

[AI 챗봇 코딩]

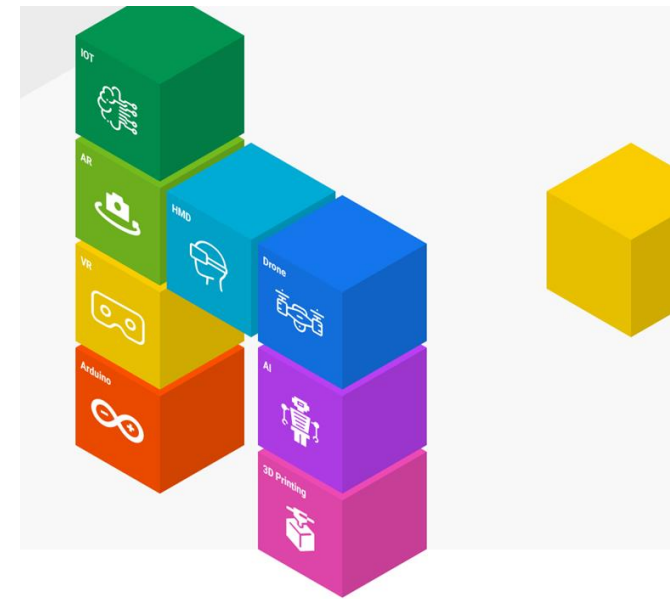
나만의 ChatGPT AI 앱 만들기
(한글 및 영어 음성 기능 추가하기)



www.helloapps.co.kr

김 영 준 / 070-4417-1559 / splduino@gmail.com

최종 결과물 예시

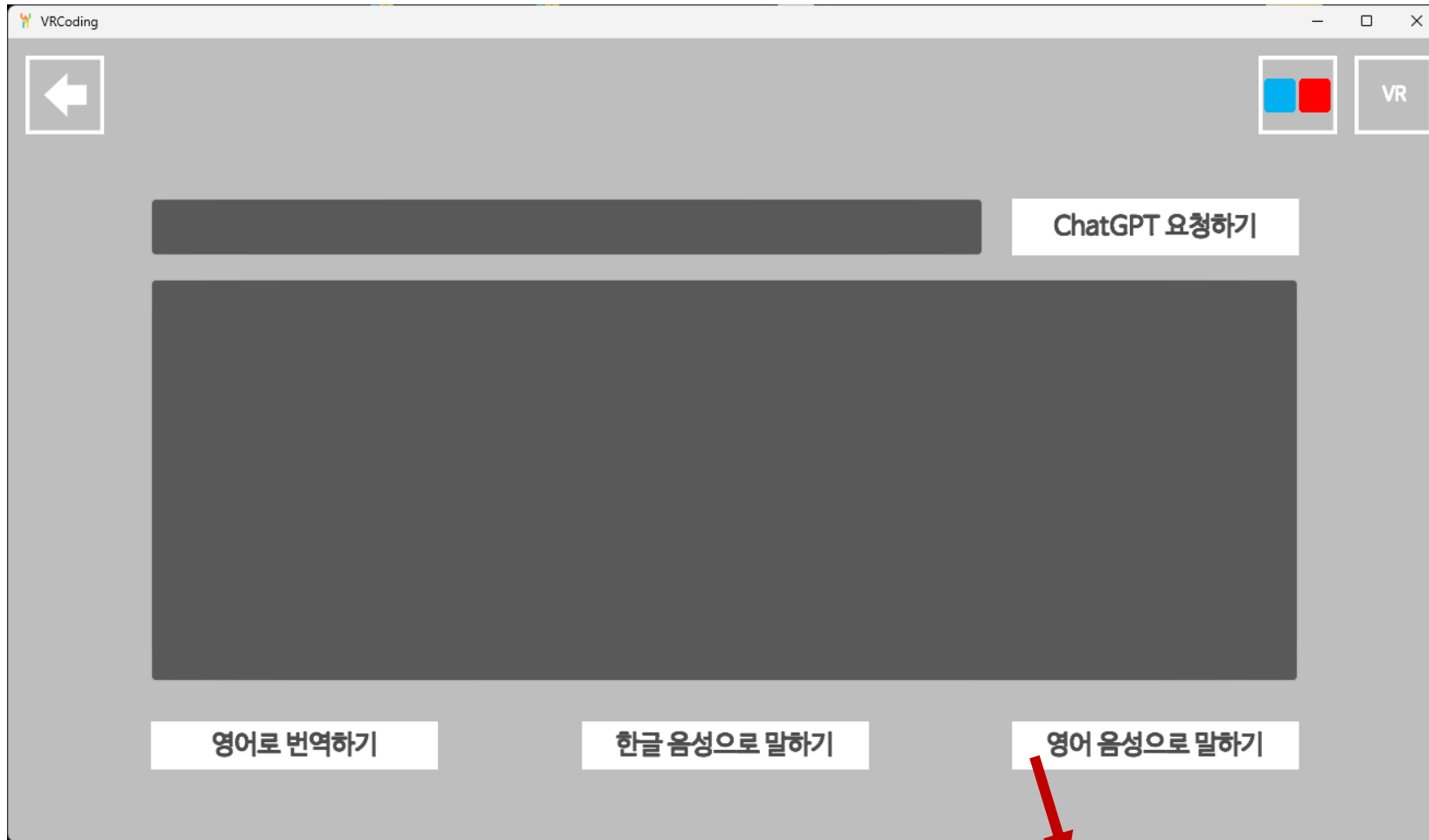


최종 결과물 화면 예시

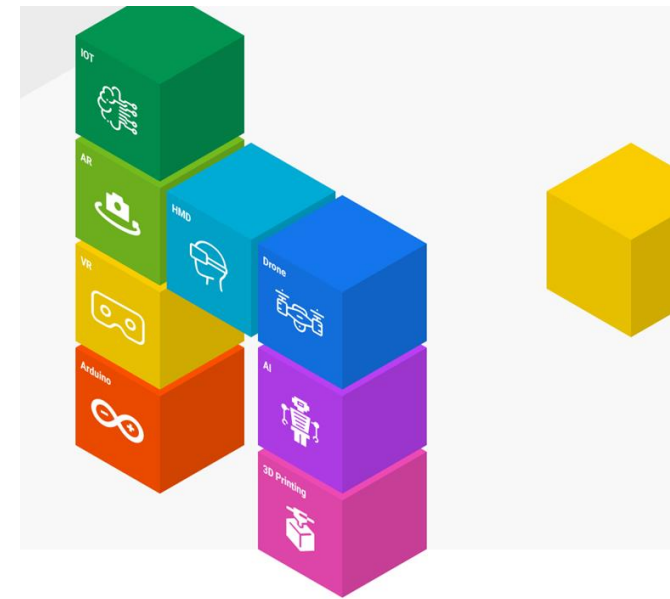


ChatGPT의 결과물을 한글 음성으로 말하게 합니다.

최종 결과물 화면 예시



기존 ChatGPT UI 프로그램 만들기 스크립트 준비하기



기존 실습 스크립트



- 하늘 설정하기 ^ x
 - 색상:190, 190, 190, 255 x
- 조이스틱 숨기기 x
- 텍스트 박스 추가하기 tb1 ^ x
 - 위치 비율:0.1, 0.2 x
 - 크기 비율:0.58, 0.07 x
- 버튼 추가하기 btn1 ^ x
 - 글자 내용:ChatGPT 요청하기 x
 - 위치 비율:0.7, 0.2 x
 - 크기 비율:0.2, 0.07 x
 - 클릭 함수:f1 x

- 텍스트 편집기 추가하기 ta1 ^ x
 - 위치 비율:0.1, 0.3, 0 x
 - 크기 비율:0.8, 0.5, 0 x
- 함수 void f1 () x
 - ta1 문자열 쓰기("ChatGPT에 요청중입니다...") x
 - t = tb1 문자열 값 읽기() x
 - ChatGPT 요청하기 ^ x
 - 챗 텍스트:{t} x
 - 응답 함수:f2 x
- 함수 void f2 () x
 - ta1 문자열 쓰기(value) x

기존 실습 스크립트



버튼 추가하기 btn2



글자 내용:영어로 번역하기



위치 비율:0.1, 0.85



크기 비율:0.2, 0.06



클릭 함수:f3



버튼 추가하기 btn3



글자 내용:한글 음성으로 말하기



위치 비율:0.4, 0.85



크기 비율:0.2, 0.06



클릭 함수:f4



버튼 추가하기 btn4



글자 내용:영어 음성으로 말하기



위치 비율:0.7, 0.85



크기 비율:0.2, 0.06



클릭 함수:f5



기존 실습 스크립트



함수 void f3() ✕

t = ta1 문자열 값 읽기() ✕

번역하기 ^ ✕

글자 내용:{t} ✕

입력 언어:한국어 ✕

번역 대상 언어:영어 ✕

저장할 변수 이름:t ✕

● ta1 문자열 쓰기(t) ✕

기존 실습 스크립트

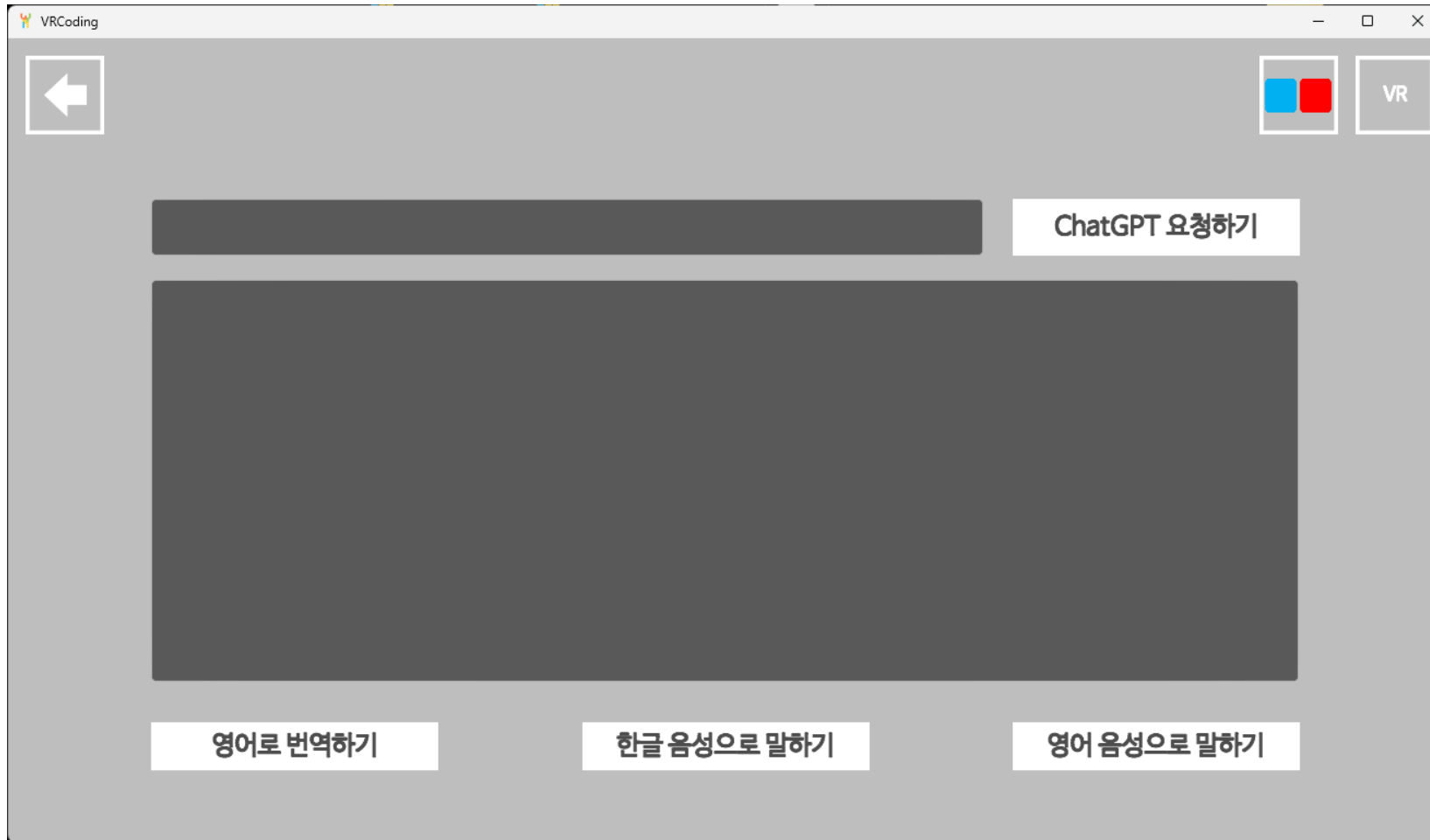


```
SetSky
    /Color:190, 190, 190, 255
HideJoystic
AddTextBox tb1
    /PositionRatio:0.1, 0.2
    /SizeRatio:0.58, 0.07
AddButton btn1
    /Text:ChatGPT 요청하기
    /PositionRatio:0.7, 0.2
    /SizeRatio:0.2, 0.07
    /OnClick:f1
AddTextArea ta1
    /PositionRatio:0.1, 0.3, 0
    /SizeRatio:0.8, 0.5, 0
void f1()
{
    t = tb1.GetText()
    RequestChatGPT
        /Prompt:{t}
        /OnResponse:f2
```

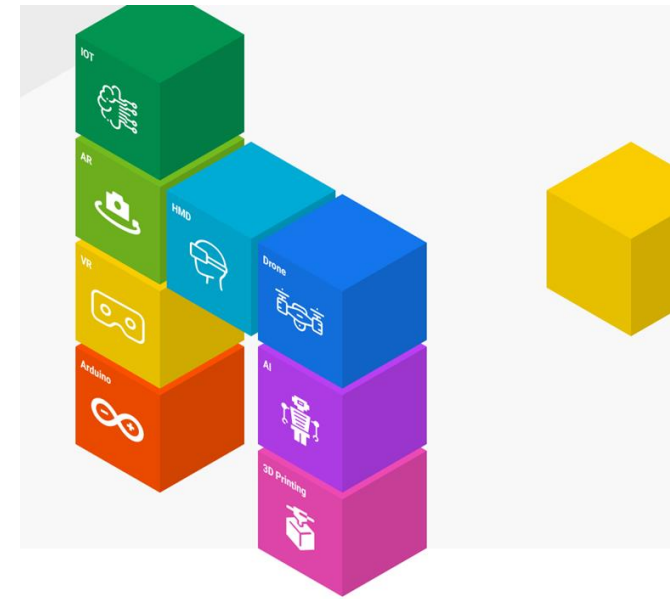
```
void f2()
{
    ta1.SetText(value)
}
AddButton btn2
    /Text:영어로 번역하기
    /PositionRatio:0.1, 0.85
    /SizeRatio:0.2, 0.06
    /OnClick:f3
AddButton btn3
    /Text:한글 음성으로 말하기
    /PositionRatio:0.4, 0.85
    /SizeRatio:0.2, 0.06
    /OnClick:f4
AddButton btn4
    /Text:영어 음성으로 말하기
    /PositionRatio:0.7, 0.85
    /SizeRatio:0.2, 0.06
    /OnClick:f5
```

```
void f3()
{
    t = ta1.GetText()
    TranslateText
        /Text:{t}
        /SourceLanguage:Korean
        /TargetLanguage:English
        /TargetVariable:t
    ta1.SetText(t)
}
```

기존 실습 스크립트



한글 음성 말하기 기능 추가



한글 음성 말하기 기능 추가



로직 명령어 그룹에 있는 함수를 하나 추가합니다. 함수 이름을 f4()로 수정합니다.

The screenshot shows a VR development interface with three main panels:

- VR 명령어 (VR Commands):** A menu on the left with a sub-menu '로직 명령어' (Logic Commands) containing: 수식 명령어 (Math Commands), 문자 출력 (Text Output), 문자라인 출력 (Text Line Output), 기다리기 (Wait), if, else, else if, for, while, and 함수 (Function).
- 목록모드 (List Mode):** A central panel showing a list of commands. The '함수 void f4()' (Function void f4()) is highlighted in red. A red arrow points from the '함수' (Function) option in the left menu to this item.
- 컴포넌트 (Component):** A right panel showing a 3D scene with a '속성 함수' (Property Function) component. The function name 'void f4()' is displayed in a text box, with a red arrow pointing from the highlighted function in the list to this text box.

A text box at the bottom right contains the text: "함수의 이름을 f4()로 수정합니다." (Change the function name to f4()).

한글 음성 말하기 기능 추가



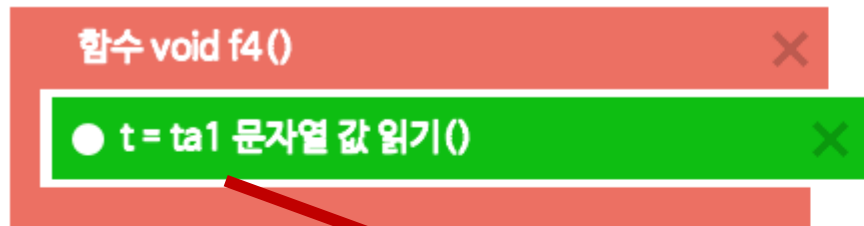
UI 명령어에서 문자열 값 읽기 명령어를 추가합니다.

The screenshot shows a software development interface with two main panels. On the left, a sidebar titled 'VR 명령어' (VR Commands) contains a list of UI commands. The 'UI 명령어' (UI Commands) section is expanded, showing options like '버튼 추가하기' (Add Button), '레이블 추가하기' (Add Label), '텍스트 박스 추가하기' (Add Text Box), '텍스트 편집기 추가하기' (Add Text Editor), '이미지 추가하기' (Add Image), '문자열 쓰기' (Write String), and '문자열 값 읽기' (Read String Value). The '문자열 값 읽기' option is highlighted in green. A red arrow points from this option to the right panel. The right panel, titled '블록모드' (Block Mode) and '텍스트모드' (Text Mode), shows a list of code blocks. The first block is '저장할 변수 이름:t' (Variable name to store: t). The second block is 'ta1 문자열 쓰기(t)' (Write string ta1(t)). The third block is '함수 void f4()' (Function void f4()). The fourth block is 't = tb1 문자열 값 읽기()' (t = Read string value from tb1()), which is highlighted in green and has a radio button selected next to it, indicating it is the active command.

한글 음성 말하기 기능 추가



문자열 값 읽기의 적용 대상을 ta1으로 수정합니다.



한글 음성 말하기 기능 추가



음성 및 번역 그룹에 있는 TTS 실행하기 명령어를 추가합니다.

VR 명령어

- 빛 라이트 명령어
- 도구 명령어
- 음성 및 번역
 - TTS 실행하기
 - 번역하기
 - 번역후 TTS 실행
- AR 명령어

블록모드 텍스트모드

저장할 변수 이름:t

ta1 문자열 쓰기(t)

함수 void f4()

t = ta1 문자열 값 읽기()

TTS 실행하기

한글 음성 말하기 기능 추가



TTS 실행하기 명령어에 아래와 같이 컴포넌트를 추가한 후 값을 설정해 줍니다.

함수 void f4() ×

- t = ta1 문자열 값 읽기() ×
- TTS 실행하기 ^ ×
- 글자 내용:{t} ×
- 음성 언어:한국어 ×

속성
글자 내용

{t} x

{t}

음성 언어는 한국어로 설정합니다.

한글 음성 말하기 기능 추가



오른쪽 아래의 실행 버튼을 클릭합니다.



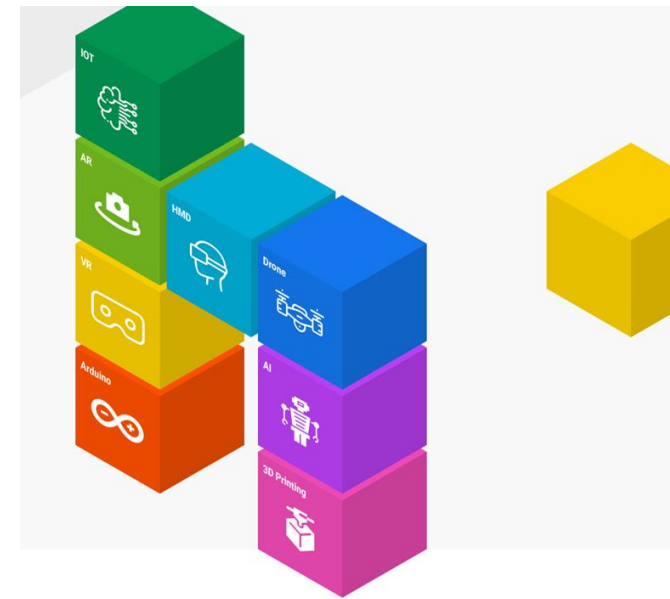
한글 음성 말하기 기능 추가



한글 문장을 입력한 후, 한글 음성으로 말하기 버튼을 클릭합니다.
잠시 후 한글 음성으로 말하기가 실행됩니다.



영어 음성 말하기 기능 추가



영어 음성 말하기 기능 추가



로직 명령어 그룹에 있는 함수를 하나 추가합니다. 함수 이름을 f5()로 수정합니다.

The screenshot shows a VR programming interface with three main panels: 'VR 명령어' (VR Commands), '블록모드' (Block Mode), and '컴포넌트' (Components). The 'VR 명령어' panel on the left has a '로직 명령어' (Logic Commands) group expanded, showing various control flow and function blocks. A red arrow points from the '함수' (Function) block in this list to a new function block '함수 void f5()' in the '블록모드' panel. The '블록모드' panel also shows other logic blocks like 'ta1 문자열 쓰기(t)', '함수 void f4()', 't = ta1 문자열 값 읽기()', and 'TTS 실행하기' with its parameters '글자 내용: {t}' and '음성 언어: 한국어'. The '컴포넌트' panel on the right shows a '속성 함수' (Property Function) block with the name 'void f5()' and a close button 'x'.

함수의 이름을 f5()로 수정합니다.

영어 음성 말하기 기능 추가



UI 명령어에서 문자열 값 읽기 명령어를 추가합니다.

The screenshot shows a VR command editor interface with two tabs: '블록모드' (Block Mode) and '텍스트모드' (Text Mode). The 'VR 명령어' (VR Command) list on the left includes 'VR 컨트롤러 명령어', 'UI 명령어', '버튼 추가하기', '레이블 추가하기', '텍스트 박스 추가하기', '텍스트 편집기 추가하기', '이미지 추가하기', '문자열 쓰기', '문자열 값 읽기', '비디오 명령어', and '사운드 명령어'. The '문자열 값 읽기' (Read String Value) command is highlighted in green. A red arrow points from this command to a new entry in the '함수 void f5()' block: '● t = tb1 문자열 값 읽기()'. Other blocks in the editor include 'ta1 문자열 쓰기(t)', '함수 void f4()', 't = ta1 문자열 값 읽기()', 'TTS 실행하기', '글자 내용: {t}', '음성 언어: 한국어', and '함수 void f5()'.

영어 음성 말하기 기능 추가



문자열 값 읽기의 적용 대상을 ta1으로 수정합니다.

함수 void f5() ✕

● t = ta1 문자열 값 읽기() ✕

속성
문자열 값 읽기

변수 이름
t

적용 대상
ta1

영어 음성 말하기 기능 추가



음성 및 번역 그룹에 있는 TTS 실행하기 명령어를 추가합니다.

The screenshot shows a VR command menu interface. On the left, under the heading 'VR 명령어', there is a list of command groups: '카메라 명령어', '빛 라이트 명령어', '도구 명령어', '음성 및 번역', and 'AR 명령어'. The '음성 및 번역' group is expanded, showing three sub-commands: 'TTS 실행하기', '번역하기', and '번역후 TTS 실행'. A red arrow points from the 'TTS 실행하기' sub-command in this list to the right-hand panel. The right-hand panel has two tabs: '블록모드' and '텍스트모드'. The '블록모드' tab is active and shows a list of blocks: 'TTS 실행하기', '글자 내용: {t}', and '음성 언어: 한국어'. The '텍스트모드' tab is also active and shows a list of blocks: '함수 void f5()', 't = ta1 문자열 값 읽기()', and 'TTS 실행하기'. A red box highlights the 'TTS 실행하기' block in the '텍스트모드' tab.

영어 음성 말하기 기능 추가



TTS 실행하기 명령어에 아래와 같이 컴포넌트를 추가한 후 값을 설정해 줍니다.

함수 void f5() ×

- t = ta1 문자열 값 읽기() ×
- TTS 실행하기 ^ ×
- 글자 내용:{t} ×
- 음성 언어:영어 ×

속성 글자 내용 {t} ×

음성 언어는 영어로 설정합니다.

영어 음성 말하기 기능 추가



- 하늘 설정하기 ^ x
 - 색상:190, 190, 190, 255 x
- 조이스틱 숨기기 x
- 텍스트 박스 추가하기 tb1 ^ x
 - 위치 비율:0.1, 0.2 x
 - 크기 비율:0.58, 0.07 x
- 버튼 추가하기 btn1 ^ x
 - 글자 내용:ChatGPT 요청하기 x
 - 위치 비율:0.7, 0.2 x
 - 크기 비율:0.2, 0.07 x
 - 클릭 함수:f1 x

- 텍스트 편집기 추가하기 ta1 ^ x
 - 위치 비율:0.1, 0.3, 0 x
 - 크기 비율:0.8, 0.5, 0 x
- 함수 void f1 () x
 - ta1 문자열 쓰기("ChatGPT에 요청중입니다...") x
 - t = tb1 문자열 값 읽기() x
 - ChatGPT 요청하기 ^ x
 - 챗 텍스트:{t} x
 - 응답 함수:f2 x
- 함수 void f2 () x
 - ta1 문자열 쓰기(value) x

영어 음성 말하기 기능 추가



버튼 추가하기 btn2 ^ x

- 글자 내용:영어로 번역하기 x
- 위치 비율:0.1, 0.85 x
- 크기 비율:0.2, 0.06 x
- 클릭 함수:f3 x

버튼 추가하기 btn3 ^ x

- 글자 내용:한글 음성으로 말하기 x
- 위치 비율:0.4, 0.85 x
- 크기 비율:0.2, 0.06 x
- 클릭 함수:f4 x

버튼 추가하기 btn4 ^ x

- 글자 내용:영어 음성으로 말하기 x
- 위치 비율:0.7, 0.85 x
- 크기 비율:0.2, 0.06 x
- 클릭 함수:f5 x

영어 음성 말하기 기능 추가



함수 void f3() ✕

t = ta1 문자열 값 읽기() ✕

번역하기 ^ ✕

글자 내용:{t} ✕

입력 언어:한국어 ✕

번역 대상 언어:영어 ✕

저장할 변수 이름:t ✕

● ta1 문자열 쓰기(t) ✕

함수 void f4() ✕

t = ta1 문자열 값 읽기() ✕

TTS 실행하기 ^ ✕

글자 내용:{t} ✕

● 음성 언어:한국어 ✕

함수 void f5() ✕

t = ta1 문자열 값 읽기() ✕

TTS 실행하기 ^ ✕

글자 내용:{t} ✕

음성 언어:영어 ✕

영어 음성 말하기 기능 추가



```
SetSky
    /Color:190, 190, 190, 255
HideJoystic
AddTextBox tb1
    /PositionRatio:0.1, 0.2
    /SizeRatio:0.58, 0.07
AddButton btn1
    /Text:ChatGPT 요청하기
    /PositionRatio:0.7, 0.2
    /SizeRatio:0.2, 0.07
    /OnClick:f1
AddTextArea ta1
    /PositionRatio:0.1, 0.3, 0
    /SizeRatio:0.8, 0.5, 0
void f1()
{
    t = tb1.GetText()
    RequestChatGPT
        /Prompt:{t}
        /OnResponse:f2
}
```

```
void f2()
{
    ta1.SetText(value)
}
AddButton btn2
    /Text:영어로 번역하기
    /PositionRatio:0.1, 0.85
    /SizeRatio:0.2, 0.06
    /OnClick:f3
AddButton btn3
    /Text:한글 음성으로 말하기
    /PositionRatio:0.4, 0.85
    /SizeRatio:0.2, 0.06
    /OnClick:f4
AddButton btn4
    /Text:영어 음성으로 말하기
    /PositionRatio:0.7, 0.85
    /SizeRatio:0.2, 0.06
    /OnClick:f5
```

```
void f3()
{
    t = ta1.GetText()
    TranslateText
        /Text:{t}
        /SourceLanguage:Korean
        /TargetLanguage:English
        /TargetVariable:t
    ta1.SetText(t)
}
void f4()
{
    t = ta1.GetText()
    PlayTTS
        /Text:{t}
        /SpeechLanguage:Korean
}
void f5()
{
    t = ta1.GetText()
    PlayTTS
        /Text:{t}
        /SpeechLanguage:English
}
```

영어 음성 말하기 기능 추가



오른쪽 아래의 실행 버튼을 클릭합니다.



영어 음성 말하기 기능 추가



영어 문장을 입력한 후, 영어 음성으로 말하기 버튼을 클릭하면 영어로 문장을 플레이 됩니다.

