Chapter 3 – PlayerPrefs

VRA 705
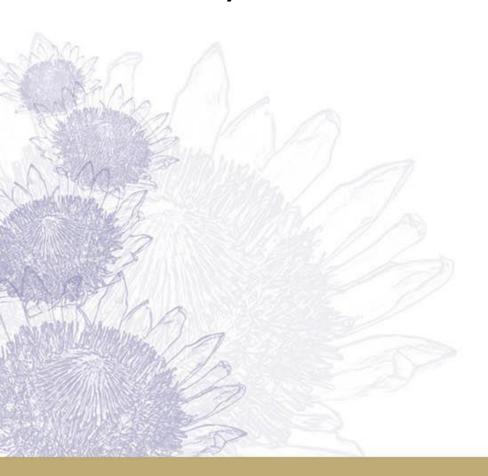
By Kessel Okinga and Mfundo Andrew
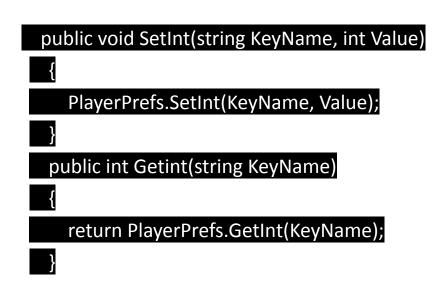
Outline

❖ PlayerPrefs

PlayerPrefs

❖ The PlayerPrefs is a class in Unity used to save small amounts of data locally.

❖ PlayerPrefs can store string, float and integer values into the user's platform registry.

❖ Unity stores the PlayerPrefs data in a local registry, without encryption. Furthermore, the specific storage mechanism employed by PlayerPrefs varies across different operating systems on which the application is executed.

Static Methods (Set)

❖ ***Set*** refers to assigning a value to a new or an existing variable. You require to provide two parameters such as a string key and integer, Float, or String value.

❖ Method of Set with PlayerPrefs:
SetInt, SetFloat , and SetString.

```
public void SetInt(string KeyName, int Value)
{
    PlayerPrefs.SetInt(KeyName, Value);
}
public int Getint(string KeyName)
{
    return PlayerPrefs.GetInt(KeyName);
}
```

Static Methods (HasKey)

- ❖ **HasKey** is a method within PlayPrefs that verify if the key exist.

- ❖ It returns true if the given key exists in PlayerPrefs, otherwise returns false.

```csharp
public void HasKey(string KeyName)
{
    if (PlayerPrefs.HasKey(KeyName))
    {
        Debug.Log("The key " + KeyName + " exists");
    }
    else
        Debug.Log("The key " + KeyName + " does not exist");
}
```

Static Methods (Get)

❖ **_Get_** is method that return the value assigned or set using the Set methods. It returns the value corresponding to key in the preference file if it exists.

❖ Method of Get with PlayerPrefs:

GetInt, GetFloat , and GetString.

```
public void HasKey(string KeyName)
    {
        if (PlayerPrefs.HasKey(KeyName))
        {
            Debug.Log("The key " + KeyName + " exists");
        }
        else
            Debug.Log("The key " + KeyName + " does not exist");
    }
```

Static Methods (DeleteKey)

❖ ***DeleteKey*** is a method used to remove a specific key and its associated value from a Set data structure. If the key does not exist in the Set, calling ***DeleteKey*** has no effect on the data.

❖ Set data structure does not allow duplicate keys, so there can be only one value associated with a specific key.
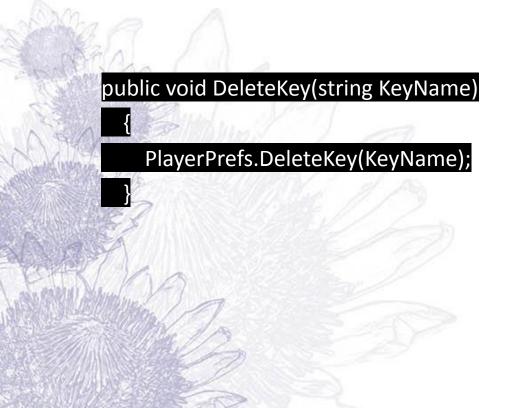
```
public void DeleteKey(string KeyName)
    {
        PlayerPrefs.DeleteKey(KeyName);
    }
```

Static Methods (DeleteAll)

❖ **DeleteAll** is a method used to remove or delete all keys and their associated values from the preferences.

```
public void DeleteKey(string KeyName)
{
    PlayerPrefs.DeleteKey(KeyName);
}
```

Key points

❖ PlayerPrefs is a method used to locally store values, with 'Set' allowing for assigning values, 'Get' enabling retrieval of values, 'DeleteKey' removing a specific key and its associated value, and 'DeleteAll' removing all data stored in the preferences.