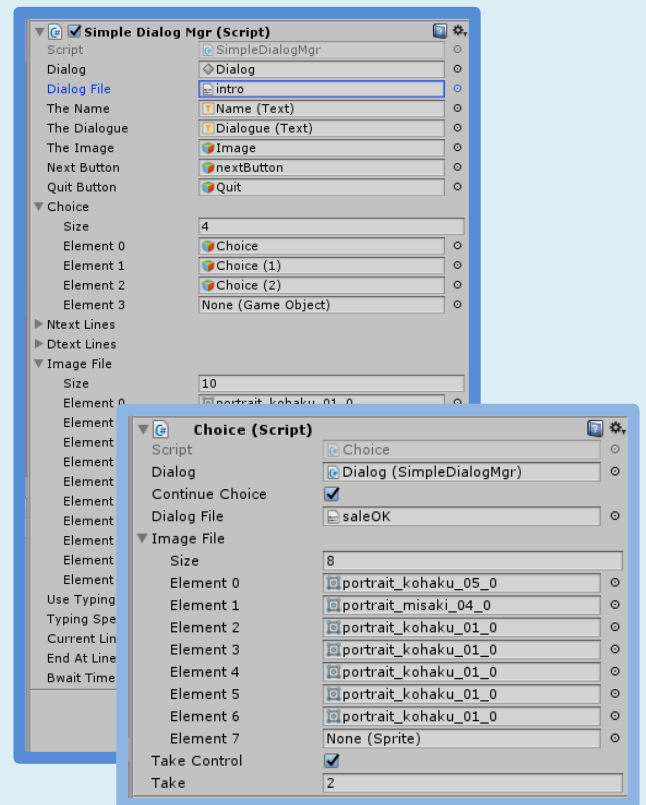


# Easy&Simple Dialog System

## 목차

1. 구성품 - - - - - 2
2. 대화창을 만들어보자 - - - - - 3~13
3. 설정창을 컨트롤해보자 - - - - - 14~16
4. 스크립트를 커스터마이징해보자 - - - 17



본 에셋을 구매해주셔서 감사합니다

Easy&Simple Dialog System은 한두 개의 스크립트로 대화창을 만들 수 있는

매우 간단한 스크립트입니다

겉내지 마시고 가이드북만 잘 따라오시면 됩니다 파이팅!!!^^

# 1. 구성품

## Easy&Simple Dialog System

- > SimpleDialogMgr.cs //이 에셋의 주인공
- > Choice.cs //선택문용 기능 확장 스크립트
- > Guide Book //사용 설명서
- > License //라이선스 문서 꼭 읽어보세요!!!
- > Example //예제 파일

User Guide.unity //예제 씬 파일

Anim //애니메이션 폴더

Script //스크립트 폴더(이런 식으로 응용 할 수 있다 정도로만 보세요)

Sprite//스프라이트 파일

Text //대화창 파일 폴더

## 2 .대화창을 만들어 보자

이 스크립트로 만들 수 있는 대화창은

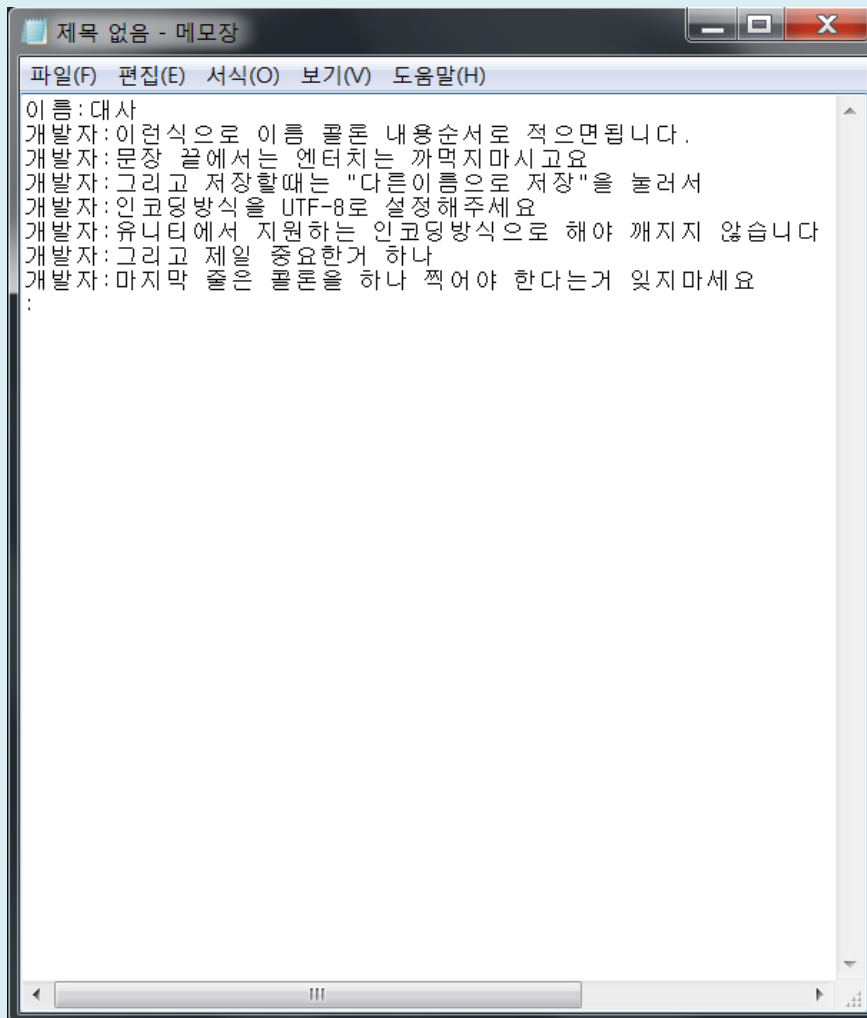
이름Text오브젝트, 대사Text오브젝트, 이미지 오브젝트, nextButton, 선택버튼, 나가기 버튼으로 이루어져 있습니다. 하지만 이름오브젝트와 대사오브젝트만 있어도 작동할 수 있습니다 나머지는 넣어도 되고 안 넣어도 상관없거나 다른걸로 대체할 수 있는 것들 입니다 여기있는것 말고도 여러분들이 추가시킬 수도 있겠죠.

기본적인 틀만 가르쳐 드릴테니 나머지는 여러분들이 자유롭게 꾸며보세요

### 0. 준비물 준비하기

몇가지 준비물이 필요합니다 당신만의 대화창을 만들고 싶다면 대화창을 꾸밀만한 재료들이 필요합니다. 대화창과 대화하는 캐릭터들의 **일러스트**를 가져오셔도 좋고 **대화창을 꾸밀만한 여러 버튼이미지들이나 배경이미지**를 들고 오셔도 좋습니다

여러 스프라이트들을 준비하셨다면 대화창에서 하게 되는 제일 중요한 대화파일을 준비하셔야 됩니다. 별거 없습니다. 메모장을 열어주세요. 그리고 대화창에서 보여주시고 싶은 대사들을 치면 됩니다.



이름:대사 순서대로 입력해주세요

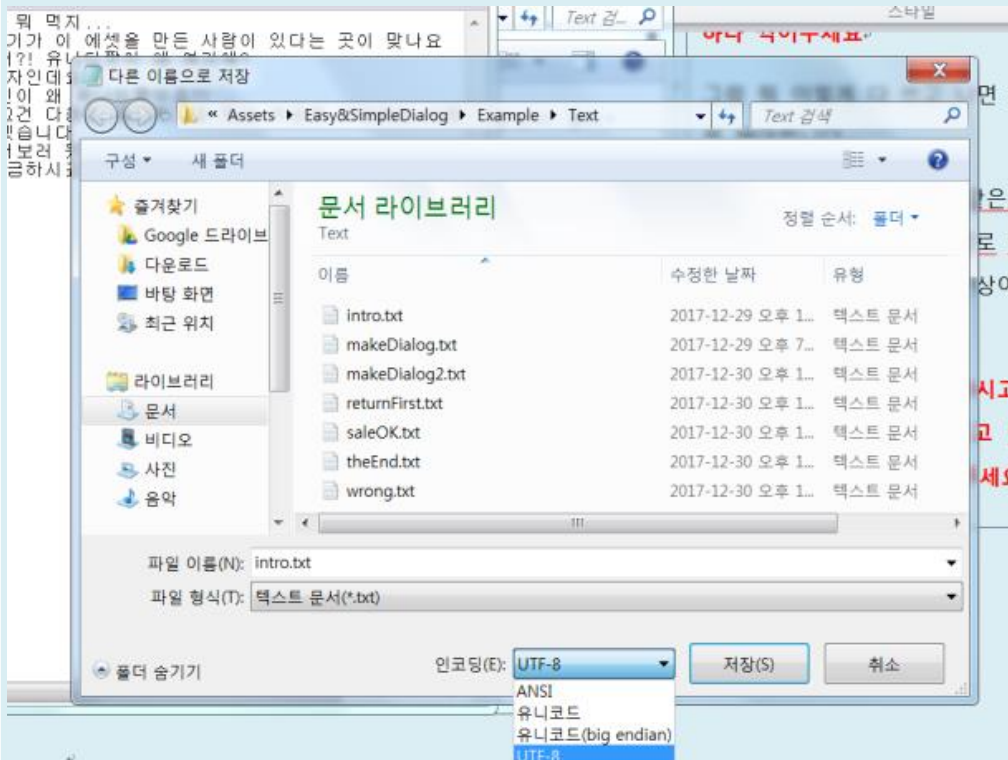
한 대사가 끝났으면 **엔터**를 쳐주시면 됩니다

그리고 **마지막 줄은 꼭 콜론(:)을 하나 찍어주세요**

그럼 이렇게 다 쓰고 나면 저장해야 합니다.

여기서 주의할점!!! 한글같은 경우나 일부 언어는 인코딩문제로 유니티로 불러오면서 깨지는 현상이 있습니다.

그래서 **그냥 저장하지 마시고 다른이름으로 저장을 누르시고 인코딩 방식을 UTF-8로 설정해주세요**



이렇게 저장하시면 됩니다

그럼 준비물 준비는 끝납니다

## 1. 스크립트 넣기

SimpleDialogMgr 스크립트를 삭제되거나 비활성화되지 않을만한 오브젝트 중

아무곳에나 넣어도 상관없습니다. 저는 관리하기 쉽도록 대화창 오브젝트의 최상위 부모오브젝트에 넣었습니다

Hierarchy 창에서 스크립트를 넣고싶은 오브젝트를 하나 선택하고

Asset 창에있는 SimpleDialogMgr 스크립트를 Inspector 창으로 드래그해주시면 끝입니다

여러가지 설정이 보이시겠지만 아직은 건드리지 마세요

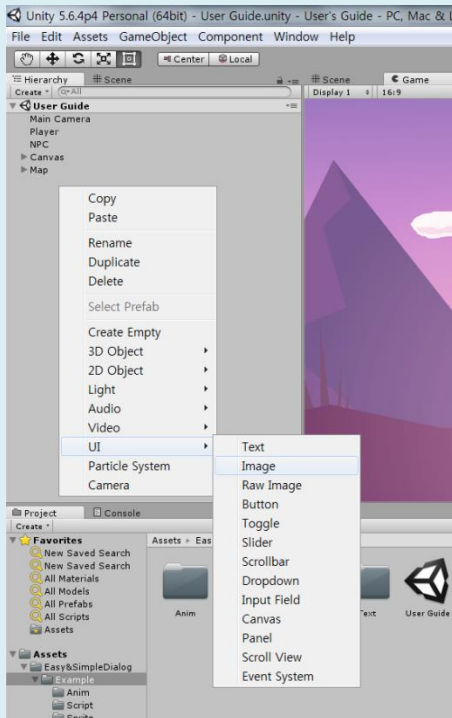
## 2. 대화창 배경을 만들자

일단 본격적으로 만들기전에 아셔야할것이 있습니다

대화창을 구성하는 모든 것들을 포함하는 부모오브젝트가 꼭 있어야 합니다

저는 대화창 배경을 대화창을 구성하는 오브젝트 중에 제일 상위인 부모오브젝트로 만들었습니다만 꼭 배경이 부모오브젝트일 필요는 없습니다.

지금부터 나오는 방법은 기본틀만 알아두시고 자유롭게 만드시면 됩니다.



일단 배경이니까 이미지가 들어가야 하겠죠

마우스 오른쪽 버튼을 눌러서 UI에 Image오브젝트를 만들어 줍니다. 그 다음은 맘대로 크기조절 하시고 꾸미세요

다만 꾸미는것들은 Hierarchy창에서 대화창배경 안으로 드래그 해서 넣으시면 됩니다

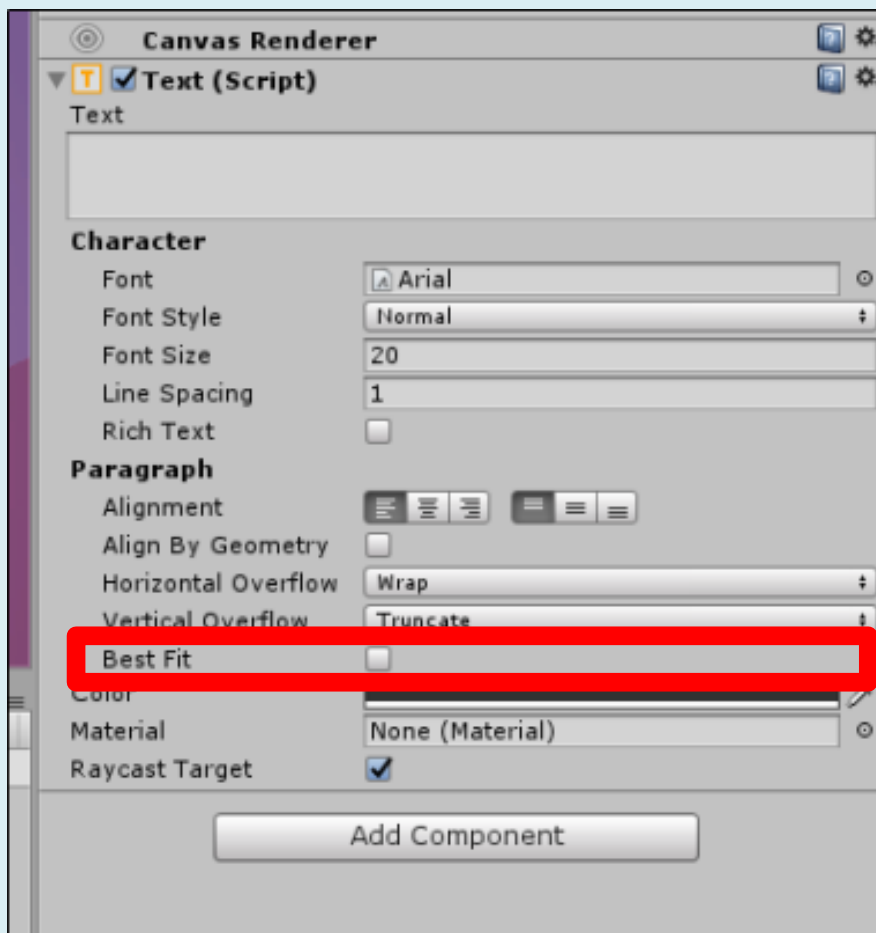
### 3. 이름Text오브젝트랑 대사Text오브젝트를 만들자

이번에는 대화창배경 오브젝트 하위로 마우스 오른쪽키를 누르고 **UI-Text**를 눌러서

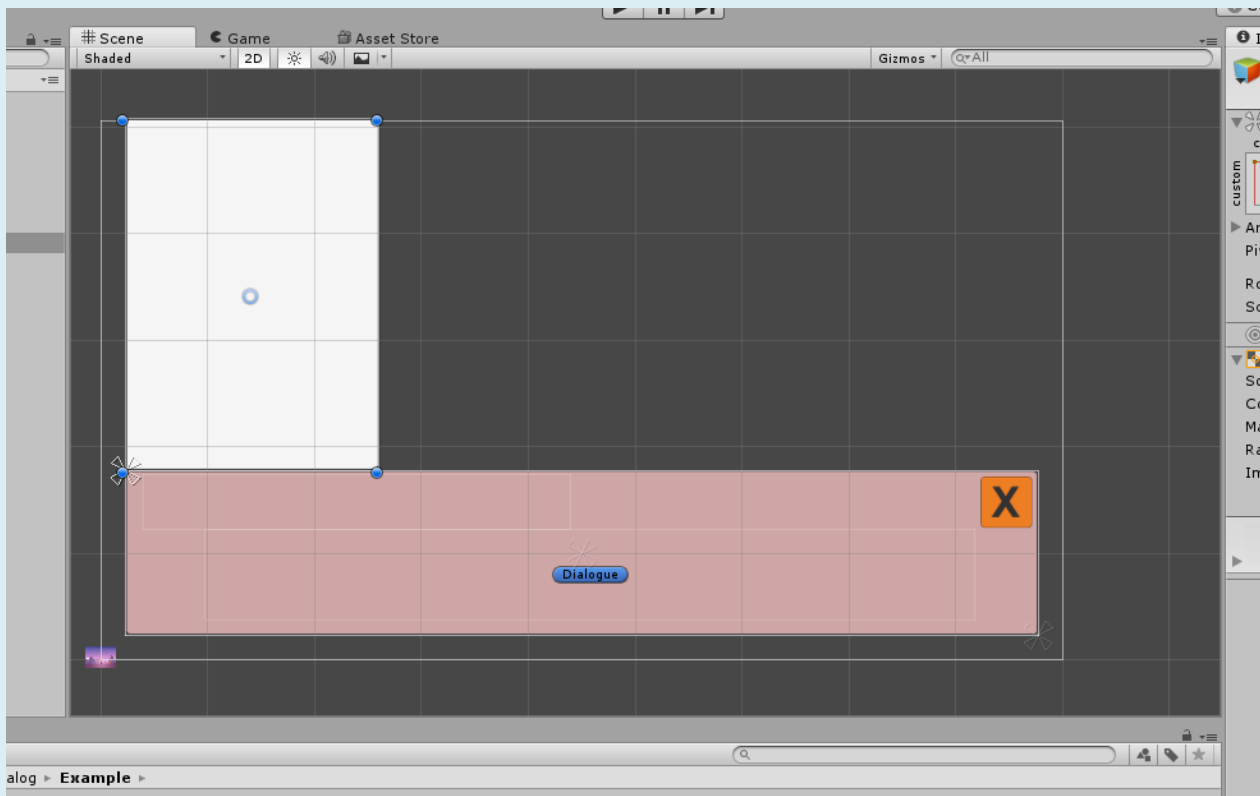
**Text오브젝트**를 두개 만들어줍니다. 각각 **이름**과 **대사**를 표시하는 것이니까  
**넉넉한 크기**로 만들어줍니다 글씨체랑 글씨색도 정해주시면됩니다  
Text안에 아무것도 안쓰셔도 괜찮습니다.

한가지 주의사항이 있습니다

**대사Text오브젝트에 보시면 Text컴포넌트가 있을텐데 거기서 BestFit이라는 설정을 해제**  
**해주세요 이걸 켜면 정상적으로 작동하지 않습니다.**



#### 4. 이미지오브젝트를 만들자(선택사항)



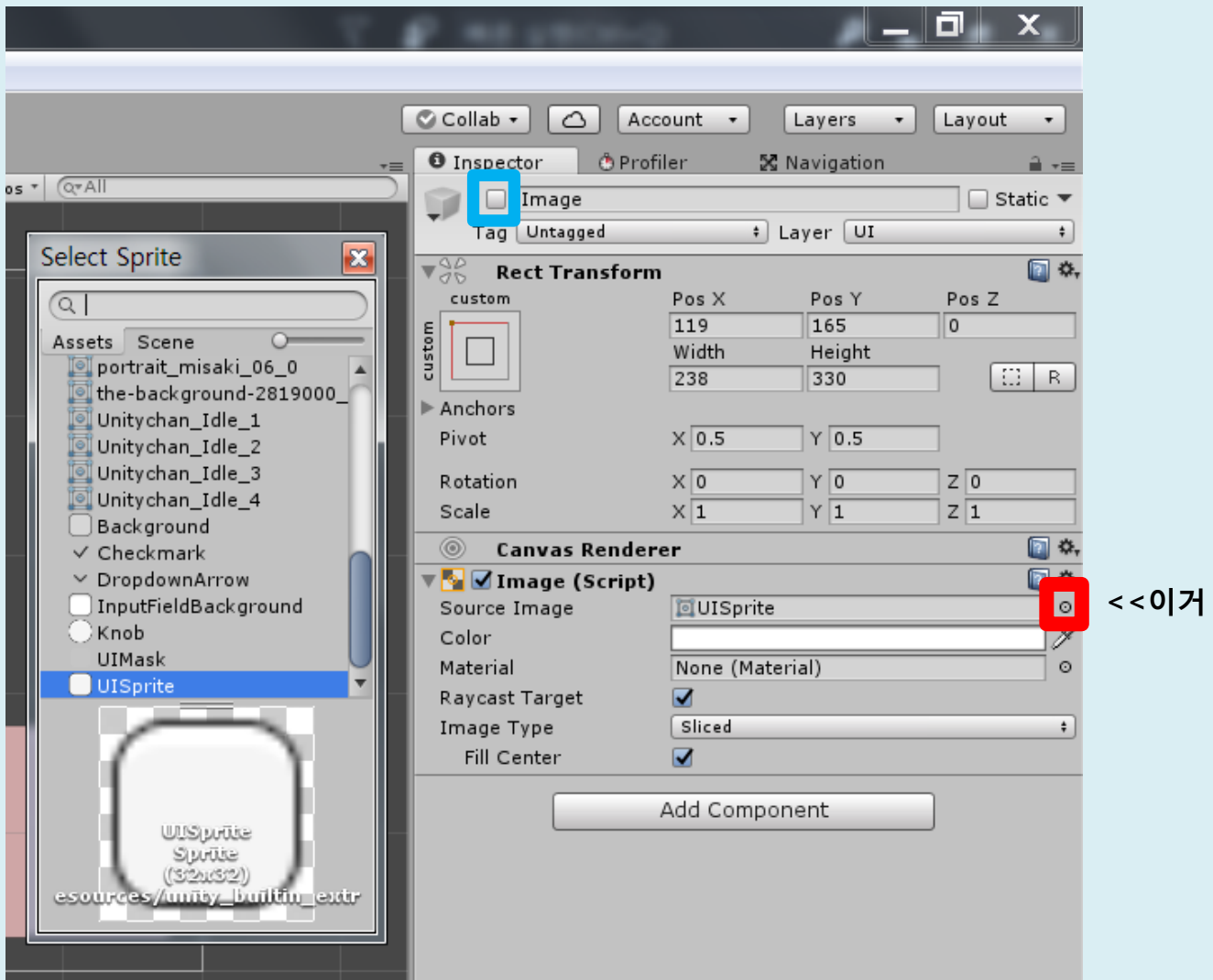
대충 저는 이정도 크기로 만들었습니다

마우스 오른쪽키 누르고 **UI-Image**해서 만들어주세요 이것도 대화창배경 오브젝트의 자식으로 넣어주세요 여기가 일러스트가 들어갈 자리니까 넣으실 **일러스트의 비율에 대략 맞춰서** 크기조절 해주세요 안 그러면 일러스트가 찌그러집니다.

그리고 보시면 아시겠지만 이미지가 있어야합니다 흰색 배경으로

**Image컴포넌트의 색상을 흰색으로** 맞춰주시고





빨간 네모박스 표시해둔버튼을 누르시면 Select Sprite라는 창이 뜹니다  
 저기서 **UISprite**를 찾아서 넣어주시면 됩니다

그리고 파란 네모박스있는곳에 체크되었을텐데 클릭해서 체크해제한다음  
**이미지오브젝트를 비활성화** 해주세요

## 5. nextButton을 만들자(키보드로 대체가능)

대화를 넘기는 버튼입니다

여러 버튼이 많이 나올 테니 부모오브젝트로 묶어 두는게 좋겠네요

싫으면 안 묶어도 됩니다. UI-Button눌러서 버튼을 만듭니다

이것도 대화창배경의 자식오브젝트로 넣어주세요 그리고 "계속하시겠습니까"라는 글자를 넣던가 멋진 버튼을 만들던가 요령껏 꾸며주세요 하지만 **꾸미는것들은 nextButton의 자식오브젝트로 넣어주세요**

그리고 중요한거 하나 대화가 넘어가기 위해선 스크립트랑 연결해야합니다

1. nextButton오브젝트를 선택하고 Inspector창을 보세요
2. 아래쪽에 Add Component 라고 적힌 버튼하나 보이시죠 누르세요
3. 그리고 Event누르시고 EventTrigger를 눌러서 추가합니다
4. Add New Event Type을 누릅니다.
5. Point Up을 누릅니다.
6. +를 눌러서 목록을 하나 추가합니다
7. None이라고 적힌곳에 아까 스크립트를 넣었던 오브젝트를 드래그해서 넣습니다
8. No Function을 누릅니다
9. SimpleDialogMgr를 누르고 skipButtonControl() 누릅니다

끝입니다 앞으로 비슷한거 많이 하니까 기억해두세요

+키보드로 대체하고 싶다면..

SimpleDialogMgr스크립트를 보면됩니다

**Void Update()**함수가 주석처리 되었을겁니다

주석처리를 해제해주시고 KeyCode.Space라고 되었을겁니다

**KeyCode.** 뒤에 원하는 키를 넣어주시면 됩니다

## 6. 나가기 버튼을 만들자(선택사항)

대화 도중에 대화창을 닫을수있는 버튼입니다. 없어도 괜찮습니다.

이것도 똑같이 적당한곳에 UI-Button 만들어주시고 적당히 사이즈조절 및 꾸며주신다음

방금처럼 이벤트트리거를 추가합니다. 그리고 Point Up 으로 설정하신 다음

None에다가 SimpleDialogMgr스크립트가 들어있는 오브젝트를 끌어옵니다.

SimpleDialogMgr를 누르고 DisableDialog() 누릅니다

끝입니다

## 7. 선택문을 만들자(선택사항)

이제 제일 어려운 선택문이 남았습니다.

이제 빈 오브젝트를 하나 만들어주세요

빈 오브젝트의 자식으로 선택을 하게되는 여러 버튼오브젝트들이 들어갑니다.

선택버튼들을 이쁘게 만들어주시고 위치도 적당히 배치해주시면 됩니다.

그리고 버튼에 이벤트트리거를 추가해서 원하는 행동을 하게

함수를 연결해주시면 됩니다.

근데 그렇게 끝나면 좋으련만 선택버튼을 누르고

그뒤로 대화가 이어지는 일이 있단 말이죠.

1. 그럴땐 Choice스크립트를 **"눌렀을 때 대화가 이어질 버튼"**에 추가해줍니다.
2. 방금 Choice스크립트를 넣은 오브젝트를 끌고와서 그 오브젝트의 이벤트트리거에 넣어줍니다. (자기자신을 이벤트트리거에 넣으라는것)
3. 이어지게 될 대화파일을 만들어서 Choice스크립트의 DialogFile에 넣어줍니다.
4. 이미지를 쓰고있다면 이미지 배열에 다음대화에 쓸 일러스트를 위에서부터 순서대로 넣어줍니다.
5. 그리고 처음에 선택버튼들을 넣은 빈오브젝트를 비활성화해줍니다.

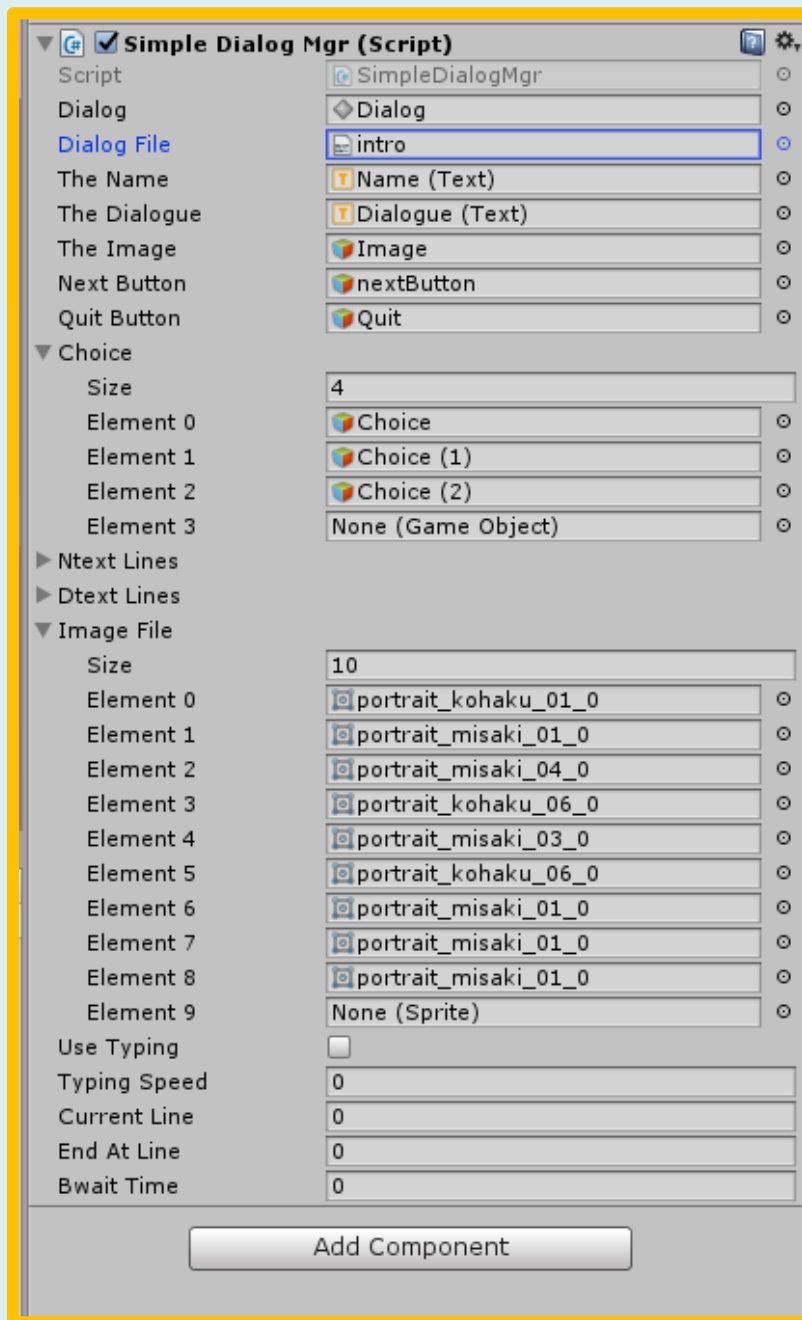
일단 이러면 끝입니다 나머지 설정들은 **3.설정창을 컨트롤 해보자**에서 자세히 설명해드리겠습니다.

## 8. 대화창 만들기 마무리

1. SimpleDialogMgr 스크립트를 오브젝트를 누르고 Inspector 창을 보세요.
2. SimpleDialogMgr 스크립트를 넣은 오브젝트를 Dialog 에 드래그해서 넣습니다..
3. 처음에 나올 대화파일을 DialogFile 에 넣습니다.
4. 이름 Text 오브젝트를 The Name 에 넣습니다.
5. 대사 Text 오브젝트를 The Dialogue 에 넣습니다.
6. 일러스트를 쓴다면 아까 만든 이미지오브젝트를 The Image 에 넣습니다.  
(+비워두면 자동으로 비활성화됩니다)
7. nextButton 을 쓴다면 nextButton 오브젝트를 nextButton 에 넣습니다.  
(+키보드로 대체하더라도 아무 오브젝트라도 넣습니다)
8. 나가기버튼을 쓴다면 나가기버튼을 QuitButton 에 넣습니다.  
(+비워두면 자동으로 비활성화됩니다)
9. Choice 버튼을 쓴다면 나올순서대로 위에서부터 차례대로 넣습니다.  
버튼 안 써도 빈 엘리먼트 하나 추가해 주기  
(+쓰는거보다 하나더 추가해서 마지막줄은 비워둡니다)
10. 일러스트를 쓴다면 일러스트파일을 위에서부터 나올 순서대로 넣는다.  
(+쓰는거보다 하나더 추가해서 마지막줄은 비워둡니다)
11. 대화창을 구성하는 최상위에있는 부모오브젝트를 비활성화합니다..
12. 어떤 방법으로든 대화창을 다시 활성화 시키면 작동하기 시작합니다

# 3. 설정창을 컨트롤 해보자

## 1. SimpleDialogMgr.cs



Dialog: 대화창 오브젝트

DialogFile: 대화창 파일

TheName: 이름 Text 오브젝트

TheDialogue: 대화 Text 오브젝트

TheImage: 이미지 오브젝트

Choice: 선택문 오브젝트들

NtextLines: 이름배열

DtextLines: 대사배열

ImageFile: 일러스트 파일들

UseTyping: 타이핑효과를 쓸건가

TypingSpeed: 타이핑속도

CurrentLine: 대화 진행변수

EndAtLine: 마지막 대사

BwaitTime: nextButton 대기시간

Choice: 선택버튼의 상위오브젝트들이 들어갑니다  
기본적으로 위에서부터 차례대로 나옵니다

ImageFile: 위에서부터 순서대로 나옵니다 중간에 이미지를 넣지않으면 아무것도 나오지않게 하는것도 가능합니다. 대화창파일에 나오는 대사들보다 한줄 크게 해주세요

UseTyping: 오토타이핑 효과를 끄고 결수있습니다

TypingSpeed: 0 에 가까울수록 빨라지며 음수가 되면 안됩니다

CurrentLine: 1 부터 시작해서 (배열크기-1) 까지 1 씩 커집니다  
//진행용변수이기에 함부로 건드리면 안됩니다

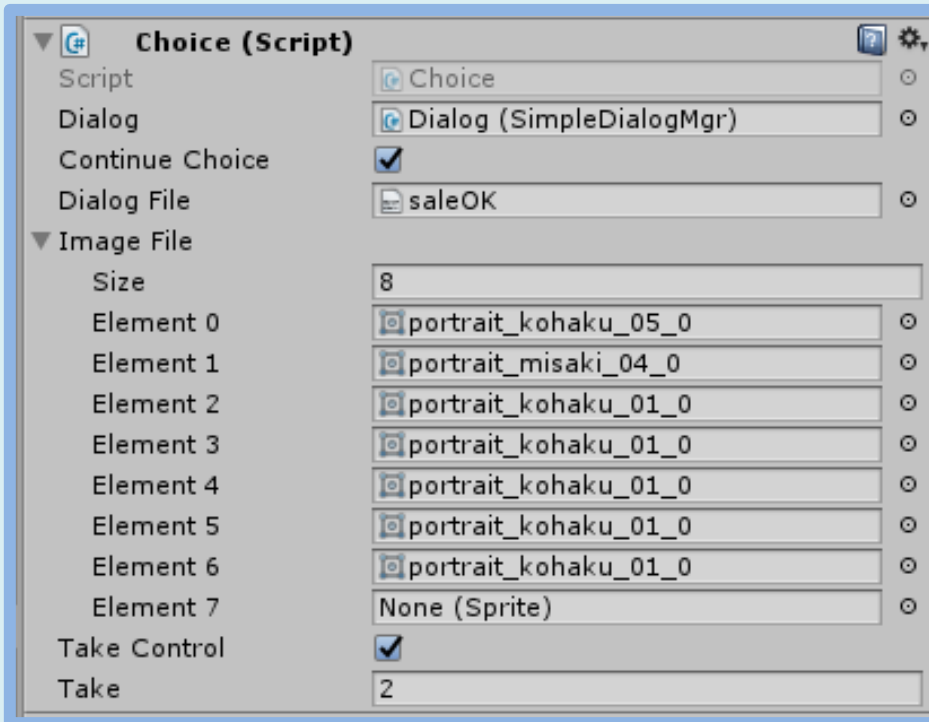
EndAtLine: (배열크기-1) //이것도 절대 건드리면 안됩니다

DisableDialog() :대화창 비활성화 함수입니다 대화창을 끄고 싶다면  
이 함수를 실행해서 비활성화 해주세요  
그리고 오토타이핑효과가 작동중이여서  
아직 대사가 다 출력되지 않았을때는  
절대 비활성화하면 안됩니다

ReadyDialog(TextAsset) :대화창 준비함수입니다 매개변수에 대화창파일을 넣어서  
대화파일을 바꿉니다 Choice 스크립트를 통해 간접적으로  
조작하도록 하고 직접 건드리는건 추천하지 않습니다

ChoiceOnOff() :다음 take 에서도 선택문을 계속할건지 결정하는 함수  
Choice 스크립트를 통해 간접적으로 조작하도록 하고 직접 건드리는건  
추천하지 않습니다

## 2. Choice.cs



Dialog: 대화창 스크립트

DialogFile: 이어질 대화파일

ContinueChoice: 이어질 대화에 선택문이 있는지 없는지

ImageFile: 이어질 대화에 들어갈 일러스트 파일

TakeControl :take변수를 컨트롤 할건지 안할건지

Take: 선택문 진행변수

ContinueChoice: 이 변수는 이어질 대화에 선택문이 있는지 없는지 정하는 변수입니다  
체크하게되면 다음take에 해당하는 Choice배열의 선택버튼이 나오게되고  
체크안하면 take변수는 커지지않고 선택버튼도 나오지 않습니다.

Take : 0부터 시작해서 (Choice배열크기-1)까지 선택버튼을 누를때마다 1씩 커집니다

“take변수=Choice배열의 인덱스번호” 라고 할수있습니다

기본적으론 Choice배열은 Choice[0]부터 순서대로 나오지만 take 변수를 컨트롤하면 사용했던 선택버튼이나 아직 사용하지 않은 선택버튼을 미리 쓸수있습니다. 자칫 잘못하면 오류가 생길 가능성이 높기 때문에 Choice배열의 크기에 주의해서 쓰기 바랍니다

TakeControl: Take변수를 컨트롤하고싶다면 체크해야합니다

체크하지않으면 아무리 take변수의 값을 바꿔도 적용되지 않습니다



## 4. 스크립트를 커스터마이징해보자

대화창을 통해 효과를 주고싶다던가

새로운 기능을 추가하고 싶다는 생각이 들수있습니다.

그 확장을 위한 팁을 드리려고 합니다

그 대부분의 열쇠는 currentLine변수,take변수,선택버튼의 이벤트트리거에 있다고 생각합니다.

그 예시로 예제파일에 Dfileguide라는 오브젝트도 선택버튼의 이벤트트리거로 만들어졌습니다. 선택버튼의 이벤트트리거로 오브젝트를 활성화시키는 간단한 원리죠

currentLine변수랑 take변수를 이용하면 어느 대사에 있는지 알수있습니다

그렇다는건 조건문으로 특정 대사가 될 때 이벤트를 발생시킬수있습니다

선택버튼의 이벤트트리거는 더욱더 많은 활용이 가능합니다 이벤트트리거와 특정 스크립트의 함수를 연결할 수 있다는 것만으로도 수많은 활용이 가능합니다

예를 하나들자면 검이라는 선택버튼을 누르면 이벤트트리거에 입력되어있는 1500이라는 매개변수가 특정스크립트의 함수에 전달되서 돈이라는 변수가 1500만큼 감소하게 만드는 것도 가능합니다

**별거 아닌 이 팁들이 멋진게임을 만들어냈으면 좋겠습니다**

**여기까지 읽어주셔서 정말 감사합니다**

**다들 파이팅하세요^^**