

〈Unity Inventory Frameworks〉

(Inventory Pro & Simple Inventory System)

1. 개요

- 게임에서 아이템 관리(Inventory System) 은 RPG, 샌드박스, 서바이벌 장르에서 필수 기능
- Unity는 기본 인벤토리 제공이 없기 때문에, 자주 쓰이는 Asset Store 패키지를 사용
- 대표적으로:
 - Inventory Pro (과거 유명, 현재는 단종 상태) 레거시 프로젝트에서 발견 가능)
 - Simple Inventory System (가볍고 단순한 무료/저가형 패키지, 혹은 커스텀 제작 기반 예제 많음)

2. Inventory Pro

2.1 특징

- 고급 인벤토리 시스템 (아이템, 장비, 스킬, 스탯 관리까지 지원)
- 드래그 앤 드롭 UI, 아이템 스택, 퀘스트/상점/제작 시스템까지 내장
- 모듈식 구조 > 확장 가능
- 현재는 공식 지원 중단되었으므로 새 프로젝트에서는 대체 솔루션 권장

2.2 설치

- Unity Asset Store (현재는 다운로드 불가, 보유자만 사용 가능)([Git으로 위치 바뀜](#))
- Import 후 InventoryManager Prefab을 Scene에 배치

2.3 주요 컴포넌트

컴포넌트	설명
InventoryManager	인벤토리 전체 관리
Item Database	아이템 정보(이름, 아이콘, 타입, 스탯 등) 저장
Inventory UI	인벤토리 창 UI Prefab
Equipment	장비 장착 관리
Loot Table	드롭 아이템 정의

2.4 기본 예제

2.4.1 Scene에 InventoryManager Prefab 배치

2.4.2 Item Database에서 Potion, Sword 아이템 생성

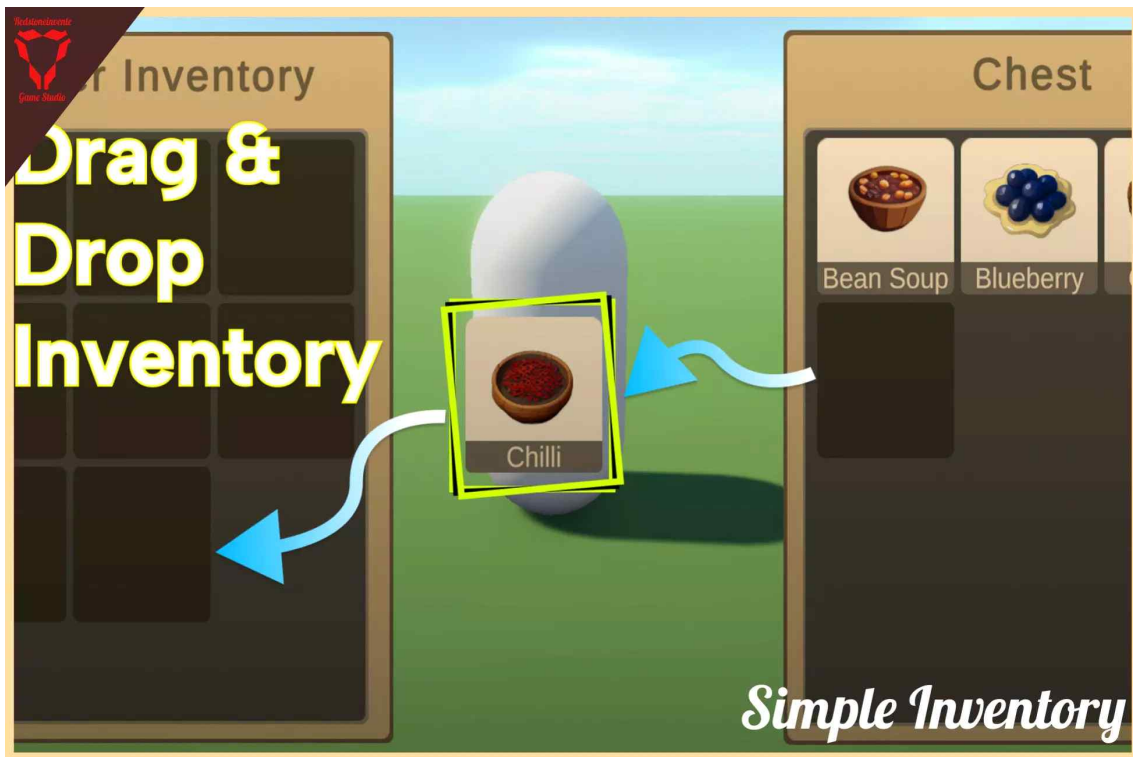
2.4.3 Player에 Inventory UI 연결

2.4.4 실행 후 'I' 키> 인벤토리 열림, 아이템 확인

3. Simple Inventory System

3.1 특징

- 가벼운 구조, 커스터마이징 용이
- 보통 Script 기반 또는 무료 Asset 형태
- 단순한 아이템 획득/사용/제거 기능 제공
- UI/저장 기능은 직접 구현 필요



3.2 기본 구조

스크립트	역할
Item	아이템 데이터(이름, 설명, 아이콘, 타입)
Inventory	아이템 리스트 관리(획득, 제거, 검색)
Inventory UI	Unity UI/TextMeshPro로 아이템 표시

3.3 스크립트 예제

```
[System.Serializable]
public class Item {
    public string itemName;
    public Sprite icon;
    public int quantity;
}

public class Inventory : MonoBehaviour {
    public List<Item> items = new List<Item>();

    public void AddItem(Item newItem) {
```

```

        items.Add(newItem);
        Debug.Log("획득: " + newItem.itemName);
    }

    public void RemoveItem(Item item) {
        items.Remove(item);
        Debug.Log("삭제: " + item.itemName);
    }
}

```

3.4 UI 연동 예제

```

using UnityEngine;
using UnityEngine.UI;

public class InventoryUI : MonoBehaviour {
    public Inventory inventory;
    public Transform itemParent;
    public GameObject itemSlotPrefab;

    void RefreshUI() {
        foreach (Transform child in itemParent) Destroy(child.gameObject);
        foreach (var item in inventory.items) {
            GameObject slot = Instantiate(itemSlotPrefab, itemParent);
            slot.GetComponentInChildren<Text>().text = item.itemName;
        }
    }
}

```

4. Inventory Pro vs Simple Inventory System 비교

항목	Inventory Pro	Simple Inventory System
가격	유료(현재 단종)	무료/저가 또는 직접 구현
기능 범위	인벤토리+장비+상점+제작+퀘스트	기본 인벤토리/아이템 관리
접근성	Prefab/DB 기반, 고급 기능 내장	스크립트 기반, 직접 확장 필요
UI	기본 UI Prefab 제공	Unity UI/TMP 직접 제작
프로젝트 규모	중~대형 RPG/서바이벌	인디/프로토타입/학습용

5. 활용 팁

- Inventory Pro

- 레거시 프로젝트 유지보수, 복잡한 RPG 구조 필요 시 여전히 유용

- 단, 공식 지원이 끊겨 업데이트/호환성 주의 필요

- Simple Inventory System:

- 직접 커스터마이징 원하는 개발자에게 적합

- 작은 인디 게임, 프로토타입, 학습용 프로젝트에서 권장

- TextMeshPro UI와 쉽게 연결 가능