

라이브러리에 오신 것을 환영합니다

Roblox Story Games 에서는 단어를 비우고 플레이어가 그 빈 부분을 채우도록 합니다! 이 1 시간 프로젝트에서 Roblox 의 무료 코딩 및 디자인 툴을 사용해 스토리 게임을 직접 만들어 보세요! 코딩 지식을 증명해서 전용 배지와 아바타 아이템 등을 획득할 수도 있죠.

아래 비디오를 재생해서 작업을 진행할 라이브러리와 게임 작동 방식을 살펴보세요.



코딩을 위한 여정

3 개의 수업을 통해 여러분만의 스토리 게임을 만들어가게 됩니다. 각 수업에서는 여러분의 게임을 코딩하는 방법을 처음부터 끝까지 다룹니다.



수업을 완료하면, Roblox 의 퀴즈 게임을 플레이해 보상을 획득할 기회가 생겨요. 보상을 획득한 다음 다시 돌아와 다음 수업을 진행하세요.



Roblox Studio 사용 준비

Roblox 에서 볼 수 있는 모든 게임은 Roblox Studio 를 통해 만들어집니다. 이미 Roblox Studio 를 설치했다면, 로그인한 후 이 페이지 하단의 다음 버튼을 클릭하세요.

1. Roblox Studio 를 다운로드해 설치합니다.

[ROBLOX STUDIO 다운로드](#)

2. 설치 후 바탕화면 아이콘(Windows)를 두 번 클릭하거나, Dock 아이콘을 클릭하세요(Mac).

3. 로그인 화면에서 Roblox 사용자 이름과 비밀번호를 입력한 다음 Log In 을 클릭하세요.

새 계정 만들기

Roblox 를 처음 이용하신다면, [여기를 클릭](#)해 가입하여 계정을 만드세요. 계정을 안전하게 보호하려면 다음 팁을 기억하세요.

- **비밀번호를 공유하지 마세요.** 실제 아는 친구와도 공유하면 안 돼요.
- **짐작하기 어려운 비밀번호를 사용하세요.** — 예를 들어 사용자 이름이 "bloxcool"이라면, 비밀번호로 "bloxcool123"을 사용해서는 안 돼요.
- **Roblox 직원은 절대 사용자의 비밀번호를 묻지 않아요.** — 비밀번호를 물어보는 사람이 있다면 [불량 이용자 신고](#) 기능을 사용해 신고하세요.
- **공짜 Robux 를 준다는 말에 속지 마세요.** — Robux 를 공짜로 얻을 수 있는 비밀을 알고 있다고 주장하는 플레이어나 웹사이트의 이야기는 절대 믿지 마세요!

더 많은 팁을 보려면 [계정을 안전하게 보호하는 법](#)을 참조하세요.

스토리 만들기

코딩하기 전에, 게임의 기본이 될 스토리를 만드세요. 어떤 이야기든 괜찮아요.

- 종이나 문서 작성 프로그램을 사용해, 첫 단락의 2~3 문장 정도를 작성하세요. 시작하기 위한 몇 가지 아이디어를 드릴게요.
 - 여러분의 가장 친한 친구가 꿈에 그리던 휴가
 - 마법사의 아침 식사
 - 슈퍼히어로의 쇼핑 여행

예시 스토리

한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 Nia. 매일 아침 이 마법사는 꿀을 발라 구운 도마뱀 나뭇잎 한 사발을 먹는답니다.

자리 표시자 만들기

스토리를 기반으로 한 단어 게임을 만들려면, 스토리에서 단어를 삭제하고 그 자리를 대신할 자리 표시자를 생성하세요. 게임에서 플레이어는 이 자리 표시자를 채울 질문에 답변을 입력하게 됩니다. 첫 번째 문장부터 시작할까요.

1. 첫 번째 문장에서 이름, 동작, 명사 등 플레이어가 대체할 수 있는 단어 하나를 선택해보세요.

단어 선택

한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 Nia.

2. 플레이어에게 요청할 질문의 답변에 해당하는 단어를 자리 표시자로 대체합니다.

원래 단어	자리 표시자 단어
한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 Nia .	한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 name .

3. 플레이어에게 명사 하나 또는 형용사 하나 이상의 답변을 받아야 하는 경우, 자리 표시자에 숫자를 붙이세요.

최종 자리 표시자
한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은
name1.

이제 이 스토리는 코드로 바뀔 준비가 된 거예요!

템플릿 열기

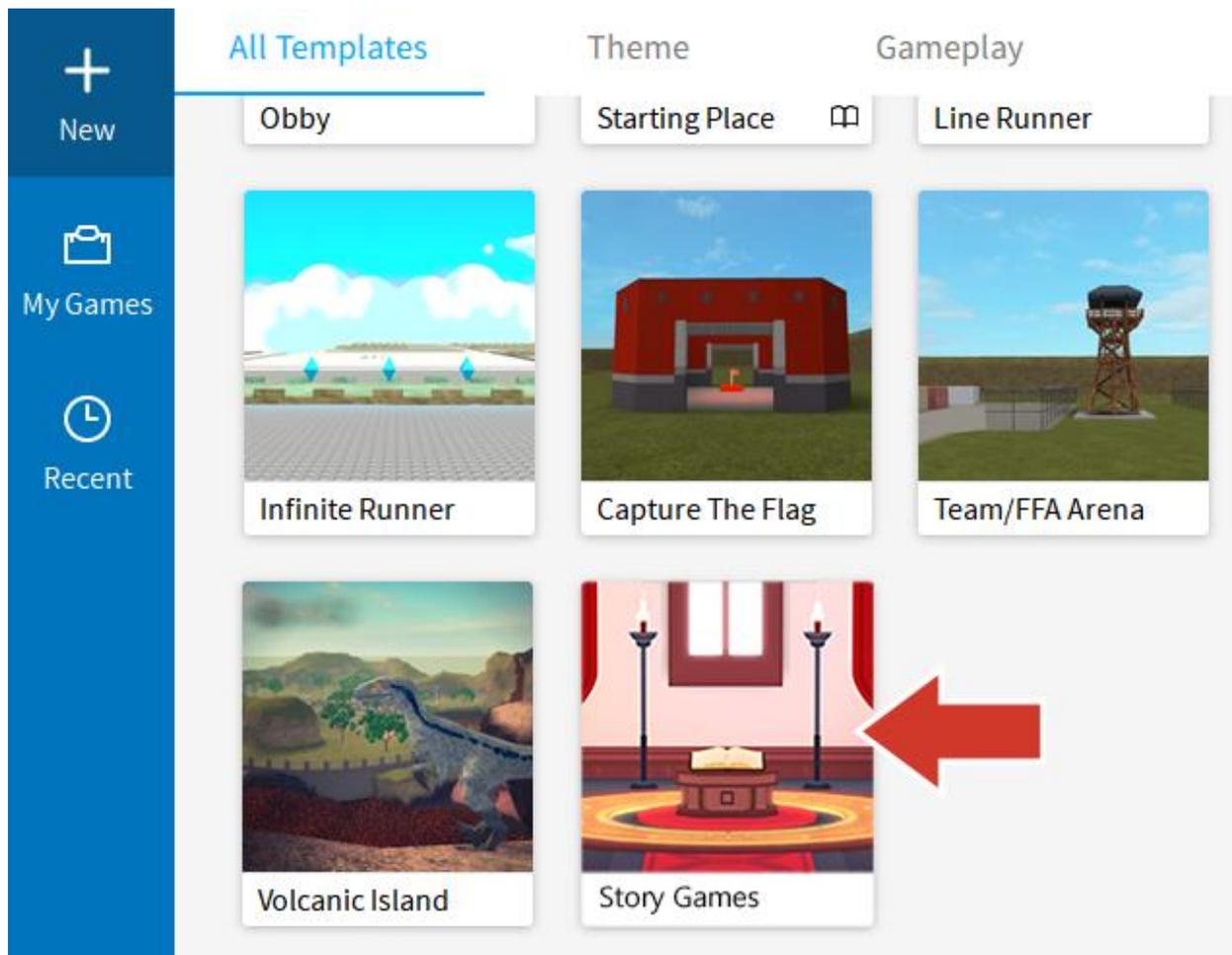
게임은 **Roblox Studio** 를 통해 만들어져요. 무료로 사용할 수 있고 iPhone, Android, Xbox Live, PC, Mac, VR 에 즉시 게임을 게시할 수 있죠.

1. 파란색 바탕화면 아이콘(Windows)를 두 번 클릭하거나, Dock 아이콘을 클릭(Mac)해서 **Roblox Studio** 를 여세요.
2. 로그인 화면에서 Roblox 사용자 이름과 비밀번호를 입력한 다음 **Log In** 을 클릭하세요.

템플릿 열기

Roblox Studio에는 실제 스토리에 대한 코드를 제외하고, 게임을 작동시키는 데 필요한 모든 것이 들어 있는 **Story Games** 템플릿이 있어요. 템플릿은 미리 제작된 세계로, 여러분의 게임을 위한 기반으로 사용할 수 있죠.

- **Story Games** 템플릿을 한 번 클릭하세요.



템플릿이 보이지 않나요?

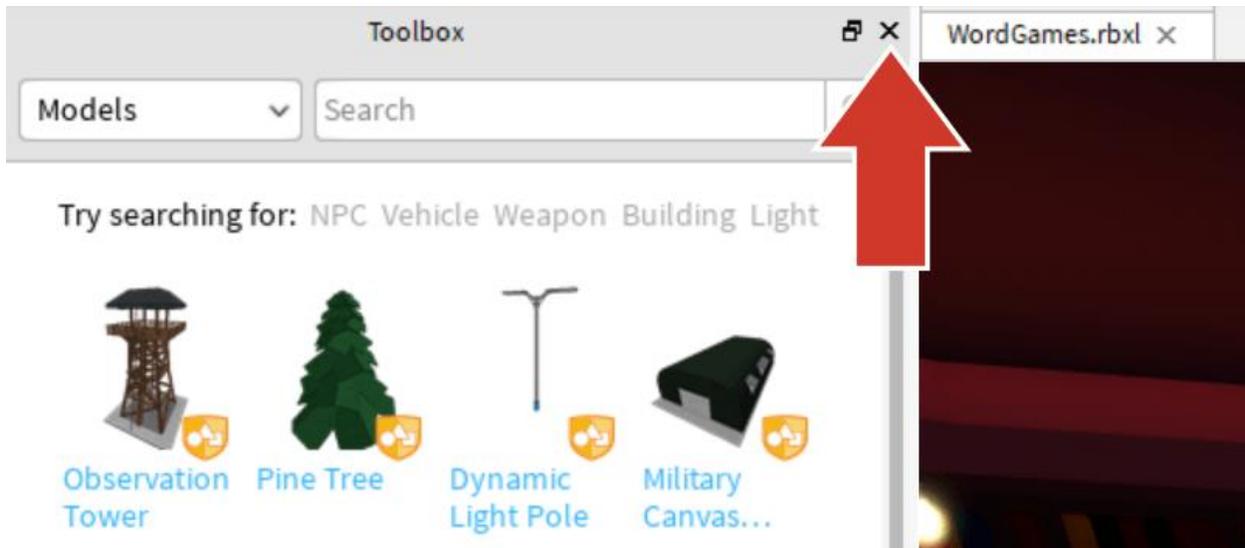
템플릿이 보이지 않는다면 **New** → **All Templates** 을 클릭하세요.

추가 창 닫기

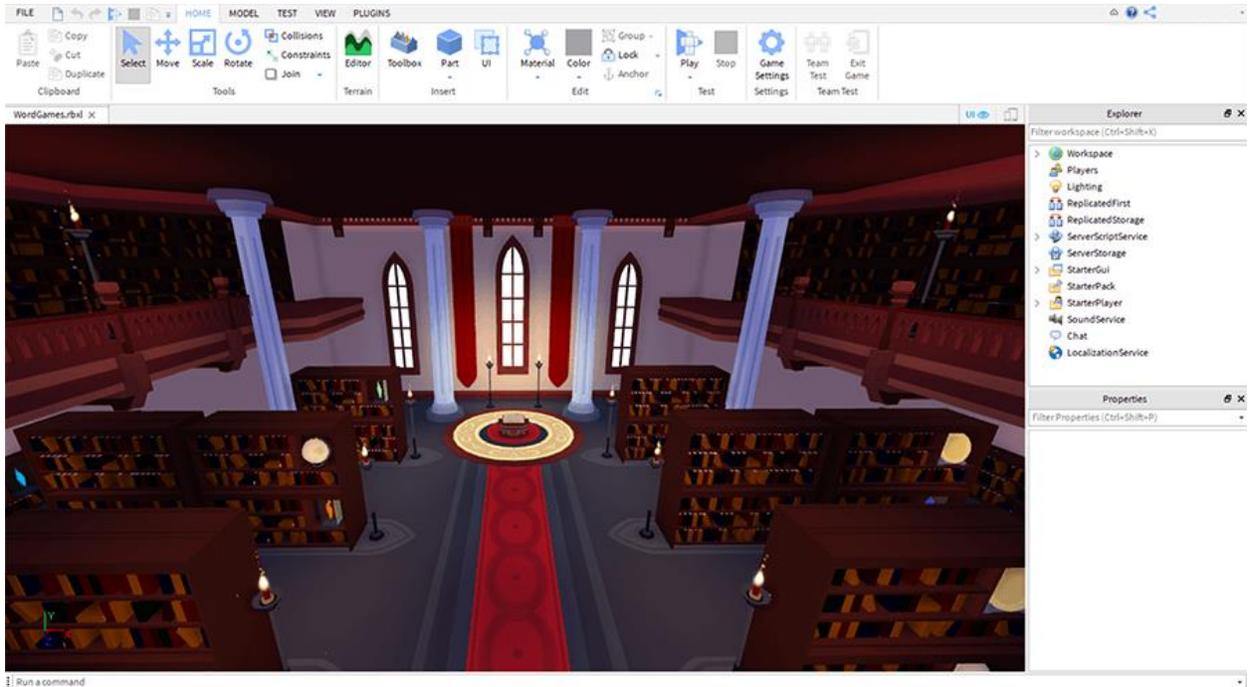


Roblox Studio 를 처음 실행하는 경우, 당장은 필요하지 않은 창이 추가로 하나 더 열릴 수 있어요. 이러한 창을 닫으면 작업할 공간이 더 넓어지죠.

1. x를 클릭해 Studio 의 **왼편**에 있는 창을 모두 닫으세요. 닫을 창이 없는 경우, 다음 단계로 이동하세요.



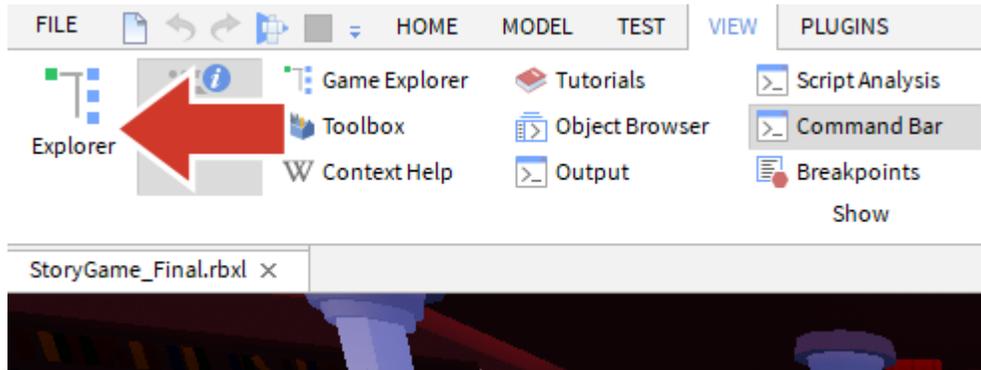
2. 오른쪽의 Explorer 창은 그대로 두세요. Studio 가 아래 이미지처럼 보여야 합니다.



Explorer 창이 보이지 않는 경우

Explorer 창이 보이지 않는 경우 아래 단계대로 해보세요.

1. **View** 탭을 선택하세요.
2. 제일 왼쪽에 있는 **Explorer** 아이콘을 선택하세요.

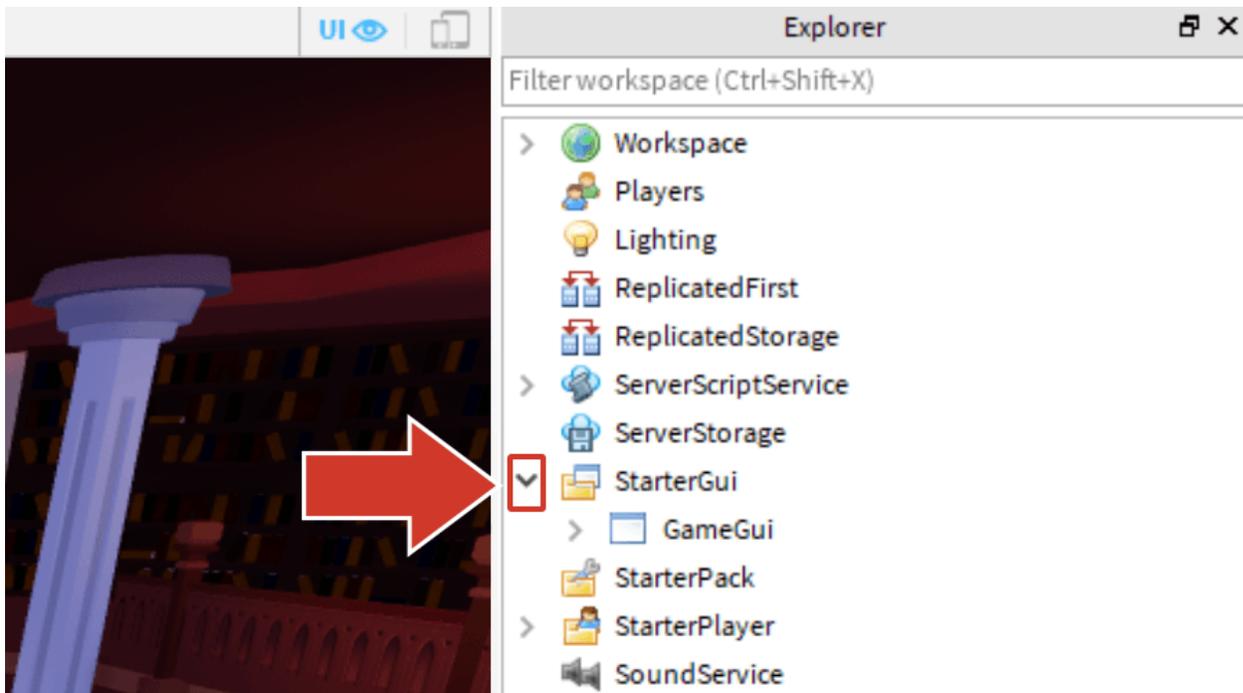


스크립트 열기

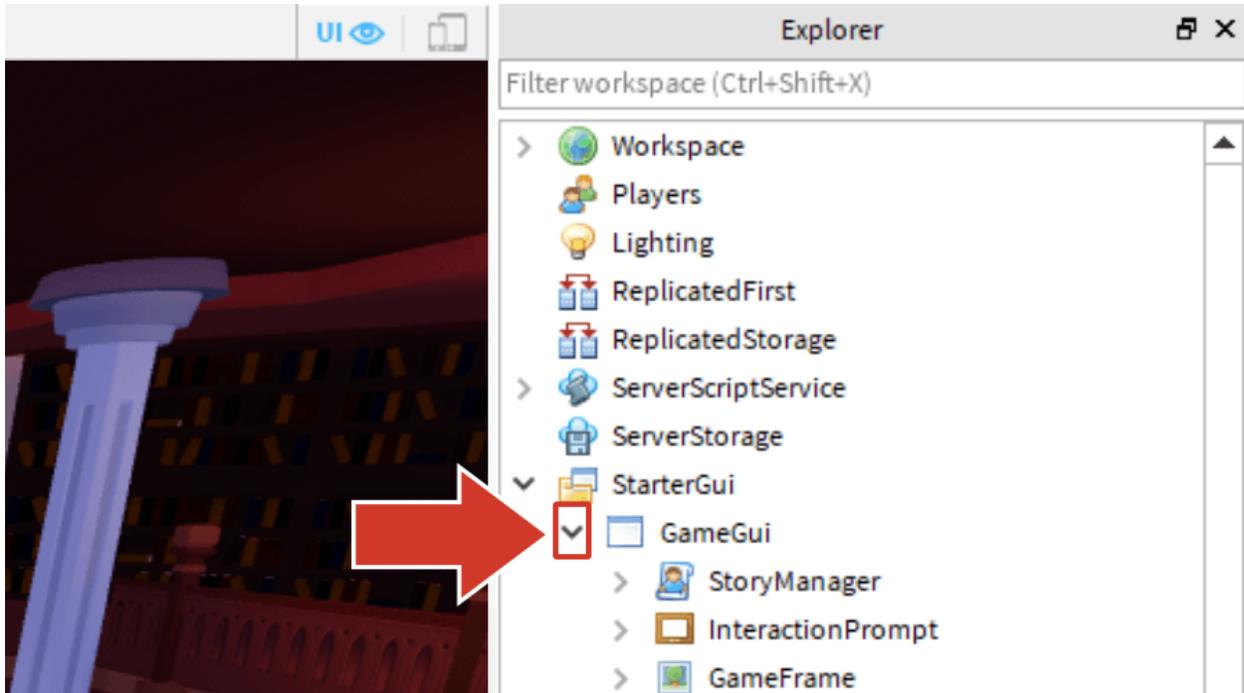
Roblox 에서 코드는 코드 언어인 **Lua** 를 사용해 **스크립트** 내에 입력돼요. 게임은 게임에서 필요한 각 요소에 대한 여러 개의 개별적인 스크립트로 이루어지죠. 라이브러리 템플릿에는 여러분만의 단어 게임을 위해 더 많은 코드를 추가할 수 있는 **StoryManager** 라는 스크립트가 이미 마련되어 있습니다.

StoryManager 스크립트 찾기

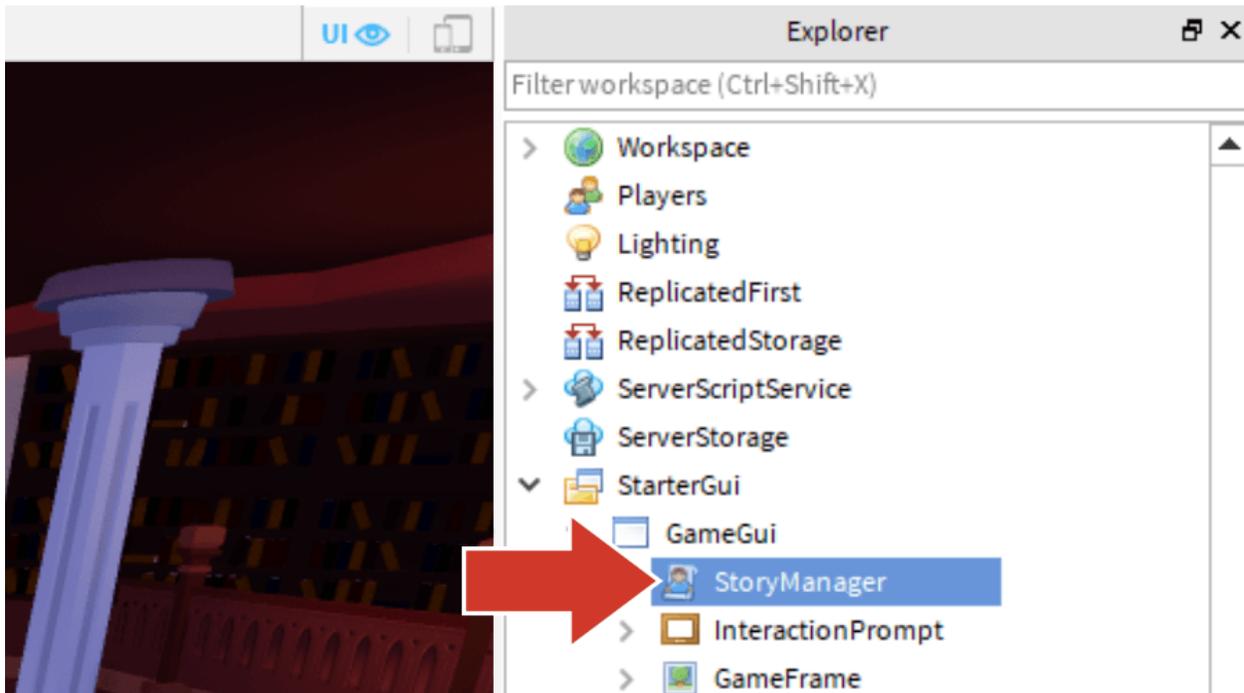
1. Explorer 창에서, **StarterGUI** 옆의 화살표를 선택해 아래 포함된 모든 항목을 확인하세요.



2. **GameGUI** 옆 화살표를 클릭해 해당 섹션을 확장하세요.



3. StoryManager 스크립트를 두 번 클릭해 여세요.



StoryManager 스크립트

스크립트에는 플레이어에게 완전한 스토리를 보여주기 위한 필수 코드가 이미 포함되어 있어요. 여러분이 작성한 모든 코드는 아래 구분선 사이에 입력되죠.

```
--@polarpanda16 에게 특별한 감사를 전합니다.  
  
-- 글로벌 변수  
local storyMaker = require(script:WaitForChild("StoryMaker"))  
  
-- 게임 제어 코드  
local playing = true  
  
while playing do  
    storyMaker:Reset()  
  
    -- 구분선 사이 스토리에 대한 코드 입력  
    -- =====  
  
    -- =====  
  
    -- 아래 괄호 안에 스토리 변수 추가  
    storyMaker:Write()  
  
    -- 다시 플레이할래요?  
    playing = storyMaker:PlayAgain()  
end
```

주석

--로 시작되는 코드 줄은 **주석**이에요. 주석은 여러분을 위해서나 다른 코드 제작자를 위한 메모를 남기는 데 사용되죠. 프로그램의 작동 방식에는 영향을 미치지 않아요.

코딩 변수

첫 번째로 코드를 작성할 항목은 플레이어가 채울 단어의 자리 표시자예요. 스크립트에서 정보에 대한 자리 표시자는 **변수**라고 해요. 변수에 포함되는 정보는 필요에 따라 변경하거나 대체할 수 있죠.

변수 만들기

Lua 코드에서 변수는 `local` 과 `nameOfTheVariable` 을 입력해서 만들 수 있어요. 변수 이름은 해시태그처럼 공백 없는 하나의 단어여야 하고, 모두 소문자여야 해요. 변수 이름을 잘 지으면 해당 변수가 어디에 사용되는지 쉽게 알아볼 수 있죠.

아까 전 적어 두었던 자리 표시자 이름을 변수의 이름으로 사용하세요.

- 첫 번째 구분선 아래 `local` 을 입력한 다음 변수의 이름을 입력하세요. 이 예제 스토리에서의 자리 표시자는 `name1` 이었으므로, 첫 번째 변수는 `local name1` 이 됩니다.

원래 자리 표시자
<code>Workspace</code> 의 대소문자를 똑같이 적용해야 합니다.
코드의 자리 표시자
<pre>while playing do storyMaker:Reset() -- 구분선 사이 스토리에 대한 코드 입력 -- ===== local name1</pre>



도전하세요!

아래 버튼을 클릭해 Roblox 퀴즈 게임을 풀며 여러분의 지식을 테스트하고, 총 세 가지 보상 중 첫 번째 보상을 받아보세요. 보상을 획득한 후, 다시 여기로 돌아와서

다음 버튼을 눌러 플레이어에게 질문하는 법을 알아보세요.

[도전하기](#)

질문하기

방금 만든 변수는 플레이어에게 질문하고 그 답변을 저장하는 데 사용됩니다.

플레이어가 라이브러리의 앞쪽에 있는 큰 책을 클릭하면 플레이어의 화면 중앙에 질문이 나타나죠.



질문 추가하기

변수 안에 정보 저장하기

변수 안에 저장될 정보는 = 기호를 사용해 변경할 수 있어요. 변수는 스크립트 내부 또는 프로그램이 실행되면서 여러 번 변경될 수 있죠.

1. local name1 과 같은 변수 이름 뒤에 =를 입력하세요.

```
while playing do
  storyMaker:Reset()

  -- 구분선 사이 스토리에 대한 코드 입력
  -- =====

  local name1 =
```

다음 단계는 플레이어에게 질문하고 질문에 대한 답변을 변수 내부에 저장하는 거예요.

2. = 다음에 storyMaker:GetInput("질문")이라고 입력하세요. 대소문자가 일치하게, 쓰여진 대로 입력해야 합니다.

```
while playing do
  storyMaker:Reset()

  -- 구분선 사이 스토리에 대한 코드 입력
  -- =====

  local name1 = storyMaker:GetInput("질문")
```

storyMaker:GetInput()

스크립트는 서로를 참조해요. 이 게임에는 storyMaker 라는 이름의 두 번째 스크립트가 있고, 해당 스크립트 안에는 플레이어 입력을 위한 코드가 포함되어 있어요. 방금 작성한 변수는 storyMaker 스크립트의 코드를 실행한 후, 플레이어의 답변을 저장해요.

문자열 유형

변수는 작은 수, True 또는 False 값, 문자열을 비롯한 다양한 유형의 데이터를 저장할 수 있어요. **문자열 유형** 변수는 문장 전체를 저장할 수 있어 특별하죠.

문자열 유형의 변수를 알아보는 건 쉬워요. 언제나 "이런 식으로" 따옴표 사이에 들어가 있기 때문이죠.

- "질문" 부분을 스토리의 다음 줄을 완료하기 위해 플레이어에게 물어보고 싶은 내용으로 바꾸세요. 따옴표 사이에 입력해야 한다는 점을 잊지 마세요("").

```
while playing do
  storyMaker:Reset()

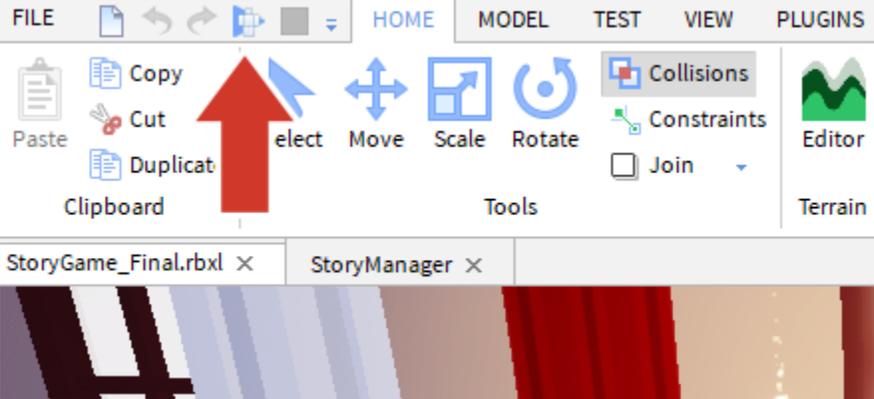
  -- 구분선 사이 스토리에 대한 코드 입력
  -- =====

  local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")
```

코드 테스트

코드를 확인하고 제대로 작동하는지 알아보기에 좋은 시점입니다. 코드를 자주 테스트하면 실수가 있는 부분이 어디인지 알아내는 게 더 쉬워지죠.

1. **Play** 버튼을 클릭해 코드를 테스트하세요.



2. 키보드의 **W A S D** 키를 사용해 기본 동작을 테스트하고 **E** 를 누르세요.

컨트롤

동작

W A S D 또는 화살표 키

캐릭터 이동

스페이스바

점프

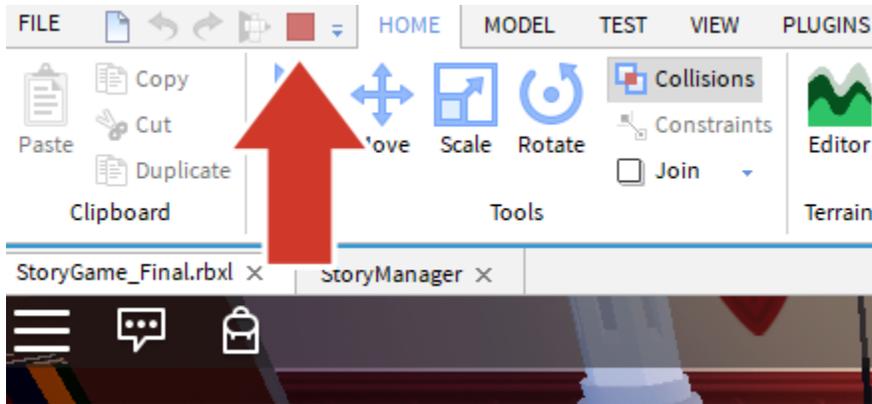
마우스 오른쪽 버튼(마우스 버튼 누른 상태에서 드래그)

주변 둘러보기

3. 입력한 질문이 표시되는지 확인하세요. 아직 질문에 답변한 후 "제출" 버튼을 누르는 건 조금 기다려주세요. 아직 코딩해야 할 스토리 부분이 남아 있으니까요.



4. 플레이 테스트를 중단하세요.

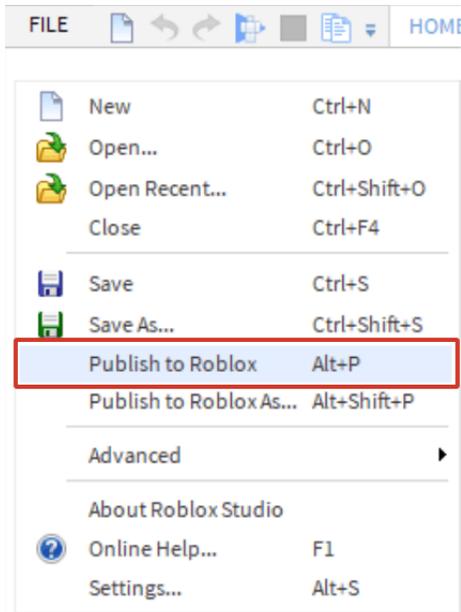


코드 저장

게임을 게시해서 작업을 저장하세요. 그러면 Roblox 에서 온라인에 해당 게임을 저장하죠. 10~15 분에 한 번씩 게임을 저장하는 것이 좋아요.

프로젝트 게시

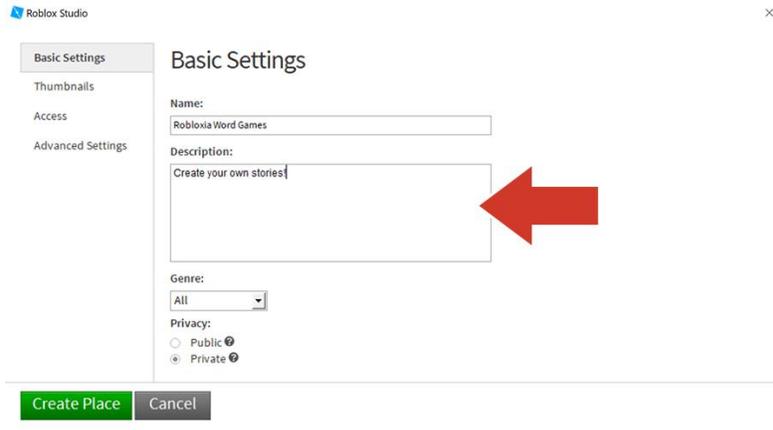
1. 플레이테스트가 중단됐는지 확인하세요.
2. **File** → **Publish to Roblox** 를 선택하세요.



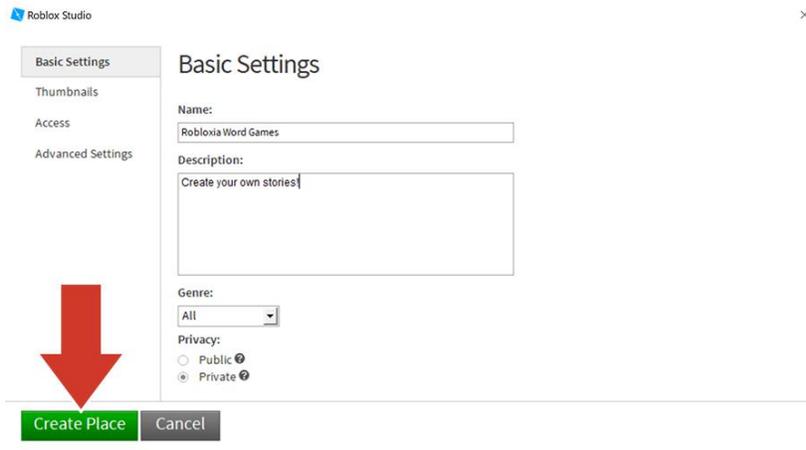
3. **New Place** 를 클릭하세요.



4. 이름과 설명을 입력하세요.



5. **Create Place** 를 선택하세요.



6. 진행 막대가 "100% Completed" 상태가 되면 파란색 **Next** 버튼을 클릭하세요.



7. **Done** 을 클릭하세요.

Gain Visitors & Make Money!

Sell gear for your game

You'll receive a 10% commission from all gear sold on your games page.

Start Now

Sell Game Passes

Grant users special access and privileges to your game with game passes. (Fees apply)

Start Now

Award Game Badges

Award users who achieve great feats rare badges and they will be more likely to play.

Start Now

Advertise Your Game

Advertise your game throughout Roblox.com and reach out to millions of players.

Start Now

Done



도전하세요!

아래 버튼을 클릭해 Roblox 퀴즈 게임을 풀며 여러분의 지식을 테스트하고, 총 세 가지

보상 중 두 번째 보상을 받아보세요. 보상을 획득한 후, 다시 여기로 돌아와서 **다음**

버튼을 눌러 여러분의 스토리를 작성하는 법을 알아보세요.

[도전하기](#)

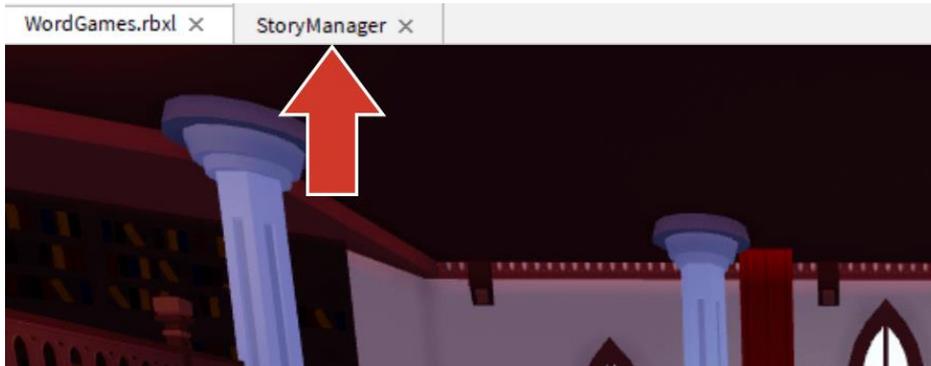
스토리 만들기

플레이어가 모든 질문에 답했다면, 해당 답변이 결합된 스토리를 볼 수 있게 돼요.

스토리는 문자열을 사용해 변수에 저장되고 플레이어의 답변이 들어 있는 문자열과 결합되죠.

첫 번째 문자열 코딩

1. 플레이테스트가 중단됐는지 확인하세요.
2. 게임 에디터 위에 있는 **StoryManager** 스크립트 탭을 클릭해 스크립트로 돌아가세요.



스크립트 다시 열기

스크립트 에디터 앱이 보이지 않는 경우, Explorer 창에서

StoryManager 를 두 번 눌러 다시 열 수 있어요.

3. 질문 입력 줄 아래에 story 라는 이름의 새로운 변수를 만드세요. 변수 이름을 소문자로 작성해야 합니다.

```
-- 구분선 사이 스토리에 대한 코드 입력
-- =====

local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")

local story
```

4. 첫 번째 문자열을 찾으려면, 원래 스토리로 돌아가세요. 첫 번째 자리 표시자 전에 있는 모든 부분에 동그라미를 치거나 강조 표시하세요. 변수가 문장 중간에 있더라도 문제 없습니다. 나머지는 나중에 추가할 수 있으니까요.

원래 자리 표시자

한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 name1

5. story 변수가 아래처럼 문자열을 포함하도록 합니다.

```
-- 구분선 사이 스토리에 대한 코드 입력
-- =====

local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")

local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 "
```

문제 해결 팁

문제 해결 - 루프

- 게임의 다른 부분에서는 정확히 일치하는 이름의 변수만 찾기

때문에 local story 는 반드시 소문자로 입력해야 해요.

- 코드는 말 그대로만을 보여주기 때문에, 문자열이 동시에 실행되는 일이 없도록 아래처럼 문자열 끝에 공백을 포함해야 해요.

"한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 "

스토리에 추가하기

스토리와 답변 결합하기

다음으로는, 스토리 첫 번째 문자열을 플레이어의 답변과 결합해야 합니다. 이렇게

결합하는 작업을 **접합**이라고 해요. 두 개의 문자열을 하나로 결합할 때는 `..`을 사용합니다.

1. `story` 변수와 동일한 줄에 `..`를 입력합니다.

```
while playing do
  storyMaker:Reset()

  -- 구분선 사이 스토리에 대한 코드 입력
  -- =====

  local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")

  local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그
이름은 " ..
```

2. 같은 줄에, 플레이어의 답변이 포함되는 변수의 이름을 입력하세요.

```
while playing do
  storyMaker:Reset()

  -- 구분선 사이 스토리에 대한 코드 입력
  -- =====

  local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")

  local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그
이름은 " .. name1
```

스토리 작성 및 플레이테스트

스토리가 입력됐으니, 이제 플레이어에게 표시되어야겠죠.

1. 두 번째 구분선 아래에 있는 storyMaker:Write()를 찾으세요. () 사이에 story 변수를 입력합니다. 이렇게 해서 프로그램에게 게임에서 해당 스토리를 쓰라고 알려주는 거죠.

```
while playing do
  storyMaker:Reset()

  -- 구분선 사이 스토리에 대한 코드 입력
  -- =====

  local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")

  local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있습니다. 그
이름은 " .. name1
```

```
-- =====  
-- 아래 괄호 안에 story 변수를 추가하세요  
storyMaker:Write (story)
```

작업 확인

storyMaker:Write(story) 줄에 story 를 입력했는지 확인하고 코드 상자와 동일하게 대소문자가 적용되어 있는지 확인하세요. 이 단계 없이는, 스토리가 플레이테스트에 나타나지 않습니다.

2. 게임에 대한 **플레이테스트**를 진행하세요. 두 문자열이 결합되어 나타나야 합니다.



문제 해결 팁

문제 해결 - 루프

질문이 표시되지 않았다면 다음을 확인하세요.

- 질문이 따옴표 안에 입력되어 있어야 합니다.

스토리가 결합되지 않았다면 다음을 확인하세요.

- 스토리의 첫 번째 부분이 따옴표 안에 입력되어 있어야 합니다.
- 플레이어 답변이 포함되는 변수의 이름이 정확히 일치해야 합니다. 대소문자도 잊지 말고 확인하세요!
- 플레이어 답변이 포함되는 변수의 이름이 따옴표 안에 포함되어 있지 않습니다.
- 두 문자열은 ..을 사용해 구분됩니다.

스토리가 표시되지 않는 경우

- story 변수를 storyMaker:Write()의 ()에 입력했는지 확인하세요.

문장 완성하기

더 많은 단어나 구두점을 추가하고 싶다면, 문자열을 추가하세요.

1. story 변수와 같은 줄에 ..를 하나 더 추가합니다.

- 나머지 문장 또는 구두점을 포함한 문자열을 추가하세요. 문장 뒤 공백 하나를 추가해야 한다는 점 잊지 마세요.

```
-- =====  
  
local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")  
  
local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 "  
.. name1 .. ". " ..
```

두 번째 질문 추가

두 번째 질문을 하려면, 새로운 질문을 만들어 스토리를 포함하고 있는 동일한 변수에 추가하면 됩니다.

- 스토리의 두 번째 문장에서 어떤 단어를 대체할 것인지 결정하세요.

원래 자리 표시자
한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 name1. 매일 아침 이 마법사는 꿀을 발라 구운 food1 한 사발을 먹는답니다.

- 첫 번째 변수 아래, 자리 표시자 역할을 할 새로운 변수를 만드세요.

```
-- 구분선 사이 스토리에 대한 코드 입력
-- =====

local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")
local food1

local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 "
.. name1 .. ". " ..
```

참고: 변수 이름

변수 이름은 소문자로 작성해야 하며 공백을 포함하지 않아야 한다는 점에 유의하세요.

3. storyMaker:GetInput("")을 사용해 플레이어에게 질문하고 답변을 저장합니다.

```
-- 구분선 사이 스토리에 대한 코드 입력
-- =====

local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")
local food1 = storyMaker:GetInput("당신이 좋아하는 요리는 무엇인가요?")
```

4. story 변수에 ..를 사용해 다음 스토리 문자열을 접합하세요. 문장 끝에 공백 하나를 포함해야 한다는 점 잊지 마세요.

```
-- 구분선 사이 스토리에 대한 코드 입력
-- =====
```

```
local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")
local food1 = storyMaker:GetInput("당신이 좋아하는 요리는 무엇인가요?")

local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 "
.. name1 .. ". " .. "매일 아침 이 마법사는 꿀을 발라 구운 "
```

5. 새로운 스토리 문자열 다음, **두 번째** 질문에 대한 답변을 접합하고 마침표로 끝내세요.

```
-- 구분선 사이 스토리에 대한 코드 입력
-- =====

local name1 = storyMaker:GetInput("당신이 좋아하는 이름은 무엇인가요?")
local food1 = storyMaker:GetInput("당신이 좋아하는 요리는 무엇인가요?")

local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 "
.. name1 .. ". " .. "매일 아침 이 마법사는 꿀을 발라 구운 " .. food1 .. "한
사발을 먹는답니다. "
```

스토리 완료하기

더 많은 변수, 질문, 접합 문자열을 추가해 게임에 요소를 추가해 가세요.

1. 스토리에 더 많은 줄 추가
2. 새로운 변수 및 문자열을 추가할 때마다 게임 플레이테스트

팁과 요령

변수 한 번 이상 사용하기

변수는 한 번 이상 사용할 수 있어요. 단어를 포함하려는 문자열 사이에서 접합을 사용하면 되죠

코드 예시

```
local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은  
" .. name1 .. ". " .. "매일 아침 " .. name1 .. " 마법사는 꿀을 발라 구운 " .. food1  
.. " 한 사발을 먹는답니다. "
```

결과

한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 Sameth. 매일
아침 Sameth 마법사는 꿀을 발라 구운 감자 한 사발을 먹는답니다.

줄 바꿈 추가

문자열에 \n 을 입력해서 줄 바꿈을 추가할 수 있어요.

줄 바꿈 1 개 추가

코드 예시
<pre>local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 " .. name1 .. ". " .. "\n 매일 아침 이 마법사는 꿀을 발라 구운 " .. food1 .. " 한 사발을 먹는답니다. "</pre>
결과
<pre>한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 Sameth. 매일 아침 이 마법사는 꿀을 발라 구운 감자 한 사발을 먹는답니다.</pre>

줄 바꿈 2 개 추가

코드 예시
<pre>local story = "한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 " .. name1 .. ". " .. "\n\n 매일 아침 이 마법사는 꿀을 발라 구운 " .. food1 .. " 한 사발을 먹는답니다. "</pre>
결과
<pre>한 언덕의 나무 한 그루에 대마법사가 살고 있었습니다. 그 이름은 Sameth. 매일 아침 이 마법사는 꿀을 발라 구운 감자 한 사발을 먹는답니다.</pre>



도전하세요!

아래 버튼을 클릭해 Roblox 퀴즈 게임을 풀며 여러분의 지식을 테스트하고, 총 세 가지 보상 중 마지막 보상을 받아보세요. 보상을 획득한 후, 다시 여기로 돌아와서 다음 버튼을 눌러 제작자 도전을 완료하는 법을 알아보세요.

[도전하기](#)

도전 완료하기



축하합니다, 마지막 Roblox 제작자 도전을 완료하셨네요! 게임을 완성했으니, 게임을 더 재미있게 만들 수 있는 몇 가지 아이디어와 추가 학습 방법을 알려드릴게요.

게임 공유

지금까지 진행한 여러분의 작업은 여러분만 볼 수 있는 클라우드에 저장되어 있어요.

Roblox에서는 모두 자신이 만든 작품을 친구와, 전 세계에 공유할 수 있어요!

공유 방법 배우기

나만의 스토리 추가

도전을 완료했지만, 스토리에 추가할 수 있는 요소는 더 많답니다. 몇 가지 아이디어를 알려드리죠.

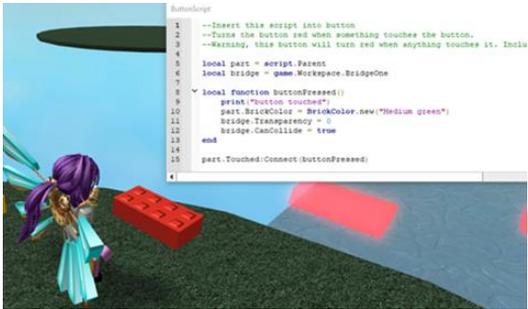
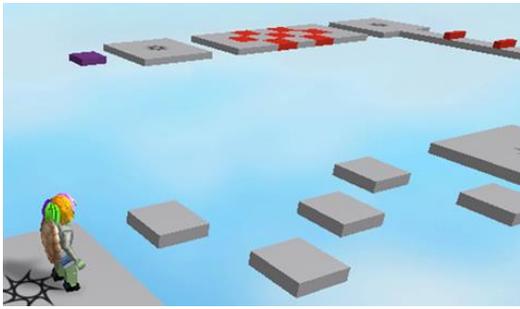
- **캐릭터 추가** – 스토리에 캐릭터를 1~2명 더 추가해 보세요. 각 캐릭터에 대한 새로운 변수를 생성해야 합니다. 예를 들면 name2 또는 name3과 같은 식으로요.
- **스토리에 더 많은 내용 추가** – 스토리에 몇 문장을 더 추가하고, 대체할 수 있는 부분을 선택하세요.

추가 학습

장애물 코스 설계, 속도 증가 부스트 코딩 등 게임 제작에 관한 더 많은 것을 배우고

싶으신가요? [개발자 허브](#)에 있는 강의를 확인하세요. Roblox 게임을 제작하기 위한

튜토리얼을 찾아볼 수 있는 사이트입니다.

 <pre>ButtonScript 1 --Insert this script into button 2 --Turns the button red when something touches the button. 3 --Warning: this button will turn red when anything touches it. Enjoy 4 5 local part = script.Parent 6 local bridge = game.Workspace.BridgeOne 7 8 local function ButtonPressed() 9 print("button touched") 10 part.BrickColor = BrickColor.new("Medium green") 11 bridge.Transparency = 0 12 bridge.CanCollide = true 13 end 14 part.Touched:Connect(ButtonPressed) 15 16</pre>	
<p>코딩 수업</p> <p>이 단계별 수업에서 코딩에 대해 더 많은 것을 알아보세요.</p> <p>보기</p>	<p>제작 수업</p> <p>장애물 코스를 만들고 맞춤 설정해 보세요.</p> <p>보기</p>

These documents are licensed by Roblox Corporation under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#). Roblox, Powering Imagination, and Robux are trademarks of Roblox Corporation, registered in the United States and other countries. The Hour of Code™ is a nationwide initiative by [Computer Science Education Week](#) and [Code.org](#) to introduce millions of students to one hour of computer science and computer programming.