

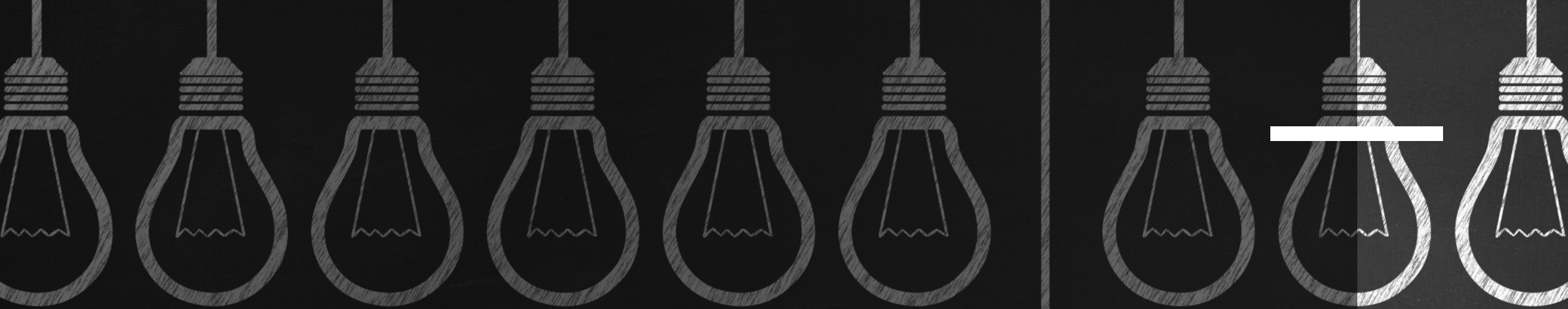
예제 8_4_1 설명

- 게임 프로그래밍
- 학번 : 2019975070
- 학과 : 소프트웨어학과
- 이름 : 한재훈



목차

- 프로그램 설명
- 헤더파일 설명
- 주요 함수 및 변수 설명
- 주요 동작 설명
- 출력 화면
- 코드를 활용하여 만들 수 있는 게임



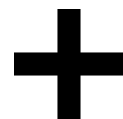
프로그램 설명

이 코드는 심리테스트 게임을 구현한 C 프로그램입니다.
사용자에게 여러 개의 질문을 제시하고, 사용자가 답변을 입력한
후에는 각 질문에 대한 해설을 출력합니다.



헤더파일 설명

- **<stdio.h>**:
 - 이 헤더 파일은 표준 입력(stdin)과 표준 출력(stdout)을 다루는 함수와 데이터 형식을 정의하고 있습니다.
 - **fopen()**, **fclose()**, **fgets()**, **printf()**, **scanf()** 등과 같은 입출력 관련 함수들을 제공합니다.
 - 코드에서 파일을 열고 읽고 쓰는 데 사용되는 **FILE** 데이터 형식도 이 헤더 파일에 정의되어 있습니다.
- **<conio.h>**:
 - 이 헤더 파일은 콘솔 창에서 사용자와 상호작용하고 화면을 조작하는 함수들을 제공합니다.
 - **getch()**, **clrscr()** 등과 같은 함수들은 사용자로부터 키 입력을 받고 콘솔 화면을 지우는 데 사용됩니다.
 - 주로 Windows 환경에서 사용되며, 이식성이 낮아 다른 운영체제에서는 지원되지 않을 수 있습니다. 모던한 C 프로그래밍에서는 이 헤더 파일 대신에 표준 라이브러리를 사용하는 것이 권장됩니다.
- **<windows.h>**:
 - **<windows.h>**는 Windows 운영체제에서 사용되는 헤더 파일로, Windows API(응용 프로그램 프로그래밍 인터페이스)를 사용할 때 필요합니다.
 - Windows API는 Windows 운영체제에서 윈도우 창 및 그래픽 인터페이스, 파일 및 디렉토리 조작, 프로세스 관리 등 다양한 시스템 레벨의 작업을 수행할 수 있도록 해줍니다.
 - **SetConsoleCursorPosition()** 함수와 같은 콘솔 창 조작 함수를 이 헤더 파일에서 사용하고 있습니다.
 - Windows 운영체제에서 콘솔 애플리케이션을 개발할 때 필요한 헤더 파일입니다.



주요 함수 및 변수 설명

- gotoxy(int x, int y): 콘솔 화면에서 커서의 위치를 (x, y)로 이동시키는 함수입니다. Windows API SetConsoleCursorPosition 함수를 사용하여 구현되었습니다.
- display_text(int count): 문제 번호(count)를 받아와서 해당 문제의 텍스트를 출력하는 함수입니다.

```
1  #include <stdio.h>
2  #include <conio.h>
3  #include <windows.h>
4  void gotoxy(int x, int y);
5  void display_text(int count);
6  void display_question(void);
7  void display_result(void);
8  void intro_game(void);
9  char sol[10][11]; //사용자의 입력 값을 저장
10 int main(void)
11 {
12     intro_game();
13     display_question();
14     printf("\n문 제 풀 이 가 모 두 끝 났 습 니 다 .\n");
15     printf("\n\n풀 이 를 보 려 면 아 무 키 나 누 르.");
16     getch();
17     display_result();
18     return 0;
19 }
20
21 void gotoxy(int x, int y)
22 {
23     COORD Pos = {x - 1, y - 1};
24     SetConsoleCursorPosition(GetStdHandle(STDOUT_HANDLE), Pos);
25 }
26 void display_text(int count)
27 {
28     gotoxy(1, 2);
29     printf("다 음 문 제 를 외 고 답 을 입 력 하 시 오");
30     printf("(%d번 문 제 ) \n\n", count+1);
31 }
32 }
```

주요 함수 및 변수 설명

- display_question(void): 심리테스트 질문 파일("심리테스트질문.txt")을 읽어와서 각 문제를 출력하고, 사용자로부터 답변을 입력받습니다. 입력된 답변은 sol 배열에 저장됩니다. 이때 텍스트 파일은 코드와 같은 디렉토리에 위치하여야 합니다.

```
33 void display_question(void)
34 {
35     FILE *fp1;
36     char chr1[82];
37     int line=5, count=0;
38     char *file1="심리테스트질문.txt";
39     fp1=fopen(file1, "r");
40     system("cls");
41     while(!feof(fp1))
42     {
43         fgets(chr1, 80, fp1);
44         if (chr1[0]==10)
45         {
46             display_text(count);
47             gotoxy(1, line);
48             scanf("%s", sol[count]);
49             system("cls");
50             line=5;
51             count++;
52         }
53         else
54         {
55             gotoxy(1, line++);
56             printf("%s", chr1);
57         }
58     }
59     display_text(count);
60     gotoxy(1, line);
61     scanf("%s", sol[count]);
62     fclose(fp1);
63 }
64 }
```



display_question(void) 상세 설명

- FILE *fp1;: 파일 포인터 fp1을 선언합니다. 이 포인터는 파일 "심리테스트질문.txt"를 열고 파일 내용을 읽기 위해 사용됩니다.
- char chr1[82];: 문자열을 저장할 배열 chr1을 선언합니다. 이 배열은 한 줄에 최대 80자의 문자열과 개행 문자('\n') 등을 저장하는 데 사용됩니다.
- int line=5, count=0;: 변수 line과 count를 선언하고 초기값을 지정합니다. line은 화면에 출력할 텍스트의 행 번호를 나타내고, count는 현재까지 읽은 질문의 수를 나타냅니다.
- char *file1="심리테스트질문.txt";: 질문이 저장된 파일의 이름을 지정합니다.
- fp1=fopen(file1, "r");: 지정한 파일을 읽기 모드("r")로 엽니다.
- system("cls");: 콘솔 화면을 지우고 초기화합니다.
- while(!feof(fp1)): 파일의 끝(end-of-file)에 도달할 때까지 반복합니다.
- fgets(chr1, 80, fp1);: 파일에서 한 줄(최대 80자)을 읽어와 chr1 배열에 저장합니다.
- if (chr1[0]==10) { ... }: 읽은 줄이 비어있는 줄(개행 문자로만 이루어진 줄)인지 확인합니다. 이것은 새로운 질문을 시작하는 시점을 나타냅니다.



display_question(void) 상세 설명

- `display_text(count);`: `display_text` 함수를 호출하여 현재 질문 번호를 출력합니다.
- `gotoxy(1, line);`: 커서의 위치를 지정하여 질문을 화면에 출력합니다.
- `scanf("%s", sol[count]);`: 사용자로부터 답변을 입력받아 `sol` 배열에 저장합니다. `sol[count]`는 현재 질문에 대한 사용자의 답변을 저장하는 배열입니다.
- `system("cls");`: 다음 질문을 표시하기 전에 콘솔 화면을 지우고 초기화합니다.
- `line=5;`: 다음 질문을 출력할 행 번호를 초기화합니다.
- `count++;`: 질문 번호를 증가시킵니다.
- `else { ... }`: 읽은 줄이 질문의 일부인 경우(비어 있지 않은 경우), 해당 질문을 출력합니다.
- `display_text(count);`: 마지막 질문의 답변을 입력받은 후에도 마지막 질문 번호를 출력합니다.
- `gotoxy(1, line);`: 마지막 질문의 답변 입력 위치로 이동합니다.
- `scanf("%s", sol[count]);`: 마지막 질문의 답변을 입력받습니다.
- `fclose(fp1);`: 파일을 닫습니다.
- 이렇게 `display_question` 함수는 파일에서 질문을 읽어오고, 사용자로부터 답변을 입력받아 배열에 저장하며, 화면에 질문을 표시하는 역할을 합니다.

주요 함수 및 변수 설명

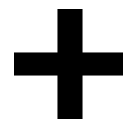
- display_result(void): 심리테스트 해설 파일("심리테스트해설.txt")을 읽어와서 각 문제의 해설과 사용자의 답변을 출력합니다.

```
65 void display_result(void)
66 {
67     FILE *fp2;
68     char chr1[82];
69     int line=5, count=0;
70     char *file2="심 리 테스트 해 설 .txt";
71     fp2=fopen(file2, "r");
72     system("cls");
73     while(!feof(fp2))
74     {
75         fgets(chr1, 80, fp2);
76         if (chr1[0]==10)
77         {
78             gotoxy(1, line+1);
79             printf("당 신 의 답 은 : %s\n", s);
80             printf("다 음 문 제 의 해 설 은 아 무");
81             getch();
82             system("cls");
83             line=5;
84             count++;
85         }
86         else
87         {
88             gotoxy(1, line++);
89             printf("%s", chr1);
90         }
91     }
92     gotoxy(1, line+1);
93     printf("당 신 의 답 은 : %s\n", sol[count]);
94     fclose(fp2);
95 }
96 }
```



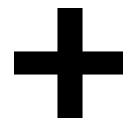
display_result(void) 상세 설명

- `FILE *fp2;`: 파일 포인터 `fp2`를 선언합니다. 이 포인터는 "심리테스트해설.txt" 파일을 열고 파일 내용을 읽기 위해 사용됩니다.
- `char chr1[82];`: 문자열을 저장할 배열 `chr1`을 선언합니다. 이 배열은 한 줄에 최대 80자의 문자열과 개행 문자('\n') 등을 저장하는 데 사용됩니다.
- `int line=5, count=0;`: 변수 `line`과 `count`를 선언하고 초기값을 지정합니다. `line`은 화면에 출력할 텍스트의 행 번호를 나타내고, `count`는 현재까지 읽은 해설의 수를 나타냅니다.
- `char *file2="심리테스트해설.txt";`: 해설이 저장된 파일의 이름을 지정합니다.
- `fp2=fopen(file2, "r");`: 지정한 파일을 읽기 모드("r")로 엽니다.
- `system("cls");`: 콘솔 화면을 지우고 초기화합니다.
- `while(!feof(fp2))`: 파일의 끝(end-of-file)에 도달할 때까지 반복합니다.
- `fgets(chr1, 80, fp2);`: 파일에서 한 줄(최대 80자)을 읽어와 `chr1` 배열에 저장합니다.
- `if (chr1[0]==10) { ... }`: 읽은 줄이 비어있는 줄(개행 문자로만 이루어진 줄)인지 확인합니다. 이것은 새로운 해설을 시작하는 시점을 나타냅니다.



display_result(void) 상세 설명

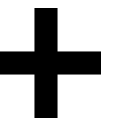
- gotoxy(1, line+1);: 커서의 위치를 지정하여 사용자의 답변과 해당 질문의 해설을 출력합니다.
- printf("당신의 답은 : %s\n", sol[count]);: 사용자의 답변을 출력합니다. sol[count] 배열에는 display_question 함수에서 저장한 사용자의 답변이 있습니다.
- printf("다음문제의 해설은 아무키나 누르시오.");: 다음 문제의 해설을 보기 위해 아무 키나 누를 것을 안내합니다.
- getch();: 사용자의 아무 키 입력을 기다립니다.
- system("cls");: 다음 해설을 표시하기 전에 콘솔 화면을 지우고 초기화합니다.
- line=5;: 다음 해설을 출력할 행 번호를 초기화합니다.
- count++;: 해설 번호를 증가시킵니다.
- else { ... }: 읽은 줄이 해설의 일부인 경우(비어 있지 않은 경우), 해당 해설을 출력합니다.
- fclose(fp2);: 파일을 닫습니다.



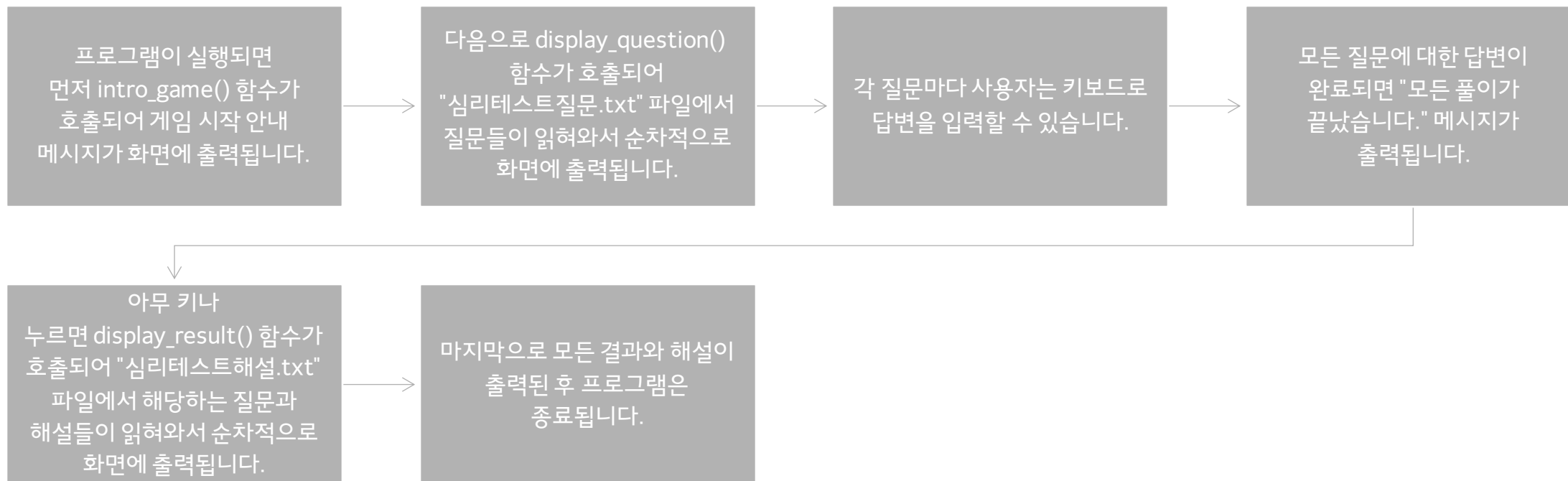
주요 함수 및 변수 설명

- intro_game(void): 게임 시작 전에 화면에 간단한 안내 메시지를 출력하는 함수입니다.
- sol[10][11]: 사용자의 입력 값을 저장하기 위한 2차원 문자열 배열입니다. 최대 10개의 문제에 대한 답변을 저장할 수 있으며, 각 답변은 최대 10글자까지 저장할 수 있습니다.

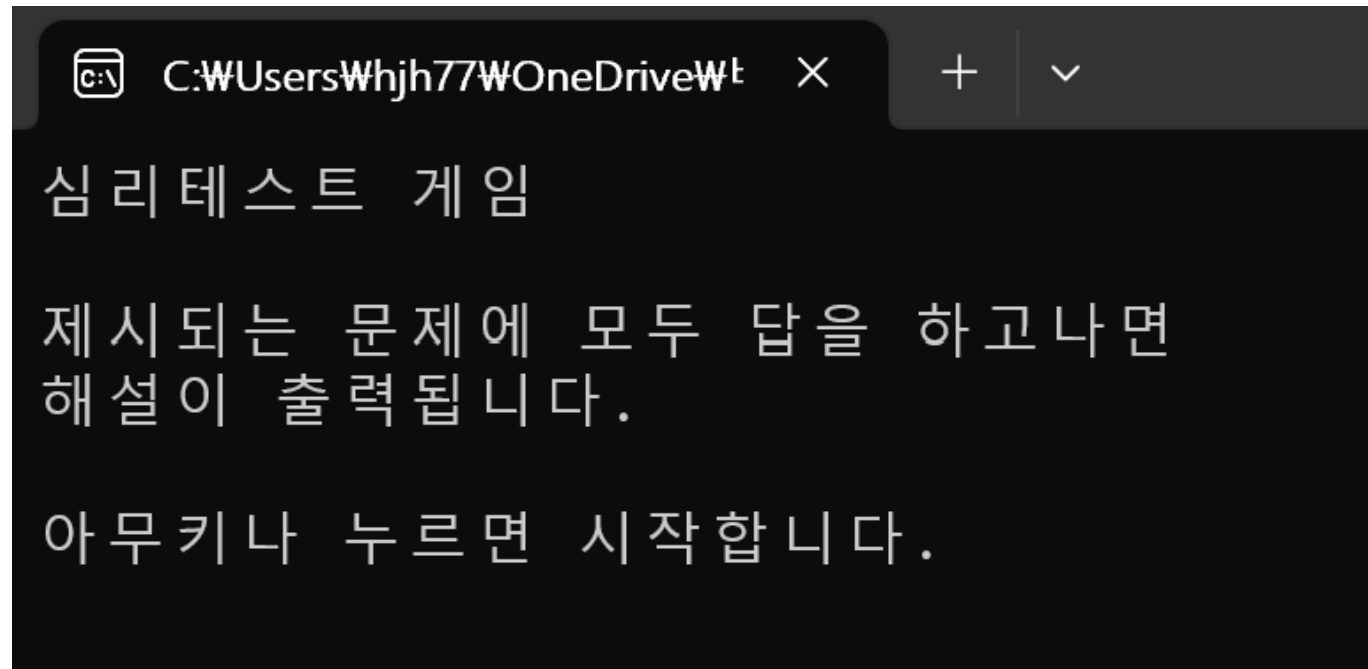
```
97 void intro_game(void)
98 {
99     system("cls");
100     printf("심 리 테스트 게 임 \n\n");
101     printf("제 시 되는 문 제 에 모 두 답 을 하 고 나 면 \n");
102     printf("해 설 이 출 력 됩 니 다 .\n\n");
103     printf("아 루 키 나 누 르 면 시 작 합 니 다 . ");
104     getch();
105 }
106
```



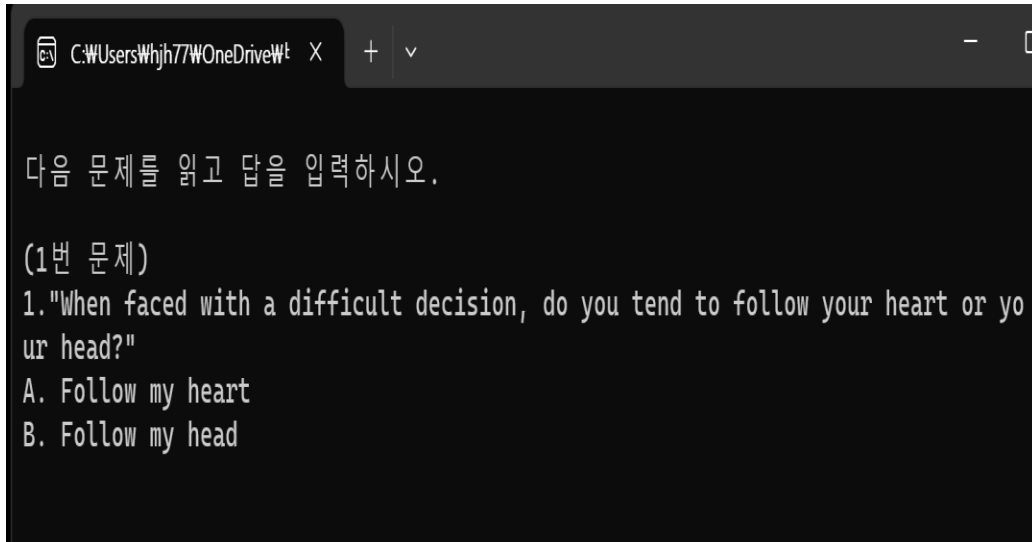
주요 동작 설명



출력화면 -게임소개-



출력화면 -질문-



A screenshot of a web browser window with a dark theme. The address bar shows a local file path. The main content area displays a question in Korean and English, followed by two multiple-choice options.

```
C:\Users\Whjh77\OneDrive\... X + v
```

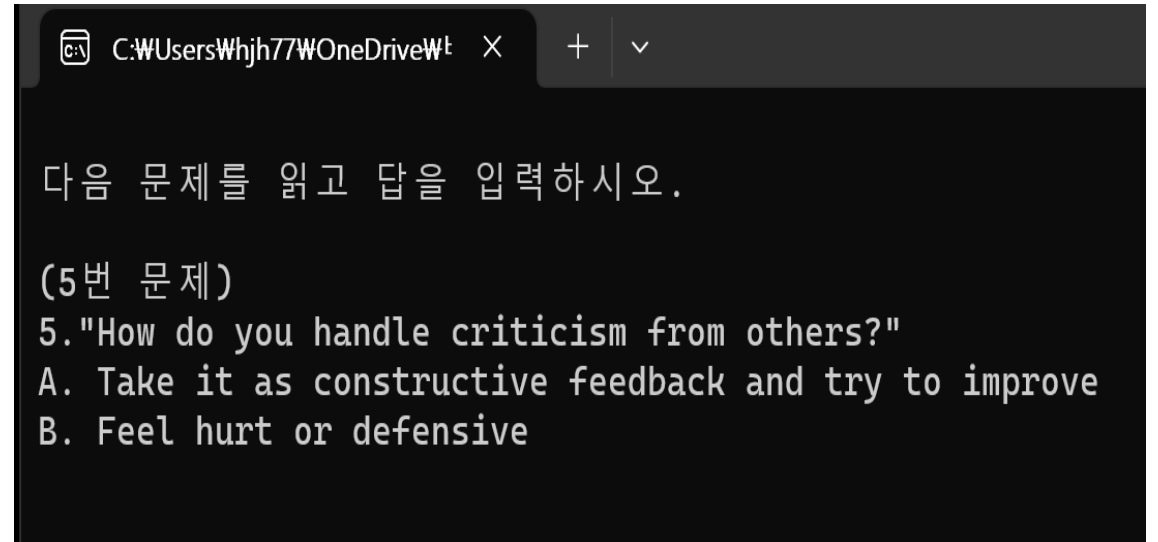
다음 문제를 읽고 답을 입력하십시오.

(1번 문제)

1. "When faced with a difficult decision, do you tend to follow your heart or your head?"

A. Follow my heart

B. Follow my head



A screenshot of a web browser window with a dark theme. The address bar shows a local file path. The main content area displays a question in Korean and English, followed by two multiple-choice options.

```
C:\Users\Whjh77\OneDrive\... X + v
```

다음 문제를 읽고 답을 입력하십시오.

(5번 문제)

5. "How do you handle criticism from others?"

A. Take it as constructive feedback and try to improve

B. Feel hurt or defensive



출력화면 -해설-

1."When faced with a difficult decision, do you tend to follow your heart or your head?"

A. Follow my heart: 媛맏절???겍썩媛??冤썩뽕???댕뿔?댕뿔.

B. Follow my head: ?심-?곡썩 ?맏뿔???겍썩?썩빅?댕뿔.

당신의 답은 : A

다음문제의 해설은 아무키나 누르시오.

3."In social situations, do you typically prefer to be the center of attention or stay in the background?"

A. Center of attention: 愿?뽕 以뿔뿔???댕린瑜??썩상?⁽⁵⁾뿔??

B. Stay in the background: 諛겍겍??癒뿔b媛겍? ?썩상?⁽⁵⁾뿔??

당신의 답은 : A

다음문제의 해설은 아무키나 누르시오.



• 코드를 활용하여 만들 수 있는 게임

- **퀴즈 게임:** "심리테스트질문.txt" 파일에는 각각의 퀴즈 질문이, "심리테스트해설.txt" 파일에는 각각의 퀴즈 답변이 들어가도록 하여 퀴즈 게임으로 변형할 수 있습니다.
- **대화형 소설:** 사용자가 주어진 선택지 중 하나를 선택하면 그에 따라 다음 스토리가 진행되는 대화형 소설 또는 어드벤처 게임으로 만들 수도 있습니다.
- **역사나 과학 등의 교육용 프로그램:** 각각의 문제와 그에 대한 해설로서 역사적 사건, 과학적 사실 등을 제시하여 교육용 프로그램으로 활용할 수도 있습니다.
- **롤플레이팅 게임(RPG):** 위 코드를 확장하여 캐릭터 생성, 능력치 설정 등 롤플레이팅 요소를 추가하거나 전투 시스템, 아이템 시스템 등을 도입할 수도 있습니다.
- **탐색/탈출 게임:** 방이나 건물 같은 장소를 탐색하며 힌트를 찾아내고 문제를 해결하는 타입의 탐색/탈출 게임으로 만들 수도 있습니다.



출처

- <https://blog.naver.com/skout123/50132263755>
- https://ko.wikipedia.org/wiki/%ED%97%A4%EB%8D%94_%ED%8C%8C%EC%9D%BC



감사합니다

