Q INVENTORY SYSTEM

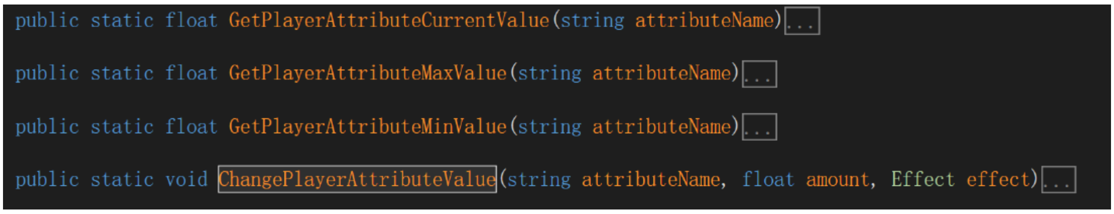
2.0.0

USER API

First of all, you need a header : `using QInventory;`

(1)Get And Change Player’s Attribute

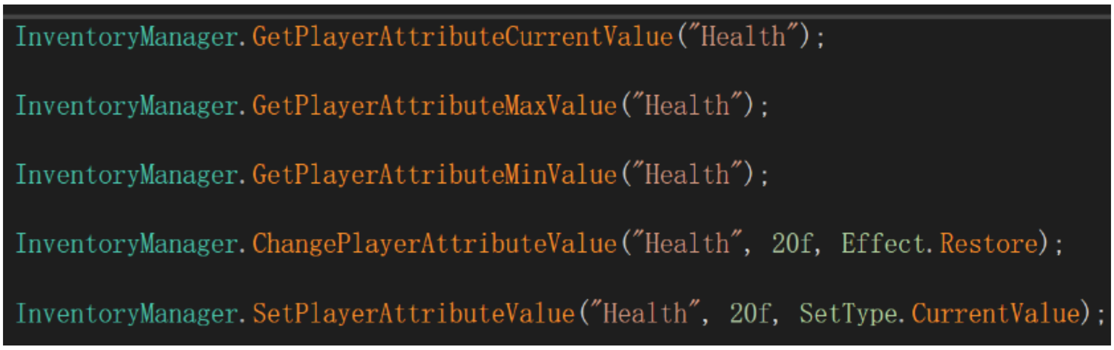
In “InventoryManager” script



```
public static float GetPlayerAttributeCurrentValue(string attributeName)...

public static float GetPlayerAttributeMaxValue(string attributeName)...

public static float GetPlayerAttributeMinValue(string attributeName)...

public static void ChangePlayerAttributeValue(string attributeName, float amount, Effect effect)...
```

Use these functions in this way. (Health, for example)

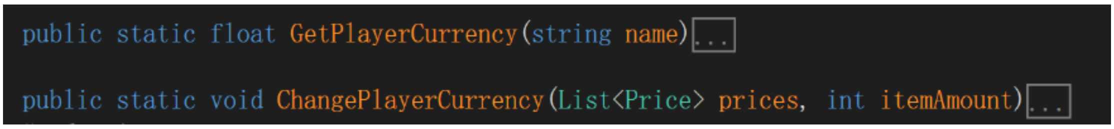
Call the functions you want to use



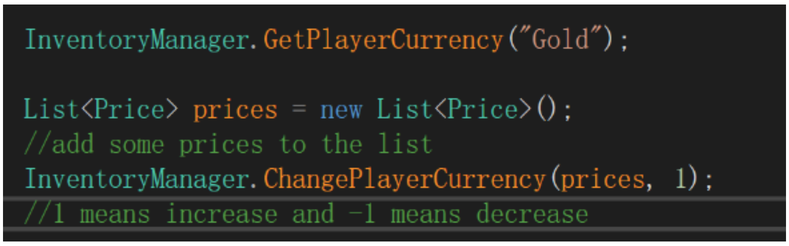
```
InventoryManager.GetPlayerAttributeCurrentValue("Health");

InventoryManager.GetPlayerAttributeMaxValue("Health");

InventoryManager.GetPlayerAttributeMinValue("Health");

InventoryManager.ChangePlayerAttributeValue("Health", 20f, Effect.Restore);

InventoryManager.SetPlayerAttributeValue("Health", 20f, SetType.CurrentValue);
```

(2) Change Player’s Currency

In “InventoryManager” script



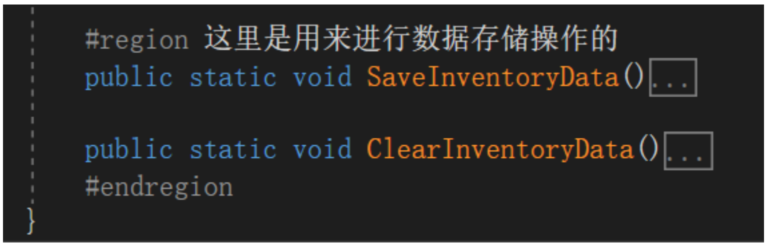
```
public static float GetPlayerCurrency(string name)...

public static void ChangePlayerCurrency(List<Price> prices, int itemAmount)...
```



```
InventoryManager.GetPlayerCurrency("Gold");

List<Price> prices = new List<Price>();
//add some prices to the list
InventoryManager.ChangePlayerCurrency(prices, 1);
//1 means increase and -1 means decrease
```

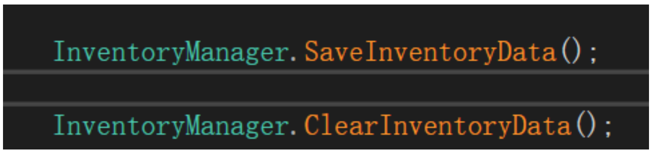
(3) Save and Clear Player’s Inventory Data

In “InventoryManager” script



```
    #region 这里是用来进行数据存储操作的
    public static void SaveInventoryData()...

    public static void ClearInventoryData()...
    #endregion
}
```

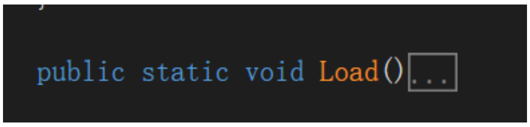
How to use them, for example



```
InventoryManager.SaveInventoryData();

InventoryManager.ClearInventoryData();
```

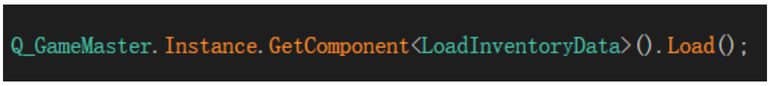
(4) Load Player’s Inventory Data

In “LoadInventoryData” script



```
public static void Load()...
```

How to use it, for example



```
Q_GameMaster.Instance.GetComponent<LoadInventoryData>().Load();
```