are listed below.

DATA TYPE	CODE EXAMPLE IN C#

Boolean	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  bool exampleBool = true;`
	
	`  // Saves the game data with name and value.`
	`  // Each game data must have a unique name.`
	`  saveSystem.Save("Data Name", exampleBool);`
	`}`

Boolean Array	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  bool[] exampleBoolArray = { true, false, true };`
	
	`  saveSystem.Save("Data Name", exampleBoolArray);`
	`}`

. . .	. . .

Color	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  Color exampleColor = new Color(0.5f, 0.25f, 0.75f, 1.0f);`
	
	`  saveSystem.Save("Data Name", exampleColor);`
	`}`

Color Array	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  Color[] exampleColorArray = { Color.white, Color.blue };`
	
	`  saveSystem.Save("Data Name", exampleColorArray);`
	`}`

DateTime	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  DateTime exampleDateTime = DateTime.Now;`
	
	`  saveSystem.Save("Data Name", exampleDateTime);`
	`}`

DateTime Array	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  DateTime[] exampleDateTimeArray = { DateTime.Now, DateTime.Min };`
	
	`  saveSystem.Save("Data Name", exampleDateTimeArray);`
	`}`

. . .	. . .

Double	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  double exampleDouble = 5.25d;`
	
	`  saveSystem.Save("Data Name", exampleDouble);`
	`}`

Double Array	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  double[] exampleDoubleArray = { 5.25d, 6.80d, 8.47d };`
	
	`  saveSystem.Save("Data Name", exampleDoubleArray);`
	`}`

Float

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	float exampleFloat = 13.07F;

	saveSystem.Save("Data Name", exampleFloat);
}
```

Float Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	float[] exampleFloatArray = { 13.07f, -20.25f };

	saveSystem.Save("Data Name", exampleFloatArray);
}
```

Integer

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	int exampleInt = 25;

	saveSystem.Save("Data Name", exampleInt);
}
```

Integer Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	int[] exampleIntArray = { 25, -50, 75 };

	saveSystem.Save("Data Name", exampleIntArray);
}
```

. . .

. . .

String

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	string exampleString = “myString”;

	saveSystem.Save("Data Name", exampleString);
}
```

String Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	string[] exampleStringArray = { “my”, “string”, “array” };

	saveSystem.Save("Data Name", exampleStringArray);
}
```

.
.
.

.
.
.

Vector 3

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	Vector3 exampleVector3 = new Vector3(30.0f, -40.0f, 50.0f);

	saveSystem.Save("Data Name", exampleVector3);
}
```

Vector 3 Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	Vector3[] exampleVector3Array =
	{
		new Vector3(-15.85f, 26.71f, 10.77f),
		new Vector3(0.37f, -9.18f, -8.49f),
		new Vector3(-5.5f, 24,62f, -20.25f)
	};

	saveSystem.Save("Data Name", exampleVector3Array);
}
```

8) How to load the game data?

DATA TYPE	CODE EXAMPLE IN C#

Boolean	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  bool exampleBool = saveSystem.Load("Data Name").AsBool();`
	
	`  // or`
	
	`  bool exampleBool2 = saveSystem.Load("Data Name", true).AsBool();`
	`  // The second parameter is a back-up value.`
	`  // If the save file for the given game data doesn't exists,`
	`  // the back-up value will be saved automatically and returned`
	`  // as the result of the method.`
	`}`

Boolean Array	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  bool[] exampleArray = saveSystem.LoadArray("Data Name").AsBoolArray();`
	
	`  // or`
	
	`  bool[] backupArray = { true, false, true };`
	`  bool[] exampleArray2 =`
	`  saveSystem.LoadArray("Data Name", backupArray).AsBoolArray();`
	`}`

. . .	. . .

Color	`using AASave;`
	
	`// Save System component on the Save System GameObject.`
	`public SaveSystem saveSystem;`
	
	`private void Start()`
	`{`
	`  Color exampleColor = saveSystem.Load("Data Name").AsColor();`
	
	`  // or`
	
	`  Color exampleColor2 = saveSystem.Load("Data Name", Color.red).AsColor();`
	`}`

Color Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	Color[] exampleArray = saveSystem.LoadArray("Data Name").AsColorArray();

	// or

	Color[] backupArray = { Color.red, Color.green, Color.blue };
	Color[] exampleArray2 =
		saveSystem.LoadArray("Data Name", backupArray).AsColorArray();
}
```

DateTime

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	DateTime exampleDateTime = saveSystem.Load("Data Name").AsDateTime();

	// or

	DateTime exampleDateTime2 =
		saveSystem.Load("Data Name", DateTime.Now).AsDateTime();
}
```

DateTime Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	DateTime[] exampleArray =
		saveSystem.LoadArray("Data Name").AsDateTimeArray();


	// or

	DateTime[] backupArray = { DateTime.Now, DateTime.MinValue };
	DateTime[] exampleArray2 =
		saveSystem.LoadArray("Data Name", backupArray).AsDateTimeArray();
}
```

. . .

. . .

Double

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	double exampleDouble = saveSystem.Load("Data Name").AsDouble();

	// or

	double exampleDouble2 = saveSystem.Load("Data Name", 23.56d).AsDouble();
}
```

Double Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	double[] exampleArray = saveSystem.LoadArray("Data Name").AsDoubleArray();

	// or

	double[] backupArray = { 75.684d, -17.24d, 36.88d };
	double[] exampleArray2 =
		saveSystem.LoadArray("Data Name", backupArray).AsDoubleArray();
}
```

Float

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	float exampleFloat = saveSystem.Load("Data Name").AsFloat();

	// or

	float exampleFloat2 = saveSystem.Load("Data Name", 36.42f).AsFloat();
}
```

Float Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	float[] exampleArray = saveSystem.LoadArray("Data Name").AsFloatArray();

	// or

	float[] backupArray = { 1.42f, 3.785f, -4.89f };
	float[] exampleArray2 =
		saveSystem.LoadArray("Data Name", backupArray).AsFloatArray()
}
```

Integer

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	int exampleInt = saveSystem.Load("Data Name").AsInt();

	// or

	int exampleInt2 = saveSystem.Load("Data Name", 23).AsInt();
}
```

Integer Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	int[] exampleArray = saveSystem.LoadArray("Data Name").AsIntArray();

	// or

	int[] backupArray = { 4, -54, 789, 1024 };
	int[] exampleArray2 =
		saveSystem.LoadArray("Data Name", backupArray).AsIntArray();
}
```

. . .

. . .

String

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	string exampleString = saveSystem.Load("Data Name").AsString();

	// or

	string exampleString2 = saveSystem.Load("Data Name", “here”).AsString();
}
```

String Array

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
	string[] exampleArray = saveSystem.LoadArray("Data Name").AsStringArray();

	// or

	string[] backupArray = { “my”, “backup”, “array” };
	string[] exampleArray2 =
		saveSystem.LoadArray("Data Name", backupArray).AsStringArray();
}
```

. . .

. . .

Vector 3	
	using AASave;
	
	// Save System component on the Save System GameObject.
	public SaveSystem saveSystem;
	
	private void Start()
	{
	Vector3 exampleVector3 = saveSystem.Load("Data Name").AsVector3();
	
	// or
	
	Vector3 exampleVector3_2 =
	saveSystem.Load("Data Name", Vector3.one).AsVector3();
	}

Vector 3 Array	
	using AASave;
	
	// Save System component on the Save System GameObject.
	public SaveSystem saveSystem;
	
	private void Start()
	{
	Vector3[] exampleArray = saveSystem.LoadArray("Data Name").AsVector3Array();
	
	// or
	
	Vector3[] backupArray = { Vector3.one, Vector3.zero };
	Vector3[] exampleArray2 =
	saveSystem.LoadArray("Data Name", backupArray).AsVector3Array();
	}

---

9) How to delete the game data?

In order to delete the game data, use the **Delete** method in the **Save System** component. This method **permanently** deletes the game data and returns **true** if the process was successful.

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
      saveSystem.Delete("Data Name");
}
```

✦ Deleting a game data is a permanent action and cannot be undone.

✦ When a game data is deleted, the save file **IS NOT** moved into the trash can on your operating system.

10) DoesDataExists method.

This method goes through all the **save files** in the save location and finds out if the game data with the given name exists or not. Returns **true** if the corresponding save file for the given data exists. Otherwise, it returns **false**.

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
      bool exists = saveSystem.DoesDataExists("Data Name");
}
```

---

11) GetDataType method.

This method finds the **data type** of the given game data. It returns the data type as an **enum**. Example on the GetDataType method:

```
using AASave;

// Save System component on the Save System GameObject.
public SaveSystem saveSystem;

private void Start()
{
      DataTypes myDataType = saveSystem.GetDataType("Data Name");
}
```

---

12) Developer contact.

AHMET ALP
Asset Store Profile
Email