Slushy's Player Prefs

An easy to use, scalable and flexible alternative to Unity's default Player Prefs. Slushy's Player Prefs are available for download on this Github's Releases (https://github.com/SlushyRH/Slushys-Player-Prefs/releases) page or on the Asset Store (). Also comes with a Player Pref viewer where you can view all your saved Player Prefs even if they are normal PlayerPrefs!

With Slushy's Player Prefs you can save and load virtually any data with two very simple methods!

Disclaimer

As of now, this tool only works on Windows machines. I don't own a Mac or Linux device so therefore can't port it, any help would be appreciated! OSX and Linux Methods (https://github.com/SlushyRH/SPP/blob/master/Slushy%27s%20Player%20Prefs/Editor/SPPEditorUtility.cs#L86)

Features

	Player Prefs	Slushy's Player Prefs
Encryption	☐	✔
Set and Get Ints	✔	✔
Set and Get Strings	✔	✔
Set and Get Floats	✔	✔
Set and Get Booleans	☐	✔
Set and Get Chars	☐	✔
Set and Get Doubles	☐	✔
Set and Get Enums	☐	✔
Set and Get Vector2	☐	✔
Set and Get Vector3	☐	✔
Set and Get Vector4	☐	✔
Set and Get Quaternion	☐	✔
Set and Get Colors	☐	✔
Set and Get Lists	☐	✔
Set and Get Arrays	☐	✔

How to Use

More information in the demo script included in the 'SRH\Slushy's Player Prefs\Demo\SPP_Demo.cs'. If you need to convert any of your existing Player Prefs to SPP, just click the Convert button in the SRH tab in the toolbar.

Save Values to Player Prefs

```
using SRH;


public class HighscoreManager : MonoBehaviour
{
    public void Awake()
    {
        // Not encrypted
        SPP.Set("myInt", 126);

        SPP.Set("myBool", true);

        SPP.Set("myVector3", transform.position);


        // Encrypted
        SPP.SetEncrypted("myInt", 126);

        SPP.SetEncrypted("myBool", true);

        SPP.SetEncrypted("myVector3", transform.position);
    }
}
```

Get Values from Player Prefs

```
using SRH;


public class HighscoreManager : MonoBehaviour
{
    public void Awake()
    {
        // Works even if pref is encrypted
        int myInt = SPP.Get<int>("myInt");

        bool myBool = SPP.Get<bool>("myBool");

        Vector3 myVector3 = SPP.Get<Vector3>("myVector3");
    }
}
```