

배이는 너냐!!

D : 황유빈 , P : 김호현 , G 문정민

목차 (기획 파트)

1. 게임 개요
2. 기획 의도
3. 게임 설정



1. 게임 개요

| TRPG 기반 추리게임 ‘범인은 너냐!’ | |
|------------------------|---|
| 제목 | ‘범인은 너냐!’ |
| 장르 | TRPG, 추리 |
| 플랫폼 | PC |
| 개발 엔진 | Unity |
| 타겟 유저 | 다양한 상황을 예상하고 경험하는 것을 즐기는 유저 한 번의 추리가 끝나면 게임이 끝나는 추리게임에 질린 유저 |

2. 기획 의도

Q1. TRPG를 혼자서 즐길 수는 없을까?

A1. ‘포 더 킹’ 과 같이 탐험.전투 기반 TRPG 게임은 많지만,
조사.추리 기반 TRPG 게임은 거의 없어서 만들어 보자고 생각함.

Q2. 추리 게임에서 한 사건에 범인은 한정되어야 하는가?

A2. 꼭 한 사건에 범인이 한 사람(혹은 한 팀)으로만 정해져 있을 필요가
없다고 생각해서 한 사건에서도 범인이 다양할 수 있게 설정함.

Q3. 어째서 ‘TRPG’ 와 ‘추리’ 라는 두 장르를 결합하였는가?

A3. 지금까지 추리 게임들은 대부분 정보 양식 등이 정형화가 되어 있는데,
어떻게 하면 특색을 줄 수 있을까 고민을 하다 보니
‘TRPG’ 의 방식을 통해 정보의 다양한 모습을 보여줄 수 있다고 생각함.



3. 게임 설정.

메인 설정

당신은 크고 작은 사건을 해결
해온 '탐정' 입니다.

어쨌든 당신의 곁에서는 사건사고가
끊이질 않는군요.

그런 사건사고를 계속 해결하는
당신은 어떤 '탐정' 이 되실 건가요?

시나리오

익숙지 못한 타지에서서의 의뢰를 마친 당신은 휴양을 겸해 돌아가는 길에
호화 크루즈를 타고 돌아가고 있었습니다.

이 호화 크루즈는 몇 십년 동안 호평일색의 후기로 유명한 크루즈이며,
시설과 식사, 서비스도 나무랄 데 없이 훌륭하지만 이 크루즈가 유명해진
이유는 정기적으로 진행되는 선상 오페라입니다.

예술에는 그다지 박식하지 않던 당신 또한 어젯밤에 상연했던 오페라에
상당히 감동을 받았던 것 같습니다.

그렇게 오페라를 너무 즐긴 나머지 당신이 일어난 시간은 11시를 조금
넘은 시간이네요.

신이 늦은 아침을 때우기 위해 방을 나왔을 때, 크루즈 내부를 뒤덮는 큰
비명이 들려왔습니다.

비명이 들리자, 상상치 않은 일이 일어났음을 직감한 당신은 비명이 들린
곳으로 빠르게 달려갑니다.



목차 (프로그래밍 파트)

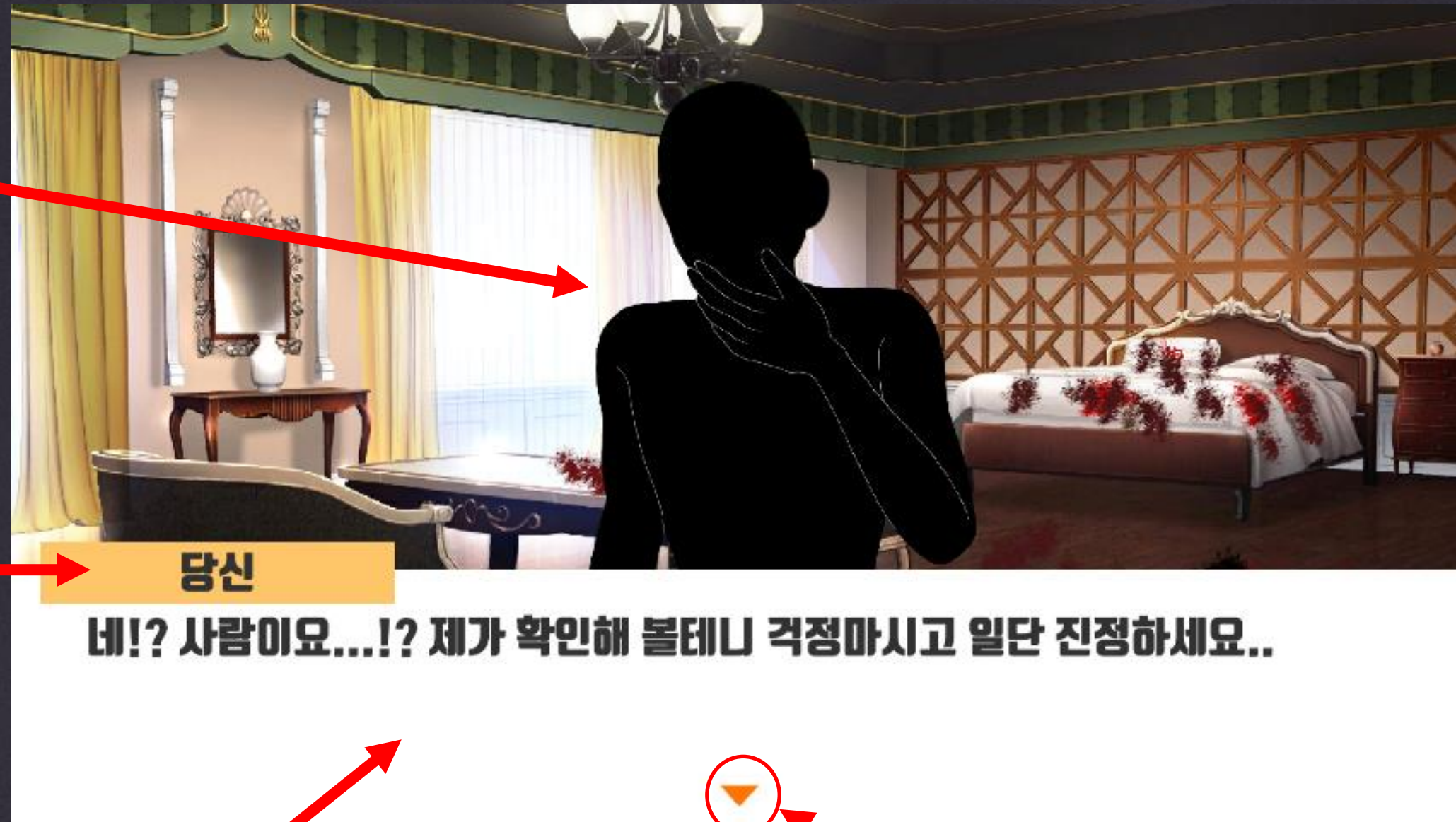
- 01. 대화창 제작
- 02. 선택지 분기 시스템
- 03. 아이템 리스트 제작
- 04. 능력치 시스템
- 05. 다중 스토리 진행 시스템
- 06. 플레이 영상



01. 대화창 제작 - 형태

일러스트

이름 표시창



대화 표시창

대화 넘기기

01. 대화창 제작 - 스크립트

```
IEnumerator Typing(Text typingText, float speed)
{
    for (int i = 0; i < ContentList[j].Length; i++)
    {
        if(typingsfx != null && i % 2 < 0.02f && View.activeSelf == true)
        {
            typingsfx.Play();
        }
        typingText.text = ContentList[j].Substring(0, i + 1);
        yield return new WaitForSeconds(speed);
    }
    Button.SetActive(true);
}
```

코루틴 함수를 통해
텍스트가 일정 속도로 출력되게 함.
효과음 또한 추가.

| Content List | | 67 |
|--------------|---|----|
| Element 0 | 으아아아아악 | |
| Element 1 | (어디선가 비명 소리가 들린다.) | |
| Element 2 | (당신은 공장 비명 소리가 들린 곳으로 가보았다.) | |
| Element 3 | (그곳엔 청소 직원이 살짝 열린 문 앞에 주저 앉아 떨고 있다.) | |
| Element 4 | 선택지 | |
| Element 5 | A | |
| Element 6 | 무슨 일입니까? | |
| Element 7 | 사...사람이...죽어 있었어요....! | |
| Element 8 | 네!? 사람이요...!? 제가 확인해 볼테니 걱정마시고 일단 진정하세요.. | |
| Element 9 | 감사합니다.... | |
| Element 10 | (당신은 방문을 벌컥 열었다.) | |
| Element 11 | B | |
| Element 12 | (활짝 열린 문 너머 방안의 풍경은 당신이 지금까지 경험했던) | |
| Element 13 | (익숙하며 더 끔찍할 수도 있는 사건 현장이 펼쳐져 있었다.) | |
| Element 14 | (충격에 잠시 넋을 잃었다.) | |
| Element 15 | 끔찍하군. 또 살인 사건인가...? | |
| Element 16 | 직원 분이시죠...? 일단 어서 선장에게 이 사실을 알려세요. | |
| Element 17 | 아...예..! 알겠습니다. | |
| Element 18 | (객실을 둘러봐야겠군) | |
| Element 19 | *힌트 : 조사 전 메뉴에서 능력치를 올려보세요. | |
| Element 20 | | |
| Element 21 | (이 객실은 크루즈 내에도 몇 없는 스위트룸 객실로) | |
| Element 22 | (쾌적한 크기의 공간과 함께 창문 넘어 보이는 푸른 바다가 인상적인 객실C | |
| Element 23 | (하지만, 이 곳은 사건 현장... 넓은 방 한 가운데에 널브러져 있는 시체와 피 | |
| Element 24 | (잠시 시체의 얼굴을 보니, 당신은 이 사람을 어디선가 본 적이 있다는 생각 | |
| Element 25 | 작까만...! 이 사람은 그때 오페라의 여주인공 배우 잭아..? | |

```
public string[] name_ = new string[1];
public string[] ContentList = new string[1];

public Sprite[] illust;
public Image illustView;
```

배열로 대화창의 모든 대화
관리해 유니티 내에서 편집 가능

02. 선택지 분기 시스템 - 형태



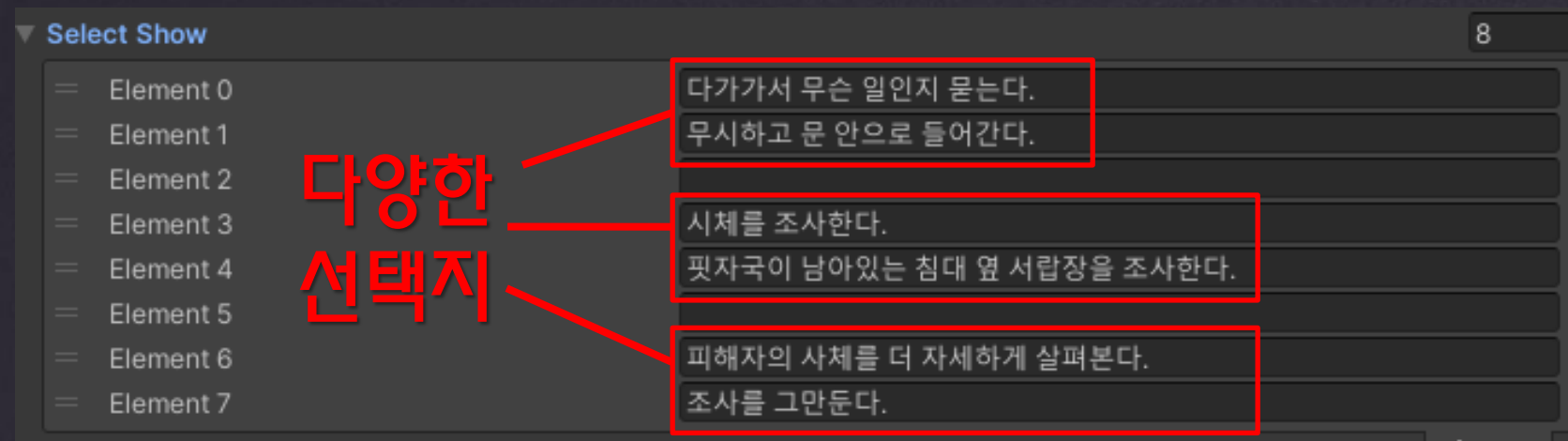
시체를 조사한다.

핏자국이 남아있는 침대 옆 서랍장을 조사한다.

선택지의 수만큼
버튼 화면에 추가

선택지에 따라 다른 대사와 이벤트 출력

02. 선택지 분기 시스템 - 스크립트



게임 내 여러 선택지를 미로 만들어 놓고,
필요한 선택지 만을 불러오는 것을 구현함.

```
public void SelectButton(int choice)
{
    switch (choice)
    {
        case 1 :
            if (name_.Length == ContentList.Length && illust.Length == ContentList.Length)
            {
                SelectDirect.SetActive(false);
                for (; ContentList[j] != "A"; j++) ;
                TextNext();
                i = j;
                k = j;
                View.SetActive(true);
                if (repeat[0] == 0)
                {
                    repeat[0] = 1;
                    repeat[1] = 1; // 청소부 상호작용
                    SAVEDATA.cleaner = 1;
                    gameObject.transform.parent.gameObject.GetComponent<Animation>().Play("Chapter1BackgroundUp");
                }
            }

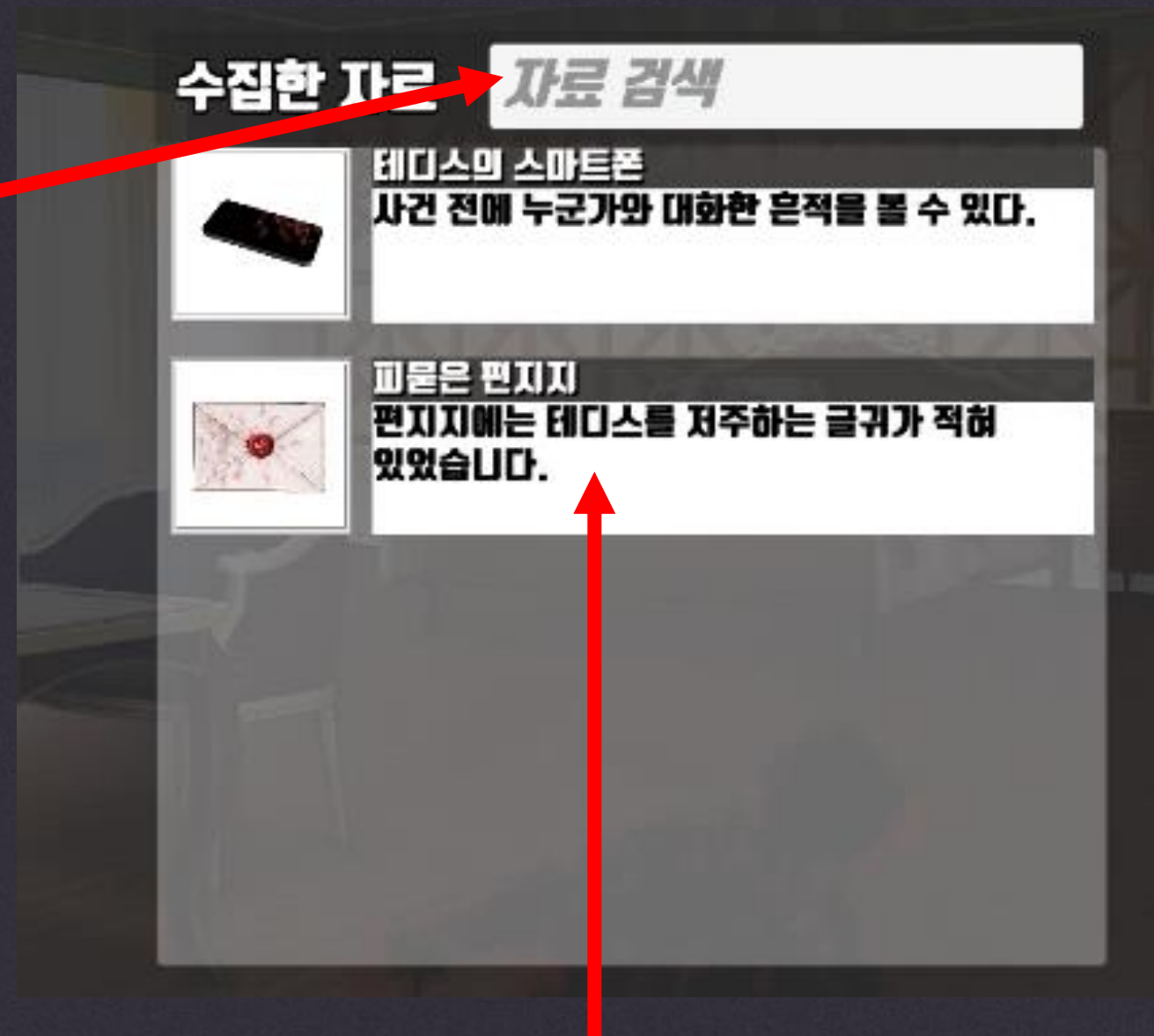
            break;

        case 2 :
            if (name_.Length == ContentList.Length && illust.Length == ContentList.Length)
            {
                SelectDirect.SetActive(false);
                for (; ContentList[j] != "B"; j++) ;
                TextNext();
                i = j;
                k = j;
                View.SetActive(true);
                if (repeat[0] == 0)
```

여러 선택지를 번호를 받아와 스위치
문의 여러 케이스로 작동되는 원리.

03. 아이템 리스트 제작 - 형태

입력 값으로
아이템 리스트
불러오기



많은 아이템들을 스크롤하며
볼 수 있음.

03. 아이템 리스트 제작 - 스크립트



하나의 오브젝트에 자식으로 여러 개의 오브젝트를 넣어 일정 간격으로 화면에 표시.

```
if (SearchText.text != "")
{
    for (; number < Operand.childCount; number++)
    {
        if (Operand.GetChild(number).name.Contains(SearchText.text.ToString()) == false)
        {
            Operand.GetChild(number).gameObject.SetActive(false);
        }
        else
        {
            Operand.GetChild(number).gameObject.SetActive(true);
        }
    }
    number = 0;
}
else
{
    number = 0;
    for (; number < Operand.childCount; number++)
    {
        Operand.GetChild(number).gameObject.SetActive(true);
    }
    number = 0;
}
```

입력 값을 받아 자식 오브젝트 명을 검색하고, 검색 값과 다른 오브젝트 명을 가진 오브젝트를 비활성화.

04. 능력치 시스템 - 형태



추가 기능치 값으로 능력을 업그레이드

04. 능력치 시스템 - 스크립트

| ▼ Action Number 67 | |
|--------------------|---|
| Element 0 | 0 |
| Element 1 | 0 |
| Element 2 | 0 |
| Element 3 | 0 |
| Element 4 | 0 |
| Element 5 | 0 |
| Element 6 | 0 |

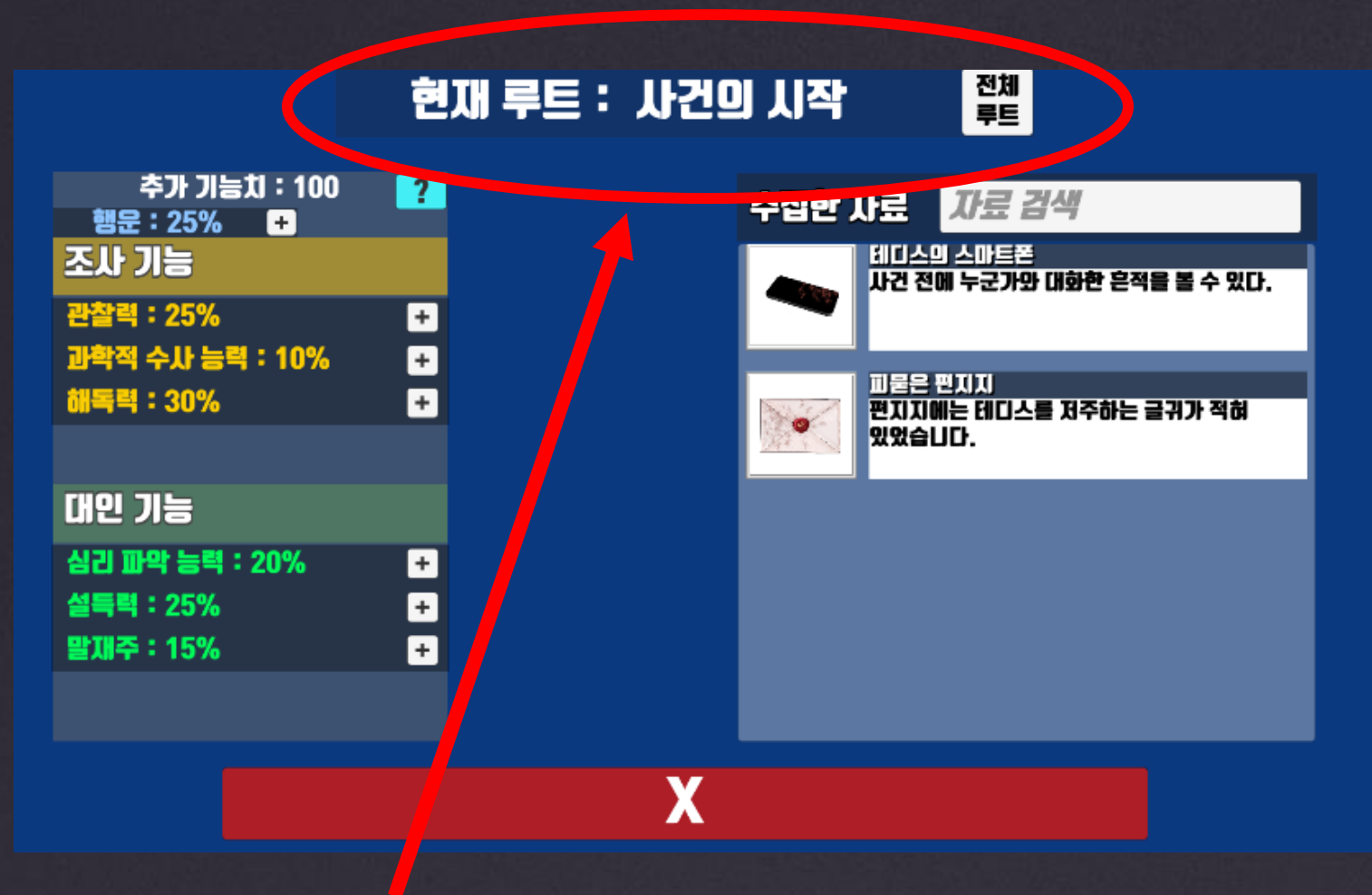
배열을 이용해
대화창의 배열 인덱스 위치와 연동하여
일정 대사 구간에서 능력치 값에 따라 다른 대사가
나오고 게임이 다르게 진행되도록 함.

```
else if (actionNumber[i] == 4) // 행운 판정
{
    Action = false;
    if (Random.Range(0, 100) <= SAVEDATA.stetLuck)
    {
        repeat[2] = 1;
        for (; ContentList[j] != "행운 판정 성공"; j++) ;
        j++;
    }
    else
    {
        repeat[2] = 2;
        for (; ContentList[j] != "행운 판정 실패"; j++) ;
        j++;
    }
}
else if (actionNumber[i] == 5) // 관찰력 판정
{
    Action = false;
    if (Random.Range(0, 100) <= SAVEDATA.stetObserve)
    {
        Debug.Log("관찰력");
        repeat[2] = 1;
        for (; ContentList[j] != "관찰력 판정 성공"; j++) ;
        j++;
    }
    else
    {
        repeat[2] = 2;
        for (; ContentList[j] != "관찰력 판정 실패"; j++) ;
        j++;
    }
}

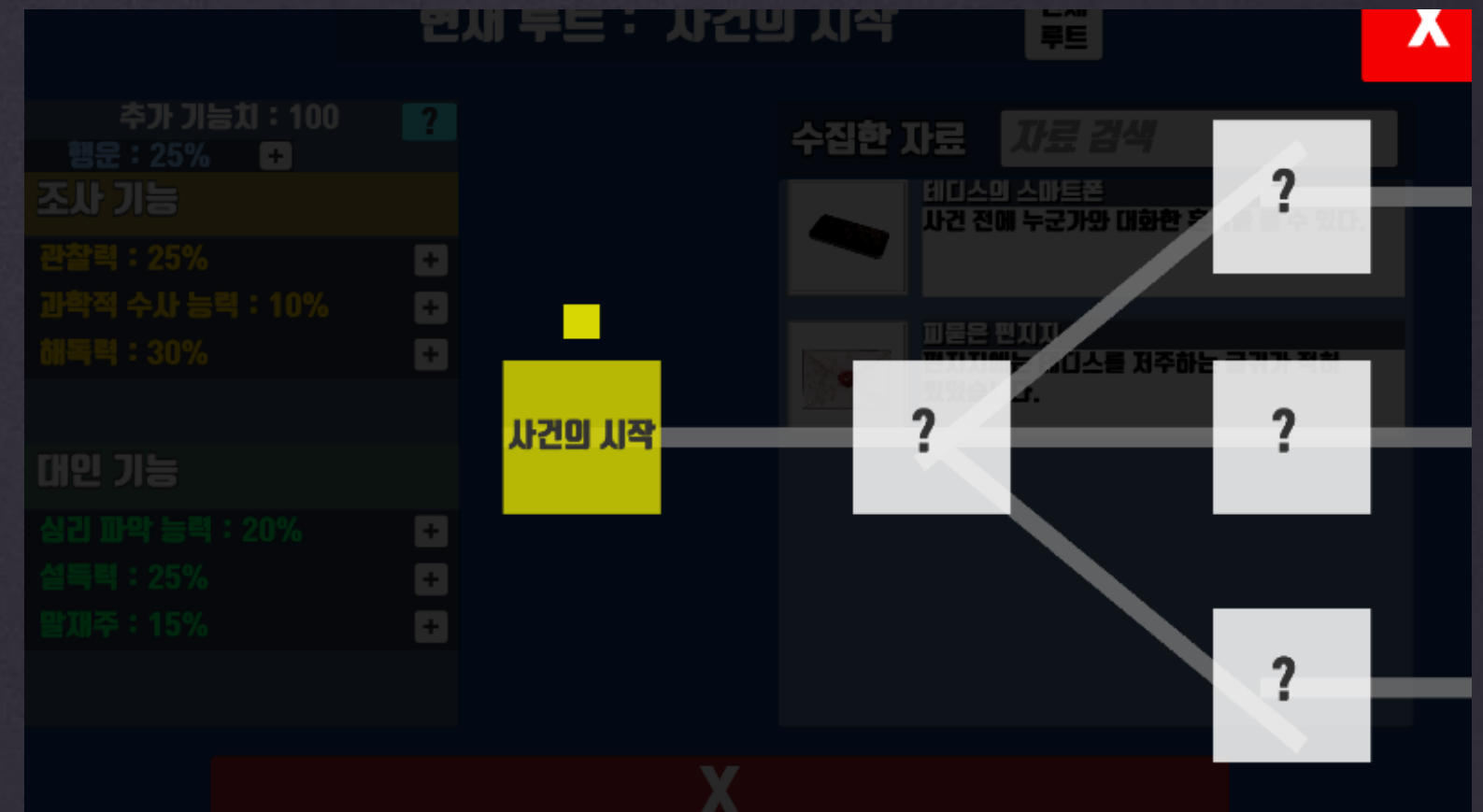
i = j;
k = j;
}
```

상황마다 사용되는
능력치 값을 마음대로
설정가능.

05. 다중 스토리 진행



현재 진행 중인 루트 표시



게임 진행 중 행동에 따라
누적된 값을 스토리 진행 루트가
결정된다.

06. 플레이 영상

<https://youtu.be/PXsiaEpGL9k>

감사합니다