SkillTree System Document

1. 소개

개요

이 문서는 유니티에서 스킬 트리 시스템을 구축하고 관리할 수 있게 해주는 패키지에 대해 설명합니다. 본 패키지는 커스텀 그래프뷰와 스크립테이블 오브젝트를 활용하여 에디터 상에서 직관적으로 스킬 트리를 구축하고, UGUI 기반의 스킬 트리 UI를 자동으로 생성합니다.

특징

- 에디터 통합: 스킬 트리의 시각적 구축과 관리를 위한 커스텀 에디터 지원.
- 자동 UI 생성: 스킬 트리 데이터에 기반한 스킬 트리 UI 생성 지원.
- 커스터마이징 가능: 개발자의 요구에 맞춰 스킬 및 스킬 트리 요소를 쉽게 커스터마이징할 수 있음.

2. 시작하기

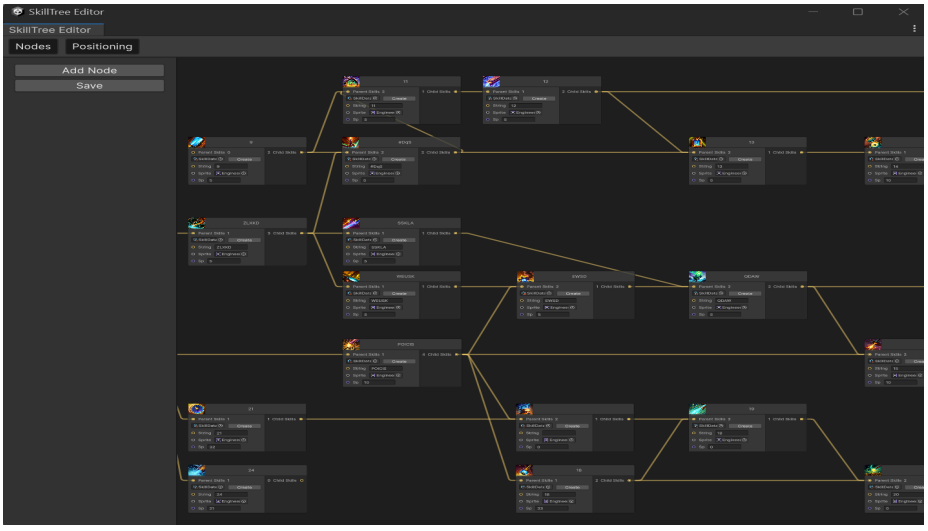
첫 스킬 트리 생성

스킬 트리 구축: 에디터에서 Create > Skill Tree Data를 선택하여 새 스킬 트리를 생성합니다.
스킬 트리 에디터: 생성된 Skill Tree Data 인스펙터 하단의 “Open Graph Editor” 버튼을 클릭합니다.
노드 추가: 스킬 트리 에디터 상에서 좌측의 “Add Node” 메뉴를 사용해 스킬 노드를 추가합니다.
스킬 설정: 각 스킬 노드에 Skill Data 필드에 필요한 스킬을 할당 또는 “Create” 버튼으로 생성합니다.

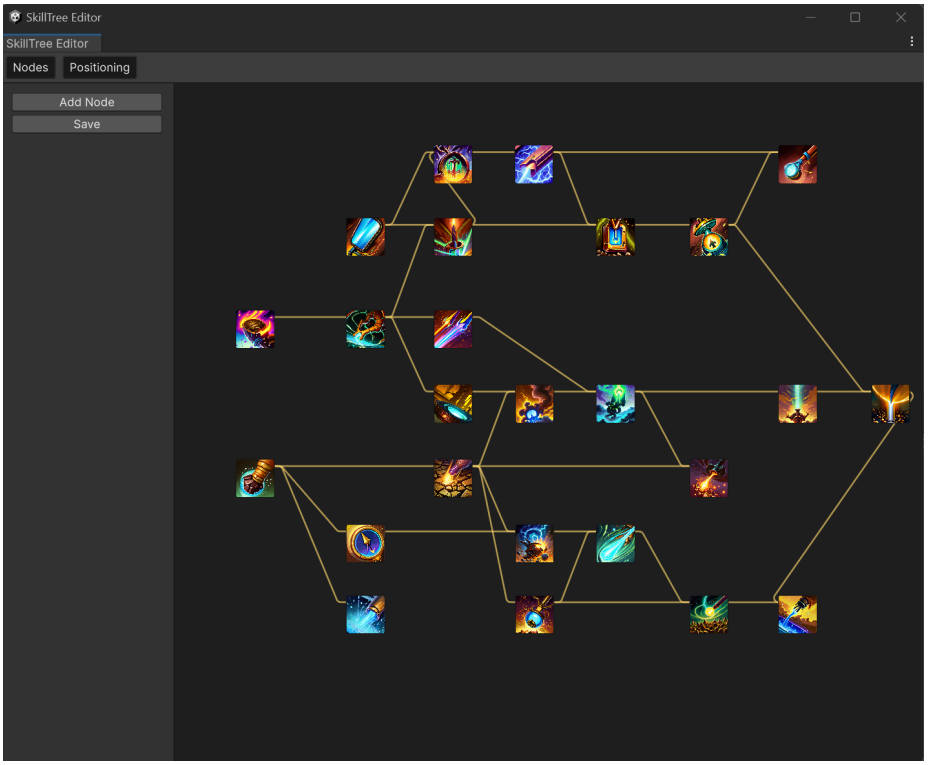
3. 사용법

스킬 트리 구축 및 편집

Node Tab: 스킬 트리의 노드를 구축합니다.



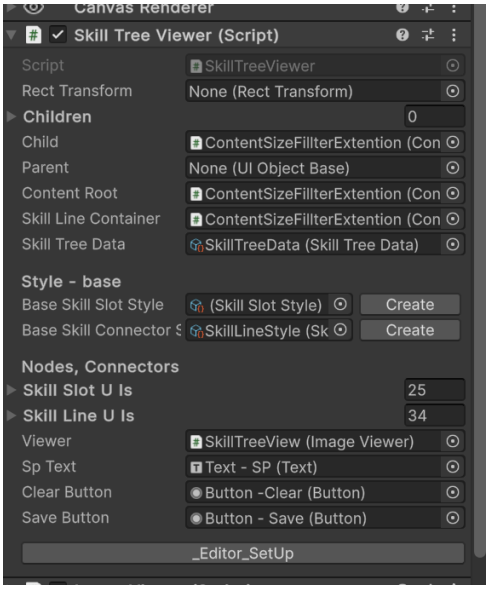
Positioning Tab: 각 스킬의 UI 상에서의 표시 위치를 구축합니다.



스킬 트리 뷰어 사용법

Canvas 를 생성합니다.
Canvas 하위에 SkillTreeView 를 추가한 오브젝트를 생성하고 원하는 크기로 RectTransform을 설정합니다.

SkillTreeView 컴포넌트



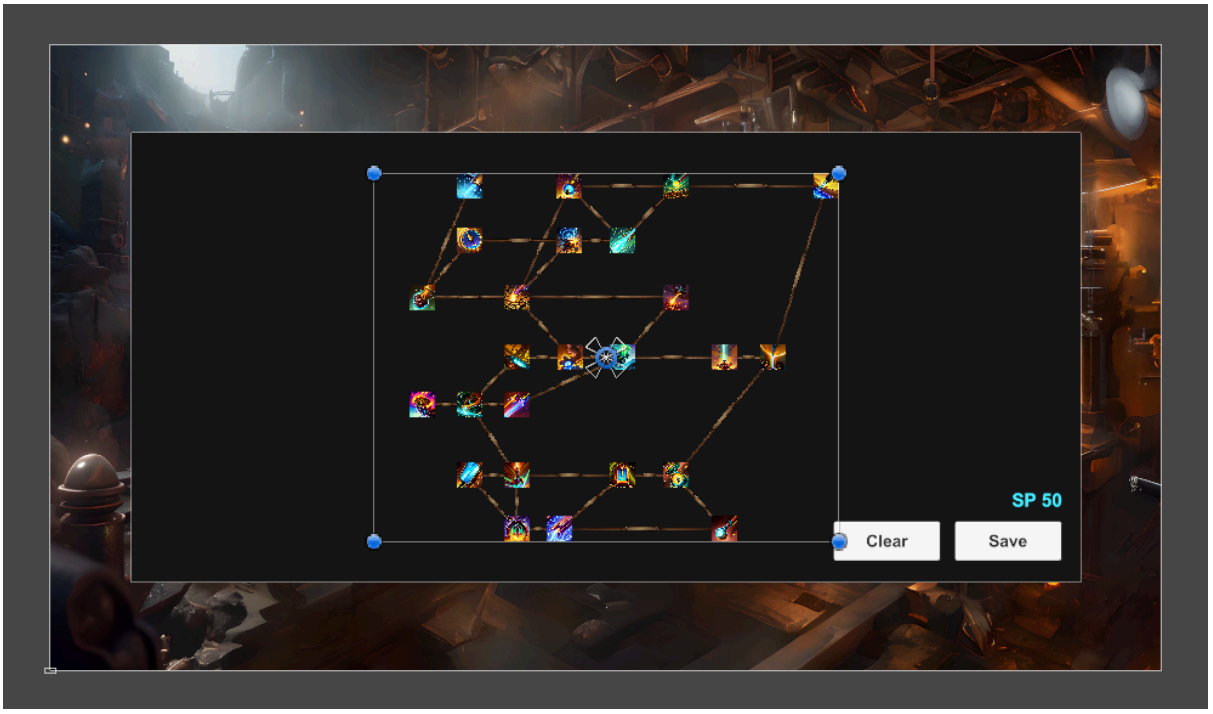
SkillTreeView 컴포넌트의 필수 할당 필드

- Skill Tree Data: 스킬 트리 데이터

SkillTreeView 컴포넌트의 옵션 필드

- Base Skill Connect Style: 스킬의 연결 UI 스타일 정의 데이터
- Base Skill Slot Style: 스킬 슬롯 UI 스타일 정의 데이터

SkillTreeView 컴포넌트 하단의 “_Editor_SetUp” 버튼을 클릭하여 UI를 자동으로 구축합니다.



스킬 트리 데이터 관리

—- (json 문자열로 제공 및 파싱)

4. API 참조

주요 클래스 및 메서드

ContentSizeFillterExtention 클래스

ContentSizeFillterExtention 클래스는 Unity UI 요소의 크기를 자식 요소의 크기와 위치에 기반하여 동적으로 조정하는 커스텀 UI 구성 요소입니다. 이 클래스는 자식 요소들의 경계를 계산하여, 부모 RectTransform의 위치와 크기를 적절하게 업데이트합니다.

속성

Padding padding: 컨텐츠 주변에 적용될 패딩 값입니다. 컨텐츠의 크기 조정 시 이 패딩을 고려하여 크기와 위치가 결정됩니다.

메서드

public static ContentSizeFillterExtention _(): ContentSizeFillterExtention 인스턴스를 생성하고 초기화하는 정적 메서드입니다. 새 GameObject를 생성하고, RectTransform 컴포넌트와 ContentSizeFillterExtention 컴포넌트를 추가한 후, UI를 초기화합니다.

public override void InitializeUI(): UI 컴포넌트의 초기 설정을 수행합니다. 이 메서드는 컴포넌트가 생성될 때 자동으로 호출됩니다.

public void UpdateContentSizeRect(): 부모 RectTransform의 크기와 위치를 자식 요소들의 경계에 맞게 조정합니다. 자식 요소들의 최소 및 최대 위치를 계산하여 부모의 새로운 위치와 크기를 결정하고 적용합니다.

public static void SetFitChildRectSize(RectTransform origin, RectTransform child): 지정된 origin RectTransform을 child의 크기와 위치에 맞게 조정합니다. origin의 위치, 회전, 크기를 child에 맞춰 설정한 후, child를 origin의 자식으로 다시 설정합니다.

ImageViewer 클래스

ImageViewer 클래스는 Unity UI를 사용하여 이미지 컨텐츠의 뷰어 기능을 제공하는 커스텀 UI 구성 요소입니다. 이 클래스는 드래그하여 이미지를 이동시키고, 마우스 스크롤을 통해 이미지를 확대/축소할 수 있는 기능을 포함합니다. 또한, 이미지 컨텐츠의 크기에 맞게 자동으로 크기를 조정하는 기능을 제공합니다.

주요 기능

드래그 이동: 사용자가 이미지를 드래그하여 뷰어 내에서 이동시킬 수 있습니다.
줌 인/아웃: 마우스 스크롤을 통해 이미지를 확대하거나 축소할 수 있습니다.
크기 자동 조정: ContentSizeFillterExtention을 사용하여 이미지 컨텐츠의 크기에 맞게 뷰어의 크기를 자동으로 조정합니다.
마스크 가능 컨텐츠: Image 컴포넌트의 마스킹 기능을 활용하여 뷰어 내에서만 컨텐츠가 보이도록 할 수 있습니다.

속성

public ContentSizeFillterExtention contentSizeFillter: 이미지 컨텐츠의 크기 조정을 담당하는 ContentSizeFillterExtention 인스턴스.
public Image backgroundImage: 뷰어의 배경 이미지.
public bool fixedRoot: 루트 위치 고정 여부.
public bool zoom: 줌 기능 활성화 여부.
public float zoomSpeed = 0.1f: 줌 속도.
public Vector2 zoomLimit = new Vector2(0.5f, 2.0f): 줌 한계값.

메서드

public void SetupZoom(float low, float max): 줌 한계값을 설정합니다.
private void ZoomUpdate(): 줌 기능을 업데이트하는 내부 메서드입니다.
public void OnBeginDrag(PointerEventData eventData): 드래그 시작 이벤트 핸들러.
public void OnDrag(PointerEventData eventData): 드래그 중 이벤트 핸들러.
public void OnEndDrag(PointerEventData eventData): 드래그 종료 이벤트 핸들러.
public void OnPointerDown(PointerEventData eventData): 포인터 다운 이벤트 핸들러.

Image Bezior

- 곡선 이미지 생성 (포인트 베지어 형식)
- 스프라이트 적용 가능