

# Unity Backend

“길드” 기능을 이용한 길드 정보 조회

Created in 2023-07-30  
Last Updated 2023-08-02  
Unity Version 2022.2.2f1

## *Index*

- ◆ 내 길드 정보 조회
- ◆ 특정 길드 정보 조회
- ◆ 랜덤 길드 정보 조회

# 내 길드 정보 조회

- 내 길드 정보 조회
- 길드 메타 데이터를 이용한 공지사항 등록
- 길드 인원수 출력
- 모든 길드원 정보 조회



# 내 길드 정보 조회

## ■ 내 길드 정보 조회

### ■ 내 길드 정보 조회 메소드

`GetMyGuildInfoV3();`

현재 유저가 속한 길드의 정보 조회

길드에 속해 있지 않은 경우 `statusCode 412, errorCode PreconditionFailed` 반환

#### Error Cases

\* Old guild의 유저가 조회한 경우 (412 / guild's version is different ..)

\* guild가 없는 유저가 조회한 경우 (412 / notGuildMember ..)





# 내 길드 정보 조회

## ■ 내 길드 정보 JsonData

```
{
  "guild":
  {
    "memberCount": "22" // 길드원 수
    "notice": "공지사항입니다." // 길드 메타 데이터
    "viceMasterList": // 부 길드 마스터 목록
    {
      "inDate": "2023-06-26T02:22:22.022Z" // 부 길드 마스터 1 inDate
      "nickname": "고박사" // 부 길드 마스터 1 닉네임
    },
    {
      "inDate": "2023-06-26T02:22:22.022Z" // 부 길드 마스터 2 inDate
      "nickname": "이름없음" // 부 길드 마스터 2 닉네임
    }
    "masterNickname": "DoctorKo" // 길드 마스터 닉네임
    "inDate": "2023-06-26T02:22:22.022Z" // 길드 inDate
    "guildName": "유니티노트" // 길드 이름
    "goodsCount": "1" // 길드 굿즈 개수
    "masterInDate": "2023-06-26T02:22:22.022Z" // 길드 마스터 inDate
    "_immediateRegistration": "true" // 즉시 가입 여부. SetRegistrationValueV3를 통해 설정한 경우에만 존재
    "_countryCode": "KR" // 국가 코드 (국가 코드를 추가한 경우에만 존재함)
  }
}
```



# 내 길드 정보 조회

- 길드 데이터를 관리하는 스크립트 생성 및 작성
  - C# Script 생성 후 스크립트의 이름을 "GuildData"로 변경

```
1 using System.Collections.Generic;
2
3 public class GuildData
4 {
5     public string guildName; // 길드 이름
6     public string guildInDate; // 길드 InDate
7     public int memberCount; // 길드 회원 수
8     public GuildMemberData master; // 길드 마스터
9     public List<GuildMemberData> viceMasterList; // 부길드 마스터 목록
10 }
```



# 내 길드 정보 조회

- GetMyGuildInfoV3() 메소드를 호출해 길드 정보 불러오기
  - BackendGuildSystem Script 수정

```
1  +using ...
5
6  -public class BackendGuildSystem : MonoBehaviour
7  {
8      [SerializeField]
9      private FadeEffect_TMP      textLog;
10     [SerializeField]
11     private GuildCreatePage      guildCreatePage;
12     [SerializeField]
13     private GuildApplicantsPage  guildApplicantsPage;
14
15     public  GuildData              myGuildData { private set; get; } = new GuildData();
16
17     + public void CreateGuild(string guildName, int goodsCount=1)...
34
35     + public void ApplyGuild(string guildName)...
53
54     + public void GetApplicants()...
130
131    + public void ApproveApplicant(string gamerInDate)...
145
146    + public void RejectApplicant(string gamerInDate)...
162
```



# 내 길드 정보 조회

- BackendGuildSystem Script 수정 (계속)

```
163 public void GetMyGuildInfo()  
164 {  
165     Backend.Guild.GetMyGuildInfoV3(callback =>  
166     {  
167         if ( !callback.IsSuccess() )  
168         {  
169             ErrorLog(callback.GetMessage(), "Guild_Failed_Log", "GetMyGuildInfoV3");  
170             return;  
171         }  
172  
173         try  
174         {  
175             LitJson.JsonData guildJson = callback.GetFlattenJSON()["guild"];  
176  
177             if ( guildJson.Count <= 0 )  
178             {  
179                 Debug.LogWarning("불러온 길드 데이터가 없습니다.");  
180                 return;  
181             }  
182
```



# 내 길드 정보 조회

## BackendGuildSystem Script 수정 (계속)

```
183         myGuildData.guildName      = guildJson["guildName"].ToString();
184         myGuildData.guildInDate     = guildJson["inDate"].ToString();
185         myGuildData.memberCount     = int.Parse(guildJson["memberCount"].ToString());
186
187         myGuildData.master           = new GuildMemberData();
188         myGuildData.master.nickname  = guildJson["masterNickname"].ToString();
189         myGuildData.master.inDate    = guildJson["masterInDate"].ToString();
190
191         myGuildData.viceMasterList  = new List<GuildMemberData>();
192
193         LitJson.JsonData viceJson = guildJson["viceMasterList"];
194         for ( int i = 0; i < viceJson.Count; ++ i )
195         {
196             GuildMemberData vice = new GuildMemberData();
197             vice.nickname         = viceJson[i]["nickname"].ToString();
198             vice.inDate           = viceJson[i]["inDate"].ToString();
199
200             myGuildData.viceMasterList.Add(vice);
201         }
202     }
203     catch ( Exception e )
204     {
205         Debug.LogError(e);
206     }
207 });
208 }
209
210 public string GetGuildInfoBy(string guildName)...
```





# 내 길드 정보 조회

- 길드 마스터일 때만 executivesOption 오브젝트 활성화
  - GuildPage Script 수정

```
1  +using ...
3
4  public class GuildPage : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private TextMeshProUGUI textGuildName; // Popup 상단에 출력되는 길드 이름 Text UI
10     [SerializeField]
11     private GameObject executivesOption;
12
13     private string guildName = string.Empty; // 길드 이름
14
15     public void Activate(string guildName, bool isMaster=false)
16     {
17         executivesOption.SetActive(isMaster);
18
19         gameObject.SetActive(true);
20
21         textGuildName.text = guildName;
22         this.guildName = guildName;
23     }
24
25     public void OnClickApplyGuild()...
29 }
```



# 내 길드 정보 조회

- PopupGuild 오브젝트의 "GuildPage" 컴포넌트 변수 설정

The image shows the Unity development environment with two panels: Hierarchy and Inspector.

**Hierarchy Panel:** Displays the scene hierarchy. The path is: Lobby > Canvas > TopPanel > PopupGuild. The selected object is the inner PopupGuild, which has several child components: TopPanel, GuildInfo, ExecutivesOption, ButtonApplyGull, OverlayBackground, and PopupApplicants. The ExecutivesOption component is highlighted with a red box.

**Inspector Panel:** Shows the properties of the selected object. The main component is 'PopupGuild'. Below it, the 'Guild Page (Script)' component is expanded and highlighted with a red box. Its properties are:

- Script: GuildPage
- Backend Guild System: BackendSystem (Backend Guild)
- Text Guild Name: Title (Text Mesh Pro UGUI)
- Executives Option: ExecutivesOption (highlighted with a red dashed box and an arrow pointing from the ExecutivesOption component in the Hierarchy panel)

An 'Add Component' button is visible at the bottom of the Inspector panel.



# 내 길드 정보 조회

- “내 길드 정보” 버튼을 눌렀을 때 처리
  - GuildDefaultPage Script 수정

```
1  +using ...
3
4  public class GuildDefaultPage : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private TMP_InputField inputFieldGuildName;
10     [SerializeField]
11     private GuildPage guildPage;
12     [SerializeField]
13     private FadeEffect_TMP textLog;
14
15     public void OnClickSearchGuild()...
38
39     public void OnClickMyGuildInfo()
40     {
41         backendGuildSystem.GetMyGuildInfo();
42     }
43
44     public void SuccessMyGuildInfo()
45     {
46         bool isMaster = UserInfo.Data.nickname.Equals(backendGuildSystem.myGuildData.master.nickname);
47         guildPage.Activate(backendGuildSystem.myGuildData.guildName, isMaster);
48     }
49 }
```





# 내 길드 정보 조회

- 내 길드 정보 조회가 완료되었을 때 SuccessMyGuildInfo() 메소드 호출
  - BackendGuildSystem Script 수정

```
1  +using ...
5
6  -public class BackendGuildSystem : MonoBehaviour
7  {
8      [SerializeField]
9      private FadeEffect TMP      textLog;
10     [SerializeField]
11     private GuildDefaultPage    guildDefaultPage;
12     [SerializeField]
13     private GuildCreatePage     guildCreatePage;
14     [SerializeField]
15     private GuildApplicantsPage guildApplicantsPage;
16
17     public GuildData            myGuildData { private set; get; } = new GuildData();
18
19     + public void CreateGuild(string guildName, int goodsCount=1) ...
36
37     + public void ApplyGuild(string guildName) ...
55
56     + public void GetApplicants() ...
132
133    + public void ApproveApplicant(string gamerInDate) ...
147
148    + public void RejectApplicant(string gamerInDate) ...
164
```



# 내 길드 정보 조회

## BackendGuildSystem Script 수정 (계속)

```
165 public void GetMyGuildInfo()
166 {
167     Backend.Guild.GetMyGuildInfoV3(callback =>
168     {
169         if ( !callback.IsSuccess() )...
174
175         try
176         {
177             LitJson.JsonData guildJson = callback.GetFlattenJSON()["guild"];
178
179             if ( guildJson.Count <= 0 )...
184
185             myGuildData.guildName      = guildJson["guildName"].ToString();
186             myGuildData.guildInDate    = guildJson["inDate"].ToString();
187             myGuildData.memberCount    = int.Parse(guildJson["memberCount"].ToString());
188
189             myGuildData.master         = new GuildMemberData();
190             myGuildData.master.nickname = guildJson["masterNickname"].ToString();
191             myGuildData.master.inDate  = guildJson["masterInDate"].ToString();
192
193             myGuildData.viceMasterList = new List<GuildMemberData>();
194
195             LitJson.JsonData viceJson = guildJson["viceMasterList"];
196             for ( int i = 0; i < viceJson.Count; ++ i )...
204
205             // 내 길드 정보 불러오기 완료 후 처리
206             guildDefaultPage.SuccessMyGuildInfo();
207         }
208         catch ( Exception e )
209         {
210             Debug.LogError(e);
211         }
212     });
213 }
```



# 내 길드 정보 조회

- BackendSystem 오브젝트의 "BackendGuildSystem" 컴포넌트 변수 설정

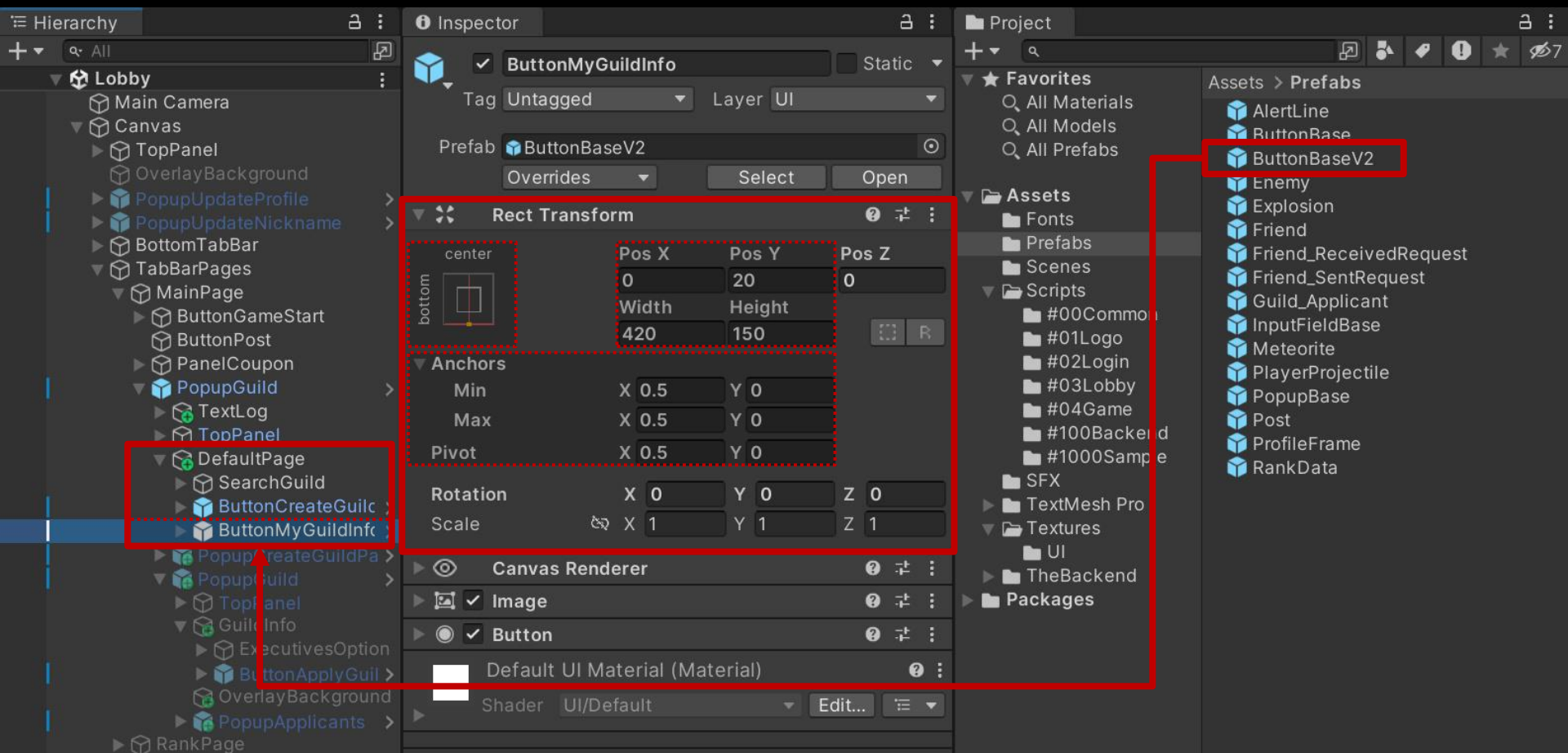
The image shows the Unity Hierarchy and Inspector panels. In the Hierarchy panel, the 'BackendSystem' object is selected under the 'Lobby\*' hierarchy. In the Inspector panel, the 'Backend Guild System (Script)' component is selected, and its variables are visible. The 'Guild Default Page' variable is highlighted with a red box, and a red arrow points from it to the 'DefaultPage' object in the Hierarchy panel.

Inspector Variable	Assigned Value
Script	BackendGuildSystem
Text Log	TextLog (Fade Effect_TMP)
Guild Default Page	DefaultPage (Guild Default Page)
Guild Create Page	PopupCreateGuildPage (Guild C)
Guild Applicants Page	PopupApplicants (Guild Applicar)



# 내 길드 정보 조회

- “내 길드 정보” Button UI 생성 및 설정
  - ButtonBaseV2 프리팹을 Hierarchy View로 Drag & Drop







# 내 길드 정보 조회

- “내 길드 정보” Button UI 생성 및 설정 (계속)

The image shows the Unity development environment. On the left is the Hierarchy panel, and on the right is the Inspector panel. The Hierarchy panel shows a tree view of the game scene, with 'Lobby\*' expanded to show various UI elements. A 'Text (TMP)' object is highlighted under 'ButtonMyGuildInfo'. The Inspector panel shows the properties for this 'Text (TMP)' object, including 'TextMeshPro - Text (UI)', 'Text Input', and 'Main Settings'. The text '내 길드 정보' is visible in the 'Text Input' field. The 'Main Settings' section shows the font asset set to 'NotoSansKR-Bold SDF (TMP\_Fc)' and the font size set to 50.



# 내 길드 정보 조회

- ButtonMyGuildInfo 오브젝트의 "Button" 컴포넌트 onClick() 이벤트 등록

The screenshot displays the Unity Hierarchy and Inspector panels. In the Hierarchy panel, the 'DefaultPage' object is selected, and its 'ButtonMyGuildInfo' child is highlighted. The Inspector panel shows the 'Button' component selected, with the 'On Click ()' event registration set to 'Runtime Only' and 'GuildDefaultPage.OnClickMyGui'. A red box highlights the 'Button' component and its 'On Click ()' event registration, with a red arrow pointing from the 'DefaultPage' object in the Hierarchy to the 'On Click ()' event in the Inspector.



# 내 길드 정보 조회

## ■ 결과 화면

고박사 2 35 / 30 6936 10721

닉네임 2 30 / 30 2734 6218

user03 2 30 / 30 0 1111



Enter text...

쿠폰등록



Enter text...

쿠폰등록



Enter text...

쿠폰등록

유니티노트

가입 요청 목록

가입 신청

유니티노트

가입 신청

notGuildMember 사전 조건을 만족하지 않습니다.

길드 (Guild)

검색할 길드명을 입력하세요.

길드 생성

게임시작

게임시작

게임시작



< 길드 마스터 >

< 길드 멤버 >

< 길드 소속 X >

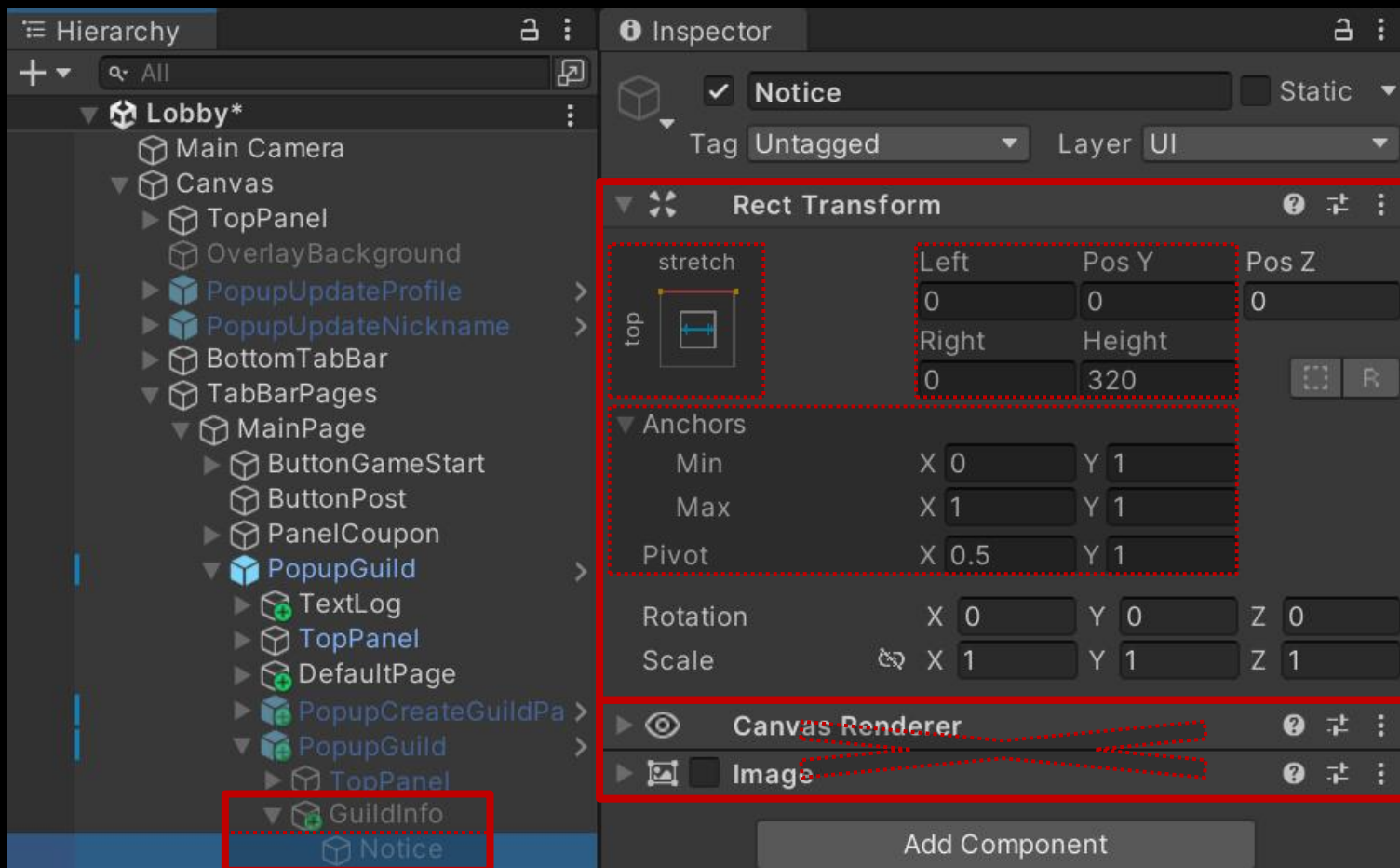


# 내 길드 정보 조회

## ■ 길드 메타 데이터를 이용한 공지사항 등록

### ■ 공지사항을 관리하는 Panel UI 생성 및 설정

#### □ GameObject - UI - Panel

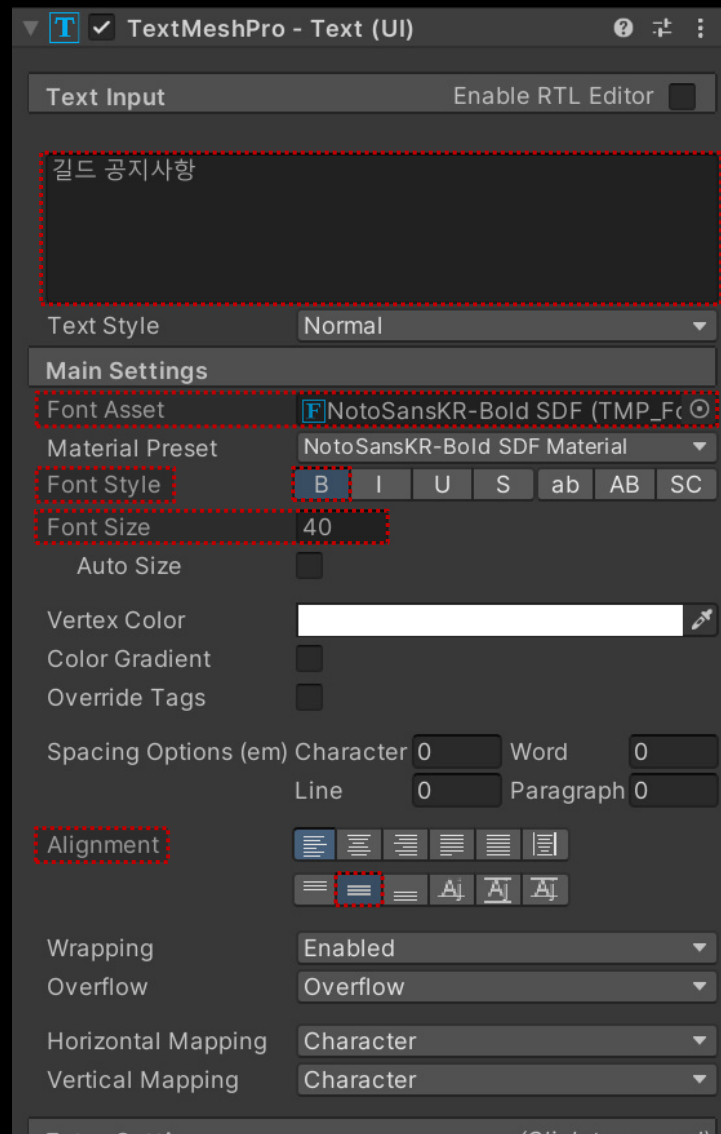
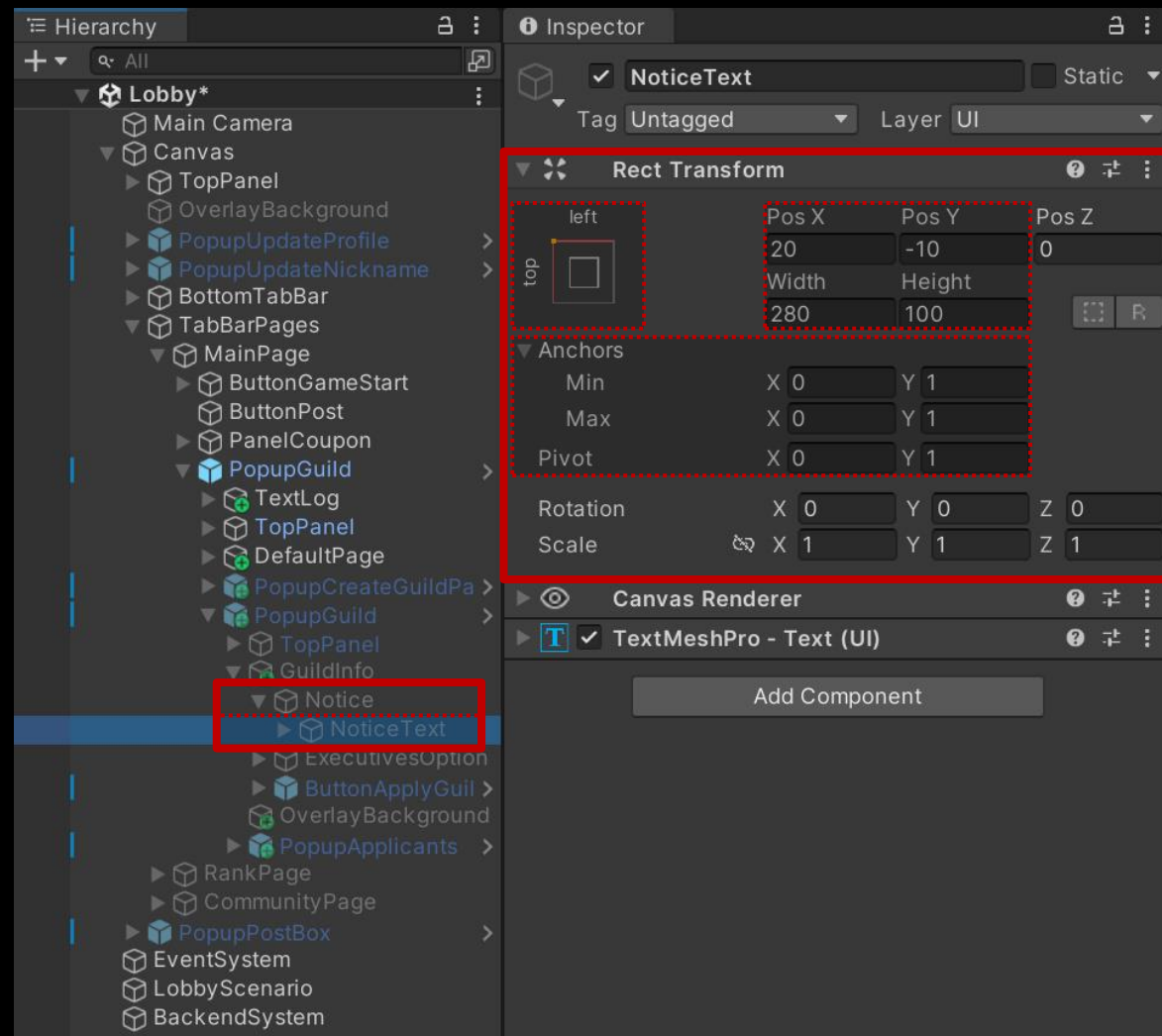






# 내 길드 정보 조회

- “길드 공지사항” 텍스트를 출력하는 “Text - TextMeshPro” UI 생성 및 설정
  - GameObject - UI - “Text - TextMeshPro”





# 내 길드 정보 조회

- 공지사항 배경을 출력하는 Image UI 생성 및 설정
  - GameObject - UI - Image

The screenshot displays the Unity development environment. On the left, the Hierarchy panel shows a tree structure under 'Lobby\*'. The 'NoticeBackground' object is selected and highlighted in blue. A red box highlights the 'NoticeBackground' object in the Hierarchy. On the right, the Inspector panel shows the properties for the selected object. The 'Rect Transform' component is expanded, showing 'stretch' checked and 'Anchors' set to Min (X 0, Y 1) and Max (X 1, Y 1). The 'Image' component is also expanded, showing 'Source Image' set to 'None (Sprite)' and 'Color' set to (152, 64, 64, 255). A red box highlights the 'Image' component and its 'Color' property.

Component	Property	Value
Rect Transform	stretch	checked
	Left	0
	Right	0
	Height	200
Anchors	Min X	0
	Min Y	1
	Max Y	1
Scale	X	1
	Y	1
	Z	1
Image	Source Image	None (Sprite)
	Color	(152, 64, 64, 255)

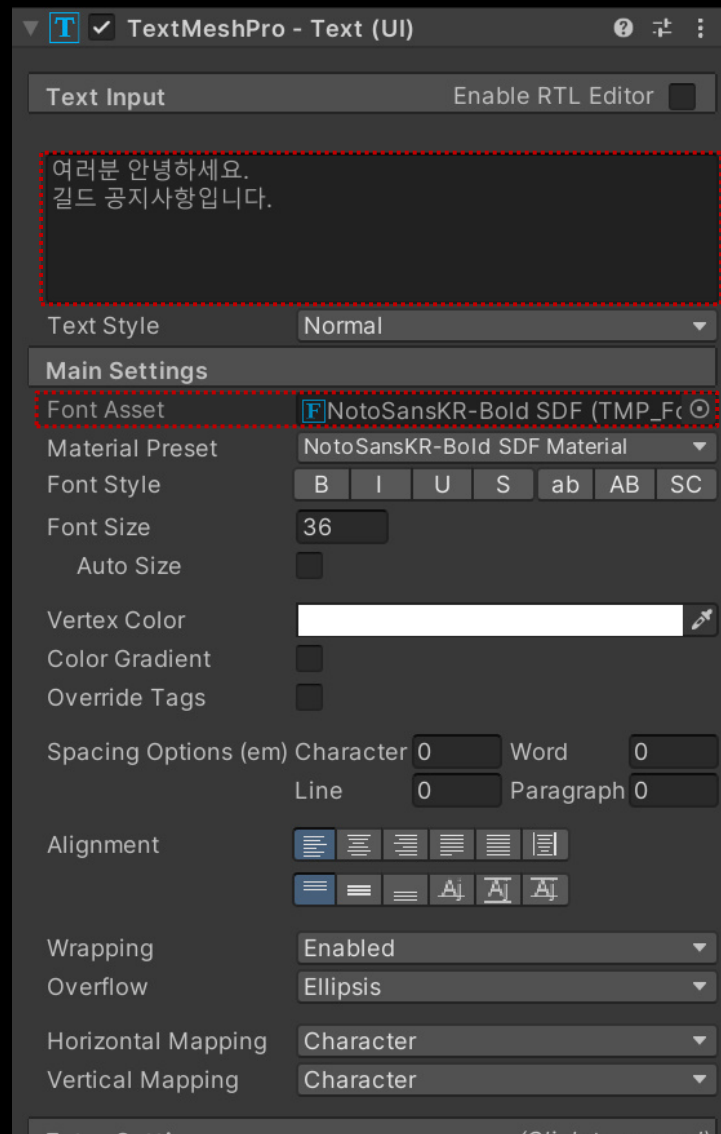
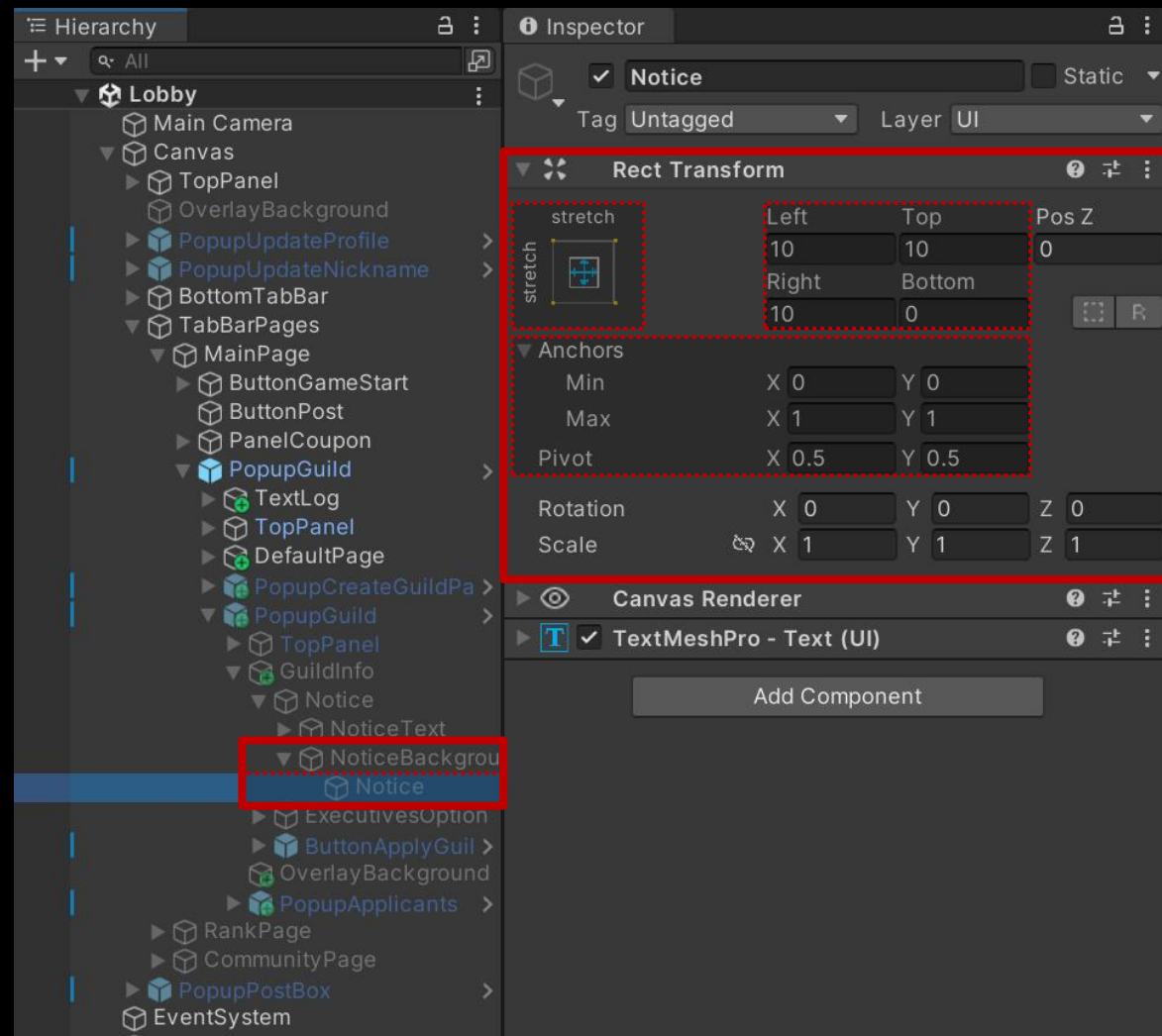
오브젝트의 이름을

NoticeBackground로 변경



# 내 길드 정보 조회

- 길드 공지사항을 출력하는 "Text - TextMeshPro" UI 생성 및 설정
  - GameObject - UI - "Text - TextMeshPro"





# 내 길드 정보 조회

- 길드 공지사항 입력 필드 생성 및 설정
  - InputFieldBase 프리팹을 Hierarchy View로 Drag & Drop

오브젝트의 이름을  
InputFieldNotice로 변경

**Inspector Panel:**

- Component: InputFieldNotice
- Tag: Untagged, Layer: UI
- Prefab: InputFieldBase
- Rect Transform:
  - Left: 0, Pos Y: -120, Pos Z: 0
  - Right: 0, Height: 200
  - Min: X 0, Y 1
  - Max: X 1, Y 1
  - Pivot: X 0.5, Y 1
  - Rotation: X 0, Y 0, Z 0
  - Scale: X 1, Y 1, Z 1
- Canvas Renderer
- Image
- TextMeshPro - Input Field

**Project Panel:**

- Assets > Prefabs
  - InputFieldBase



# 내 길드 정보 조회

## ■ 길드 공지사항 입력 필드 생성 및 설정 (계속)

**TextMeshPro - Input Field**

Interactable

Transition: Color Tint

Target Graphic: InputFieldNotice (Image)

Normal Color: [Color Picker]

Highlighted Color: [Color Picker]

Pressed Color: [Color Picker]

Selected Color: [Color Picker]

Disabled Color: [Color Picker]

Color Multiplier: 1

Fade Duration: 0.1

Navigation: Automatic

Visualize

Text Viewport: Text Area (Rect Transform)

Text Component: Text (Text Mesh Pro UGUI)

**Text**

여러분 안녕하세요.  
길드 공지사항입니다.

**Input Field Settings**

Font Asset: NotoSansKR-Regular SDF (T)

Point Size: 36

Character Limit: 0

Content Type: Standard

Line Type: Multi Line Newline

Line Limit: 3





# 내 길드 정보 조회

- “공지사항 등록” Button UI 생성 및 설정
  - ButtonBaseV2 프리팹을 Hierarchy View로 Drag & Drop

오브젝트의 이름을 ButtonNoticeRegistration으로 변경

Rect Transform		Pos X	Pos Y	Pos Z
right		-310	0	0
middle		Width	Height	
		300	100	

Anchors		X	Y
Min		1	0.5
Max		1	0.5
Pivot		1	0.5

Rotation		X	Y	Z
		0	0	0

Scale		X	Y	Z
		1	1	1

Inspector Panel: ButtonNoticeRegistration (Static) | Tag: Untagged | Layer: UI | Prefab: ButtonBaseV2 | Overrides | Select | Open

Project Panel: Assets > Prefabs | ButtonBaseV2



# 내 길드 정보 조회

- “공지사항 등록” Button UI 생성 및 설정 (계속)

The screenshot displays the Unity development environment. On the left, the Hierarchy panel shows a tree view of the scene objects. The 'Lobby\*' folder is expanded, showing 'Canvas' > 'TopPanel' > 'OverlayBackground' > 'PopupUpdateProfile' > 'PopupUpdateNickname' > 'BottomTabBar' > 'TabBarPages' > 'MainPage' > 'ButtonGameStart' > 'ButtonPost' > 'PanelCoupon' > 'PopupGuild' > 'TextLog' > 'TopPanel' > 'DefaultPage' > 'PopupCreateGuildPa' > 'PopupGuild' > 'TopPanel' > 'GuildInfo' > 'Notice' > 'ExecutivesOption' > 'ButtonNotice' > 'Text (TMP)'. The 'Text (TMP)' object is selected and highlighted in blue.

On the right, the Inspector panel shows the properties for the selected 'Text (TMP)' object. The 'TextMeshPro - Text (UI)' component is active. The 'Text Input' field contains the text '공지사항 등록'. The 'Text Style' is set to 'Normal'. The 'Main Settings' section shows 'Font Asset' set to 'NotoSansKR-Bold SDF (TMP\_Fc)', 'Material Preset' set to 'NotoSansKR-Bold SDF Material', and 'Font Style' set to 'B'. The 'Font Size' is set to '40'. The 'Auto Size' checkbox is unchecked.



# 내 길드 정보 조회

- NoticeBackground, InputFieldNotice 오브젝트 비활성화

The screenshot displays the Unity Hierarchy and Inspector panels. The Hierarchy panel on the left shows the scene structure under 'Lobby\*'. The Inspector panel on the right shows the properties of the selected 'InputFieldNotice' object. The 'Rect Transform' component is expanded, showing the 'stretch' and 'top' properties. The 'Canvas Renderer' component is also visible. A red box highlights the 'NoticeBackground' and 'InputFieldNotice' objects in the Hierarchy panel. A text box at the bottom right explains the object activation status.

길드 마스터는 InputFieldNotice 오브젝트가 활성화되고, 길드원은 NoticeBackground 오브젝트가 활성화된다.





# 내 길드 정보 조회

- 길드 공지사항을 저장하는 변수 선언
  - GuildData Script 수정

```
1  using System.Collections.Generic;
2
3  public class GuildData
4  {
5      public string      guildName;      // 길드 이름
6      public string      guildInDate;    // 길드 InDate
7      public string      notice;         // 길드 공지사항 (메타 데이터)
8      public int          memberCount;    // 길드 회원 수
9      public GuildMemberData master;     // 길드 마스터
10     public List<GuildMemberData> viceMasterList; // 부길드 마스터 목록
11 }
```



# 내 길드 정보 조회

- ModifyGuildV3() 메소드를 호출해 길드 메타 데이터 설정 [공지사항]
  - BackendGuildSystem Script 수정

```
19 public void CreateGuild(string guildName, int goodsCount=1)
20 {
21     Backend.Guild.CreateGuildV3(guildName, goodsCount, callback =>
22     {
23         if ( !callback.IsSuccess() )...
29
30         Debug.Log($"길드가 생성되었습니다. : {callback}");
31
32         // 길드를 생성할 때 "길드 공지사항입니다."를 공지사항으로 기본 설정
33         SetNotice("길드 공지사항입니다.");
34
35         // 길드 생성에 성공했을 때 호출
36         guildCreatePage.SuccessCreateGuild();
37     });
38 }
39
40 public void ApplyGuild(string guildName)...
58
59 public void GetApplicants()...
135
136 public void ApproveApplicant(string gamerInDate)...
150
151 public void RejectApplicant(string gamerInDate)...
167
```



# 내 길드 정보 조회

## BackendGuildSystem Script 수정 (계속)

```
168 public void GetMyGuildInfo()
169 {
170     Backend.Guild.GetMyGuildInfoV3(callback =>
171     {
172         if ( !callback.IsSuccess() )...
173     }
174     try
175     {
176         LitJson.JsonData guildJson = callback.GetFlattenJSON()["guild"];
177
178         if ( guildJson.Count <= 0 )...
179
180         myGuildData.guildName      = guildJson["guildName"].ToString();
181         myGuildData.guildInDate    = guildJson["inDate"].ToString();
182         myGuildData.notice         = guildJson["NOTICE"].ToString();
183         myGuildData.memberCount    = int.Parse(guildJson["memberCount"].ToString());
184
185         myGuildData.master         = new GuildMemberData();
186         myGuildData.master.nickname = guildJson["masterNickname"].ToString();
187         myGuildData.master.inDate  = guildJson["masterInDate"].ToString();
188
189         myGuildData.viceMasterList = new List<GuildMemberData>();
190
191         LitJson.JsonData viceJson = guildJson["viceMasterList"];
192         for ( int i = 0; i < viceJson.Count; ++ i )...
193
194         // 내 길드 정보 불러오기 완료 후 처리
195         guildDefaultPage.SuccessMyGuildInfo();
196     }
197     catch ( Exception e )
198     {
199         Debug.LogError...
200     }
201 });
202 }
```

**Tip.** 길드를 생성할 때와 길드마스터가 공지사항을 수정할 때만 공지사항 업데이트가 되기 때문에 이전에 만든 길드의 경우 NOTICE 를 불러오지 못해 에러가 발생할 수 있다.  
(JsonData를 불러올 때 값이 없으면 기본 값을 별도로 설정)



# 내 길드 정보 조회

## BackendGuildSystem Script 수정 (계속)

```
219 public void SetNotice(string notice)
220 {
221     Param param = new Param { { "NOTICE", notice } };
222
223     Backend.Guild.ModifyGuildV3(param, callback =>
224     {
225         if ( !callback.IsSuccess() )
226         {
227             ErrorLog(callback.GetMessage(), "Guild_Failed_Log", "SetNotice");
228
229             return;
230         }
231
232         Debug.Log($"길드 메타 데이터[공지사항] 변경에 성공했습니다. : {callback}");
233     });
234 }
```

Tip. 버프 레벨 등과 같이 길드에 추가적으로 필요한 데이터를 메타 데이터로 저장/불러오기 가능  
교재에서는 길드 공지사항을 메타 데이터로 저장/불러오기 함

```
236 public string GetGuildInfoBy(string guildName) {...}
261
262 private void ErrorLogCreateGuild(BackendReturnObject callback) {...}
281
282 private void ErrorLogApplyGuild(BackendReturnObject callback) {...}
304
305 private void ErrorLogApproveApplicant(BackendReturnObject callback) {...}
321
322 private void ErrorLog(string message, string behaviorType="", string paramKey="") {...}
335 }
```





# 내 길드 정보 조회

- 길드 공지사항을 관리하는 스크립트 생성 및 작성
  - C# Script 생성 후 스크립트의 이름을 "Notice"로 변경

```
1  using UnityEngine;
2  using TMPro;
3
4  public class Notice : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private GameObject noticeBackground; // 길드원에게 보이는 오브젝트
10     [SerializeField]
11     private TextMeshProUGUI textNotice; // 길드원에게 보이는 공지사항
12     [SerializeField]
13     private TMP_InputField inputFieldNotice; // 길드마스터에게 보이는 공지사항 오브젝트
14     [SerializeField]
15     private FadeEffect_TMP textLog;
16
```



# 내 길드 정보 조회

- 길드 공지사항을 관리하는 스크립트 생성 및 작성 (계속)

```
17 public void Setup(bool isMaster=false)
18 {
19     textNotice.text      = backendGuildSystem.myGuildData.notice;
20     inputFieldNotice.text = backendGuildSystem.myGuildData.notice;
21
22     noticeBackground.SetActive(!isMaster);
23     inputFieldNotice.gameObject.SetActive(isMaster);
24 }
25
26 public void OnClickNoticeRegistration()
27 {
28     string notice = inputFieldNotice.text;
29
30     if ( notice.Trim().Equals("") )
31     {
32         textLog.FadeOut("공지사항 내용이 비어있습니다.");
33         return;
34     }
35
36     backendGuildSystem.SetNotice(notice);
37 }
38 }
```



# 내 길드 정보 조회

- Notice 오브젝트에 "Notice" 컴포넌트 추가 및 설정

The screenshot displays the Unity Hierarchy and Inspector panels. The Hierarchy panel on the left shows a tree structure under 'Lobby\*'. A 'Notice' object is selected, and its components are listed in the Inspector panel on the right. Red boxes and arrows highlight the configuration process:

- Hierarchy Panel:**
  - 'TextLog' is highlighted with a red box.
  - 'Notice' is highlighted with a red box.
  - 'Notice' sub-objects ('Notice Text', 'NoticeBackgrol', 'Notice', 'InputFieldNot') are highlighted with red boxes.
  - 'BackendSystem' is highlighted with a red box at the bottom of the hierarchy.
- Inspector Panel:**
  - The 'Notice (Script)' component is selected.
  - The 'Script' dropdown is set to 'Notice'.
  - The 'Backend Guild System' dropdown is set to 'BackendSystem (Backend Guild)'. A red box and arrow point from this dropdown to the 'BackendSystem' object in the Hierarchy.
  - The 'Notice Background' dropdown is set to 'NoticeBackground'.
  - The 'Text Notice' dropdown is set to 'Notice (Text Mesh Pro UGUI)'. A red box and arrow point from this dropdown to the 'Notice' sub-object in the Hierarchy.
  - The 'Input Field Notice' dropdown is set to 'InputFieldNotice (TMP\_Input Fiel)'. A red box and arrow point from this dropdown to the 'InputFieldNot' sub-object in the Hierarchy.
  - The 'Text Log' dropdown is set to 'TextLog (Fade Effect\_TMP)'. A red box and arrow point from this dropdown to the 'TextLog' object in the Hierarchy.



# 내 길드 정보 조회

- ButtonNoticeRegistration 오브젝트의 "Button" 컴포넌트 onClick() 이벤트 등록

The image shows the Unity development environment. On the left is the Hierarchy panel, and on the right is the Inspector panel.

**Hierarchy Panel:** Shows a tree view of the scene. The 'Lobby\*' folder is expanded, showing various UI elements. The 'Notice' folder is expanded, and 'ButtonNoticef' is selected. A red box highlights the 'ButtonNoticef' object.

**Inspector Panel:** Shows the properties of the selected 'Button' component. The 'On Click ()' event list is visible, containing one event: 'Notice.OnClickNoticeRegistratio'. A red box highlights the event list, and a red arrow points from the 'ButtonNoticef' object in the Hierarchy panel to the event list.





# 내 길드 정보 조회

- 길드 페이지가 활성화될 때 notice.Setup() 메소드 호출
  - GuildPage Script 수정

```
1  +using ...
3
4  public class GuildPage : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private TextMeshProUGUI textGuildName; // Popup 상단에 출력되는 길드 이름 Text UI
10     [SerializeField]
11     private Notice notice;
12     [SerializeField]
13     private GameObject executivesOption;
14
15     private string guildName = string.Empty; // 길드 이름
16
17     public void Activate(string guildName, bool isMaster=false)
18     {
19         notice.Setup(isMaster);
20         executivesOption.SetActive(isMaster);
21
22         gameObject.SetActive(true);
23
24         textGuildName.text = guildName;
25         this.guildName = guildName;
26     }
27
28     public void OnClickApplyGuild()...
32 }
```



# 내 길드 정보 조회

- PopupGuild 오브젝트의 "GuildPage" 컴포넌트 변수 설정

The image shows the Unity Hierarchy and Inspector panels. The Hierarchy panel on the left shows a tree structure with 'Lobby\*' expanded to 'MainPage' > 'PopupGuild'. The Inspector panel on the right shows the 'Guild Page (Script)' component with the following variables:

- Script: GuildPage
- Backend Guild System: BackendSystem (Backend Guild)
- Text Guild Name: Title (Text Mesh Pro UGUI)
- Notice: Notice (Notice)
- Executives Option: ExecutivesOption

A red box highlights the 'Notice' variable in the Inspector, and a red arrow points from the 'Notice' object in the Hierarchy to it.



# 내 길드 정보 조회

## ■ 결과 화면 (길드 공지사항 수정)

The screenshot displays a game interface with a top toolbar showing 'Scene', 'Game', 'Animator', and 'Animation' views. The main area features a player profile for '교박사' (Gyobaksa) with level 2, 35/30 health, 6936 diamonds, and 10721 coins. Below the profile is a text input field with the placeholder 'Enter text...' and a '쿠폰등록' (Coupon Register) button. A '길드 (Guild)' section contains a search input field with the text '검색할 길드명을 입력하세요.' and a '길드 검색' (Guild Search) button. At the bottom of this section are '길드 생성' (Guild Create) and '내 길드 정보' (My Guild Info) buttons. A large yellow '게임시작' (Start Game) button is positioned below the guild section. The bottom navigation bar includes icons for a shop, inventory, combat, a leaderboard, and a profile.

The console log on the right side of the interface shows the following messages:

- [23:23:35] Level : 10, Max Exp : 51200, Reward Gold : 100000  
UnityEngine.Debug.Log (object)
- [23:23:35] 게임 정보 데이터 불러오기에 성공했습니다. : statusCode : 200  
message : Success
- [23:23:43] 길드가 생성되었습니다. : statusCode : 204  
message : Success
- [23:23:43] 길드 메타 데이터[공지사항] 변경에 성공했습니다. : statusCode : 204  
message : Success
- [23:23:54] 길드 메타 데이터[공지사항] 변경에 성공했습니다. : statusCode : 204  
message : Success



# 내 길드 정보 조회

## ■ 결과 화면

고박사 2 35 / 30 6936 10721

닉네임 2 30 / 30 2734 6218



Enter text...

쿠폰등록



Enter text...

쿠폰등록

유니티노트 ✕

길드 공지사항 공지사항 등록 가입 요청 목록

안녕하세요.  
유니티노트 길드입니다.  
잘 부탁드립니다.

가입 신청

유니티노트 ✕

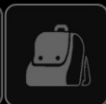
길드 공지사항

안녕하세요.  
유니티노트 길드입니다.  
잘 부탁드립니다.

가입 신청

게임시작

게임시작



< 길드 마스터 >

< 부길드 마스터 / 길드 멤버 >



# 내 길드 정보 조회

## ■ 길드 인원수 출력

- 길드 인원수를 출력하는 "Text - TextMeshPro" UI 생성 및 설정
  - GameObject - UI - "Text - TextMeshPro"

오브젝트의 이름을  
GuildMemberCount로 변경

길드 인원 1/100

Text Style Normal

Main Settings

Font Asset NotoSansKR-Bold SDF (TMP\_Fc)

Material Preset NotoSansKR-Bold SDF Material

Font Style B I U S ab AB SC

Font Size 40

Auto Size

Vertex Color

Color Gradient

Override Tags

Spacing Options (em) Character 0 Word 0 Line 0 Paragraph 0

Alignment

Wrapping Enabled

Overflow Overflow





# 내 길드 정보 조회

## ■ 길드 인원수 출력

### □ GuildPage Script 수정

```
1  using ...
3
4  public class GuildPage : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private TextMeshProUGUI textGuildName;           // Popup 상단에 출력되는 길드 이름 Text UI
10     [SerializeField]
11     private Notice notice;
12     [SerializeField]
13     private GameObject executivesOption;
14     [SerializeField]
15     private TextMeshProUGUI textMemberCount;
16
17     private string guildName = string.Empty; // 길드 이름
18
19     public void Activate(string guildName, bool isMaster=false)
20     {
21         notice.Setup(isMaster);
22         executivesOption.SetActive(isMaster);
23
24         gameObject.SetActive(true);
25
26         textGuildName.text = guildName;
27         this.guildName = guildName;
28         textMemberCount.text = $"길드 인원 {backendGuildSystem.myGuildData.memberCount}/100";
29     }
30
31     public void OnClickApplyGuild()...
35 }
```



# 내 길드 정보 조회

- PopupGuild 오브젝트의 "GuildPage" 컴포넌트 변수 설정

The screenshot displays the Unity development environment. On the left, the Hierarchy panel shows the scene structure under 'Lobby', with 'PopupGuild' selected and highlighted by a red box. On the right, the Inspector panel shows the 'Guild Page (Script)' component. The variables are listed as follows:

Variable Name	Assigned Value
Script	GuildPage
Backend Guild System	BackendSystem (Backend Guild)
Text Guild Name	Title (Text Mesh Pro UGUI)
Notice	Notice (Notice)
Executives Option	ExecutivesOption
Text Member Count	GuildMemberCount (Text Mesh Pro UGUI)

The 'Text Member Count' variable and its assigned value are highlighted with a red dashed box. A red arrow points from this variable to the 'GuildMemberCount' object in the Hierarchy panel, which is also highlighted with a red box.



# 내 길드 정보 조회


## ■ 결과 화면


닉네임
30 / 30
2734
6218



Enter text...

쿠폰등록


user03
30 / 30
0
1111



Enter text...

쿠폰등록

**유니티노트** [X]

길드 공지사항

안녕하세요.  
유니티노트 길드입니다.  
잘 부탁드립니다.

길드 인원 2/100

가입 신청

**unitynote** [X]

길드 공지사항

공지사항 등록

가입 요청 목록

길드 공지사항입니다.

길드 인원 1/100

가입 신청

게임시작

게임시작



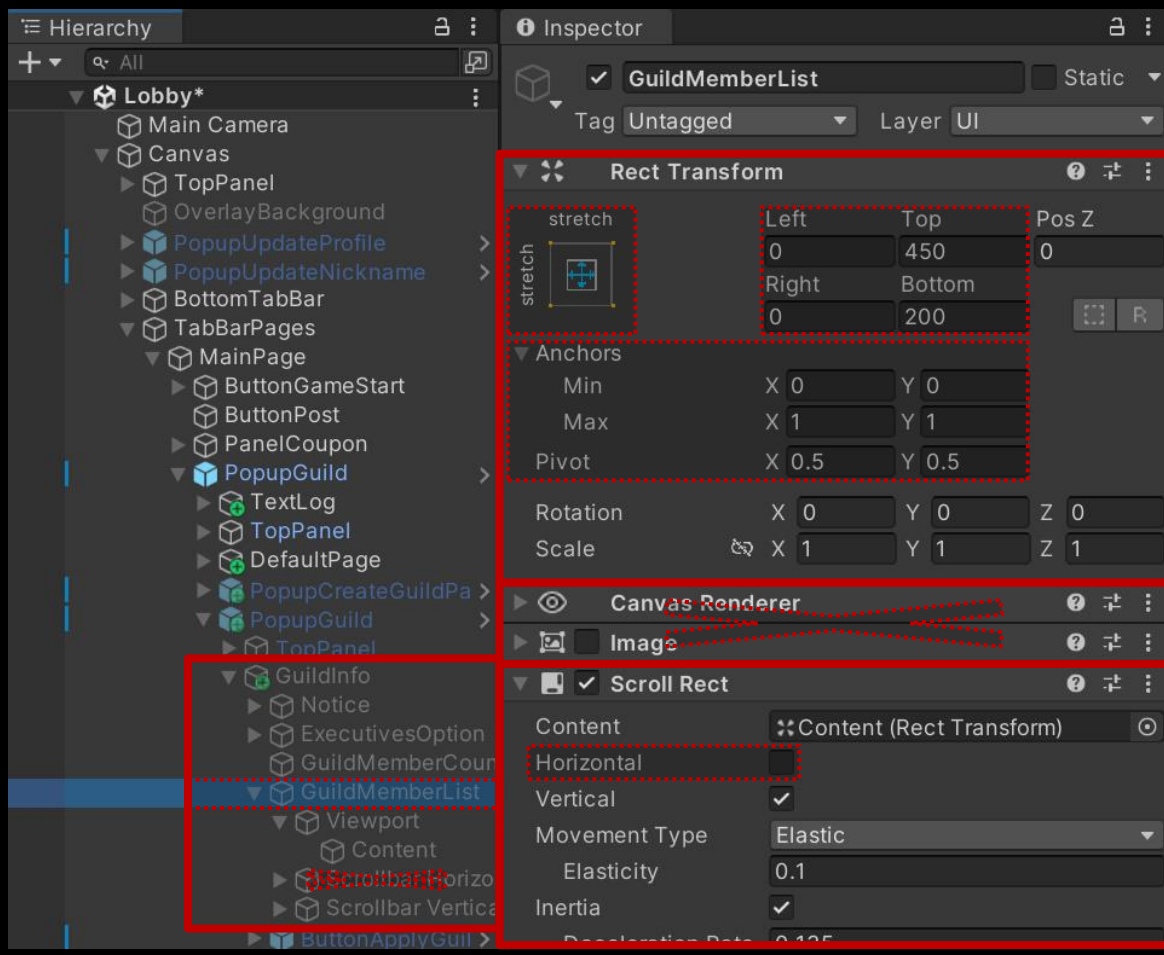


# 내 길드 정보 조회

## ■ 모든 길드원 정보 조회

### ■ 길드원 목록을 출력하는 Scroll View UI 생성 및 설정

#### □ GameObject - UI - Scroll View





# 내 길드 정보 조회

- Scrollbar가 보이지 않도록 Width를 0으로 설정

The screenshot shows the Unity Inspector window with the following components and settings:

- Rect Transform** (highlighted with a red box):
  - right: Pos X: 0, Top: 0, Pos Z: 0
  - stretch:
  - Width: 0 (highlighted with a red dashed box)
  - Bottom: 17
- Anchors**:
  - Min: X 1, Y 0
  - Max: X 1, Y 1
  - Pivot: X 1, Y 1
- Rotation**: X 0, Y 0, Z 0
- Scale**: X 1, Y 1, Z 1

Below the Rect Transform component, the following components are listed:

- Canvas Renderer
- Image
- Scrollbar

An "Add Component" button is visible at the bottom of the Inspector.

In the Hierarchy window on the left, the "Scrollbar Vertical" component is highlighted with a red box.





# 내 길드 정보 조회

- Content 오브젝트에 컴포넌트 추가 및 설정

The screenshot displays the Unity development environment. On the left, the Hierarchy panel shows a tree structure under 'Lobby\*'. The 'Content' object is selected at the bottom. On the right, the Inspector panel shows the properties for the selected 'Content' object. The 'Rect Transform' component is visible above the 'Vertical Layout Group' component. The 'Vertical Layout Group' component is highlighted with a red dashed box and has the following settings:

- Padding: Left (20), Right (20), Top (20), Bottom (20), Spacing (20)
- Child Alignment: Upper Center
- Reverse Arrangement:
- Control Child Size:  Width,  Height
- Use Child Scale:  Width,  Height
- Child Force Expand:  Width,  Height

Below the 'Vertical Layout Group' component is the 'Content Size Fitter' component, also highlighted with a red dashed box, with the following settings:

- Horizontal Fit: Unconstrained
- Vertical Fit: Preferred Size

At the bottom of the Inspector panel, there is an 'Add Component' button.



# 내 길드 정보 조회

- 길드원 정보를 출력하는 Button UI 생성 및 설정
  - GameObject - UI - "Button - TextMeshPro"

The screenshot displays the Unity development environment with the following components highlighted:

- Hierarchy Panel:** The `Guild_Member` GameObject is selected under the `Lobby*` Canvas.
- Inspector Panel:**
  - Rect Transform:** The `stretch` property is checked. The `Left` and `Right` values are set to 10, and the `Height` is set to 170.
  - Anchors:** The `Min` and `Max` anchors are set to X 0 and X 1, and the `Pivot` is set to X 0.5 and Y 0.5.
  - Canvas Renderer:** The `Image` component is active.
  - Image Component:** The `Source Image` is set to `UI_RoundedRect_Empty`.
- Project Panel:** The `Assets > Textures > UI` folder structure is visible, with `UI_RoundedRect_Empty` highlighted.



# 내 길드 정보 조회

- 길드 직책을 출력하는 "Text - TextMeshPro" UI 설정

Hierarchy

- Lobby\*
  - Main Camera
  - Canvas
    - TopPanel
    - OverlayBackground
    - PopupUpdateProfile
    - PopupUpdateNickname
    - BottomTabBar
    - TabBarPages
    - PopupPostBox
    - Guild\_Member**
      - Position**
    - EventSystem
    - LobbyScenario
    - BackendSystem

Inspector

Position

Tag Untagged Layer UI

Rect Transform

left	Pos X	Top	Pos Z
10	10	10	0
Width	Bottom		
300	10		

Anchors

Min	X	Y
0	0	0
Max	X	Y
0	0	1

Pivot

X	Y
0	0.5

Rotation

X	Y	Z
0	0	0

Scale

X	Y	Z
1	1	1

Canvas Renderer

TextMeshPro - Text (UI)

NotoSansKR-Bold SDF Material (Material)

Shader TextMeshPro/Distance Fi

Add Component

Text (TMP) 오브젝트의 이름을 Position으로 변경

TextMeshPro - Text (UI)

Text Input Enable RTL Editor

길드마스터

Text Style Normal

Main Settings

Font Asset NotoSansKR-Bold SDF (TMP\_Fc)

Material Preset NotoSansKR-Bold SDF Material

Font Style B I U S ab AB SC

Font Size 50

Auto Size

Vertex Color Color (255, 255, 255, 255)

Color Gradient

Override Tags

Spacing Options (em) Character 0 Word 0 Line 0 Paragraph 0

Alignment

Wrapping Enabled

Overflow Overflow

Horizontal Mapping Character

Vertical Mapping Character



# 내 길드 정보 조회

- 길드원 닉네임을 출력하는 "Text - TextMeshPro" UI 생성 및 설정
  - Position 오브젝트를 Ctrl+D로 복제

Hierarchy

- Lobby\*
- Main Camera
- Canvas
  - TopPanel
  - OverlayBackground
  - PopupUpdateProfile
  - PopupUpdateNickname
  - BottomTabBar
  - TabBarPages
  - PopupPostBox
  - Guild\_Member
    - Position
    - Nickname**
  - EventSystem
  - LobbyScenario
  - BackendSystem

Inspector

Nickname

Tag Untagged Layer UI

Rect Transform

left	Pos X	Top	Pos Z
stretch	320	10	0
	Width	Bottom	
	400	10	

Anchors

Min	X	Y
	0	0
Max	X	Y
	0	1
Pivot	X	Y
	0	0.5

Rotation

X	Y	Z
0	0	0

Scale

X	Y	Z
1	1	1

Canvas Renderer

TextMeshPro - Text (UI)

NotoSansKR-Bold SDF Material (Material)

Shader TextMeshPro/Distance Fi

Add Component

TextMeshPro - Text (UI)

Text Input Enable RTL Editor

고박사

Text Style Normal

Main Settings

Font Asset NotoSansKR-Bold SDF (TMP\_Fc)

Material Preset NotoSansKR-Bold SDF Material

Font Style B I U S ab AB SC

Font Size 50

Auto Size

Vertex Color

Color Gradient

Override Tags

Spacing Options (em) Character 0 Word 0

Line 0 Paragraph 0

Alignment

Wrapping Enabled

Overflow Overflow

Horizontal Mapping Character

Vertical Mapping Character



# 내 길드 정보 조회

- 기부한 굿즈 개수를 출력하는 "Text - TextMeshPro" UI 생성 및 설정
  - Position 오브젝트를 Ctrl+D로 복제

The screenshot shows the Unity Hierarchy and Inspector panels. In the Hierarchy panel, the 'GoodsCount' object is selected under the 'Lobby\*' folder. In the Inspector panel, the 'Rect Transform' component is expanded, and a red dashed box highlights the 'right' anchor and 'stretch' properties. The 'right' anchor is set to Pos X: -420, Top: 10, Pos Z: 0, Width: 270, Bottom: 10. The 'stretch' property is set to 'right'. Below the 'Rect Transform' component, the 'Canvas Renderer' and 'TextMeshPro - Text (UI)' components are visible.

The screenshot shows the Unity Inspector panel for the 'TextMeshPro - Text (UI)' component. The 'Text Input' field contains the number '0'. The 'Text Style' is set to 'Normal'. The 'Main Settings' section shows the 'Font Asset' as 'NotoSansKR-Bold SDF (TMP\_Fc)', 'Material Preset' as 'NotoSansKR-Bold SDF Material', 'Font Style' as 'B I U S ab AB SC', 'Font Size' as '50', and 'Auto Size' as 'Auto Size'. The 'Vertex Color' is set to white. The 'Spacing Options' section shows 'Character' spacing as 0, 'Word' spacing as 0, 'Line' spacing as 0, and 'Paragraph' spacing as 0. The 'Alignment' section shows 'Left' alignment. The 'Wrapping' is set to 'Enabled' and 'Overflow' is set to 'Overflow'. The 'Horizontal Mapping' and 'Vertical Mapping' are both set to 'Character'.





# 내 길드 정보 조회

- 접속 기록을 출력하는 "Text - TextMeshPro" UI 생성 및 설정
  - Position 오브젝트를 Ctrl+D로 복제

Hierarchy: Lobby\* > Date

Inspector: Rect Transform

right	Pos X	Top	Pos Z
stretch	-10	10	0
	Width	Bottom	
	400	10	

Anchors

Min	X	Y
	1	0
Max	X	Y
	1	1
Pivot	X	Y
	1	0.5

Rotation: X 0, Y 0, Z 0

Scale: X 1, Y 1, Z 1

Canvas Renderer

TextMeshPro - Text (UI)

NotoSansKR-Bold SDF Material (Material)

Shader: TextMeshPro/Distance Fi

Add Component

Text Input: Enable RTL Editor

72시간 전

Text Style: Normal

Main Settings

Font Asset: NotoSansKR-Bold SDF (TMP\_Fc)

Material Preset: NotoSansKR-Bold SDF Material

Font Style: B I U S ab AB SC

Font Size: 50

Auto Size: Off

Vertex Color: [Color Picker]

Color Gradient: [Color Gradient]

Override Tags: Off

Spacing Options (em): Character 0, Word 0, Line 0, Paragraph 0

Alignment: Left

Wrapping: Enabled

Overflow: Overflow

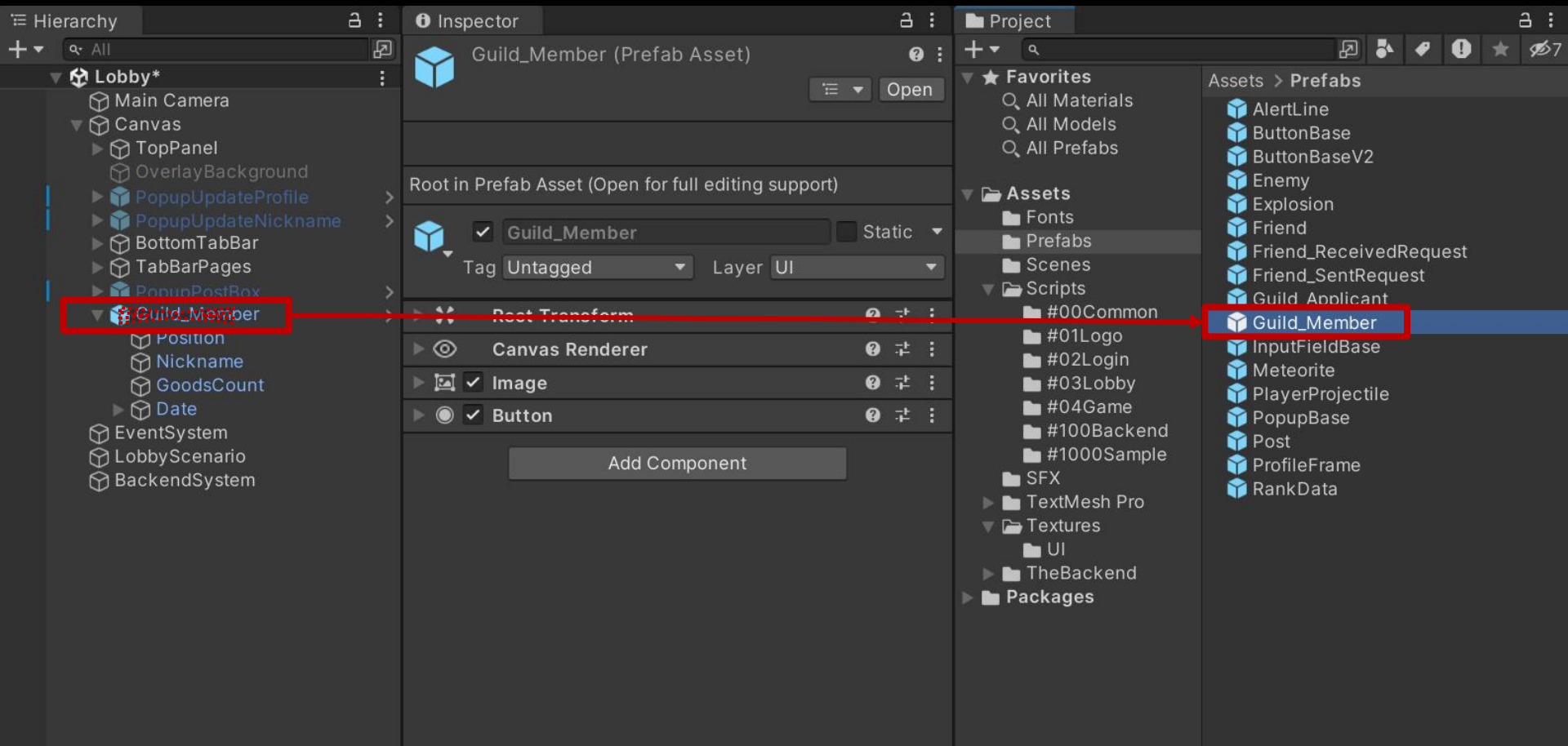
Horizontal Mapping: Character

Vertical Mapping: Character



# 내 길드 정보 조회

- Guild\_Member 오브젝트 Prefab 생성
  - Hierarchy View의 "Guild\_Member" 오브젝트를 Project View로 Drag & Drop
  - Hierarchy View에 있는 "Guild\_Member" 오브젝트 삭제





# 내 길드 정보 조회

- 길드원 정보에 직책, 굿즈 개수, 마지막 접속 날짜 변수 추가
  - GuildMemberData Script 수정

```
1 public class GuildMemberData
2 {
3     public string nickname; // 닉네임
4     public string inDate; // 해당 유저의 inDate
5     public string level; // 해당 유저의 level
6     public string position; // 길드 직책
7     public int goodsCount; // 굿즈 개수
8     public string lastLogin; // 마지막 접속 날짜
9
10 public override string ToString()
11 {
12     string result = string.Empty;
13     result += $"nickname : {nickname}\n";
14     result += $"inDate : {inDate}\n";
15     result += $"level : {level}\n";
16     result += $"position : {position}\n";
17     result += $"goodsCount : {goodsCount}\n";
18     result += $"lastLogin : {lastLogin}\n";
19
20     return result;
21 }
22 }
```



# 내 길드 정보 조회

## ■ 길드 회원 목록 조회 메소드

```
GetGuildMemberListV3(string guildInDate);  
GetGuildMemberListV3(string guildInDate, int limit);  
GetGuildMemberListV3(string guildInDate, string firstKey);  
GetGuildMemberListV3(string guildInDate, int limit, string firstKey);  
guildInDate 길드의 회원 목록 조회
```

Value	설명	default
guildInDate	회원 목록을 불러올 길드의 inDate	-
limit	(Optional) 불러올 길드 회원 목록 수	100
firstKey	(Optional) 불러올 길드 회원 목록의 시작점(offset)	-

### Error Cases

\* 존재하지 않는 guildInDate일 경우 (404 / guild not found, ..)



# 내 길드 정보 조회

## ■ 길드 회원 목록 JsonData

```
{
  "rows":
  [
    {
      // 길드원 1
      // 길드의 굿즈 개수에 따라 다름
      "totalGoods1Amount": "22" // 길드원 1의 굿즈 1 기부(+사용) 총량
      "totalGoods2Amount": "22" // 길드원 1의 굿즈 2 기부(+사용) 총량
      "nickname": "고박사" // 닉네임
      "inDate": "2020-02-22T02:22:22.022Z" // 길드원 1의 길드 inDate (길드 가입 시점)
      "gamerInDate": "2020-02-22T02:22:22.022Z" // 길드원 1의 inDate
      "position": "고박사" // 직책 (master, viceMaster, member)
      "lastLogin": "2020-02-22T02:22:22.022Z" // 마지막 접속 날짜
    },
    {
      // 길드원 2
      ..
    },
    ..
  ]
}
```





# 내 길드 정보 조회

- 길드의 길드원 정보를 조회하는 GetGuildMemberList() 메소드 정의
  - BackendGuildSystem Script 수정

```
219 public void SetNotice(string notice)...
235
236 public void GetGuildMemeberList(string guildInDate)
237 {
238     Backend.Guild.GetGuildMemberListV3(guildInDate, callback =>
239     {
240         if ( !callback.IsSuccess() )
241         {
242             ErrorLog(callback.GetMessage(), "Guild_Failed_Log", "GetGuildMemberList");
243             return;
244         }
245         try...
246     });
270
271 }
272
273 public string GetGuildInfoBy(string guildName)...
298
```

try... 뒷장



# 내 길드 정보 조회

- BackendGuildSystem Script 수정 (계속)

```
246 try
247 {
248     LitJson.JsonData memberJson = callback.GetFlattenJSON()["rows"];
249
250     if ( memberJson.Count <= 0 )
251     {
252         Debug.LogWarning("불러온 길드원 데이터가 없습니다.");
253         return;
254     }
255
256     foreach ( LitJson.JsonData member in memberJson )
257     {
258         GuildMemberData guildMember = new GuildMemberData();
259
260         guildMember.position      = member["position"].ToString();
261         guildMember.nickname      = member["nickname"].ToString();
262         guildMember.goodsCount    = int.Parse(member["totalGoods1Amount"].ToString());
263         guildMember.lastLogin    = member["lastLogin"].ToString();
264     }
265 }
266 catch ( Exception e )
267 {
268     Debug.LogError(e);
269 }
```



# 내 길드 정보 조회

- 길드원 정보 출력을 제어하는 스크립트 생성 및 작성
  - C# Script 생성 후 스크립트의 이름을 "GuildMember"로 변경

```
1  using UnityEngine;
2      using TMPro;
3      using BackEnd;
4      using System;
5
6  public class GuildMember : MonoBehaviour
7  {
8      [SerializeField]
9      private TextMeshProUGUI    textPosition;
10     [SerializeField]
11     private TextMeshProUGUI    textNickname;
12     [SerializeField]
13     private TextMeshProUGUI    textGoodsCount;
14     [SerializeField]
15     private TextMeshProUGUI    textLastLogin;
16
```



# 내 길드 정보 조회

- 길드원 정보 출력을 제어하는 스크립트 생성 및 작성 (계속)

```
17 public void Setup(GuildMemberData memberData)
18 {
19     SetPosition(memberData.position);
20     SetDate(memberData.lastLogin);
21
22     textNickname.text = memberData.nickname;
23     textGoodsCount.text = memberData.goodsCount.ToString();
24 }
25
26 private void SetPosition(string position)
27 {
28     if ( position.Equals("master") )           position = "길드마스터";
29     else if ( position.Equals("viceMaster") )   position = "임원";
30     else if ( position.Equals("member") )       position = "길드원";
31
32     textPosition.text = position;
33 }
34
```





# 내 길드 정보 조회

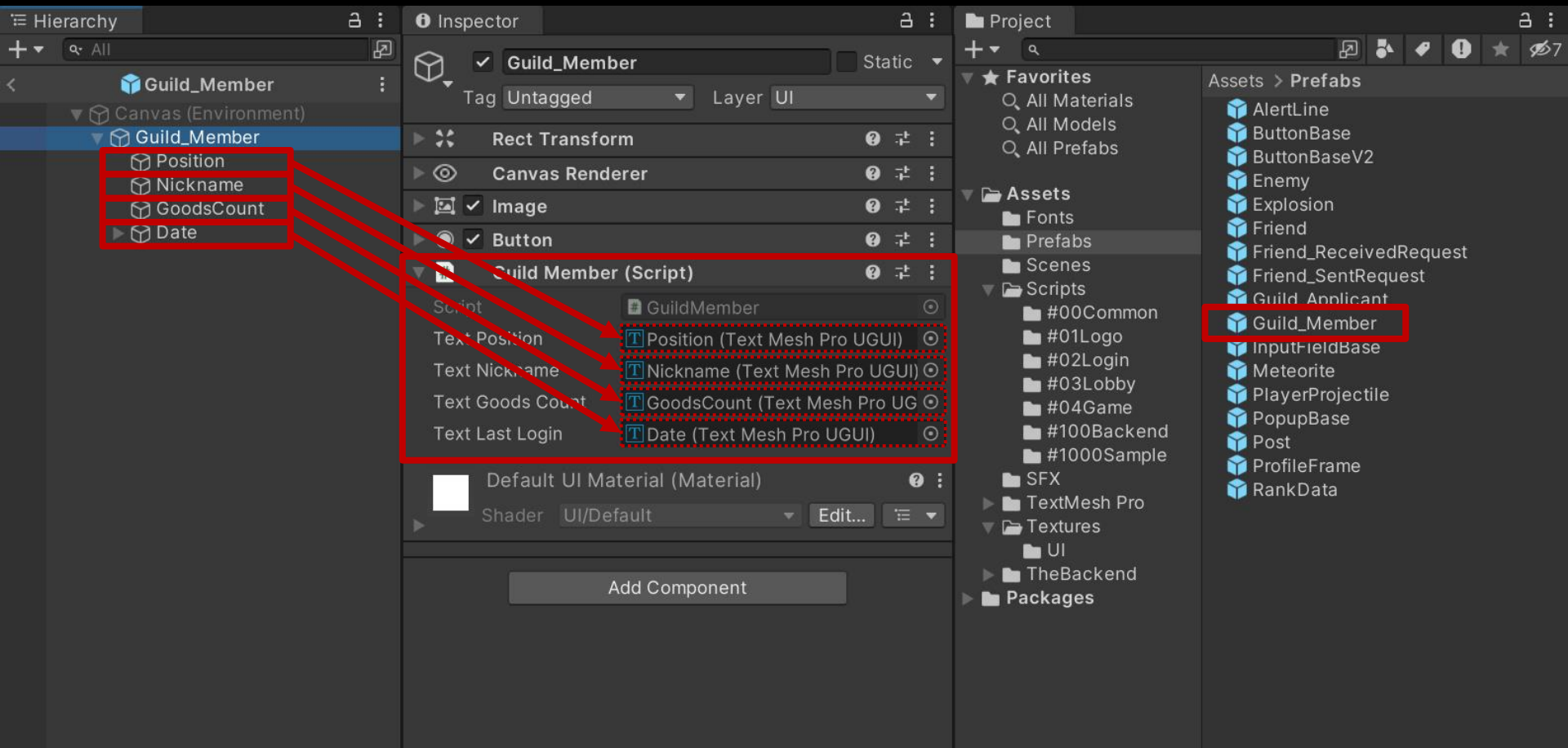
## ■ 길드원 정보 출력을 제어하는 스크립트 생성 및 작성 (계속)

```
35 private void SetDate(string lastLogin)
36 {
37     // GetServerTime() - 서버 시간 불러오기
38     Backend.Utills.GetServerTime(callback =>
39     {
40         if ( !callback.IsSuccess() )
41         {
42             Debug.LogError($"서버 시간 불러오기에 실패했습니다. : {callback}");
43             return;
44         }
45
46         // JSON 데이터 파싱 성공
47         try
48         {
49             // 현재 서버 시간
50             string serverTime = callback.GetFlattenJSON()["utcTime"].ToString();
51             // 현재 시간 - 마지막 접속 시간
52             TimeSpan timeSpan = DateTime.Parse(serverTime) - DateTime.Parse(lastLogin);
53
54             // timeSpan.TotalHours로 남은 기간을 시(hour)로 표현
55             if ( timeSpan.TotalHours < 24 ) textLastLogin.text = $"{timeSpan.TotalHours:F0}시간 전";
56             else textLastLogin.text = $"{timeSpan.TotalDays:F0}일 전";
57         }
58         // JSON 데이터 파싱 실패
59         catch ( Exception e )
60         {
61             // try-catch 에러 출력
62             Debug.LogError(e);
63         }
64     });
65 }
66 }
```



# 내 길드 정보 조회

- Guild\_Member 프리팹에 "GuildMember" 컴포넌트 추가 및 설정







# 내 길드 정보 조회

- 길드 페이지가 활성화될 때 해당 길드의 길드원 정보 조회
  - GuildPage Script 수정

```
1  +using ...
2
3
4  public class GuildPage : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private TextMeshProUGUI textGuildName;           // Popup 상단에 출력되는 길드 이름 Text UI
10     [SerializeField]
11     private Notice notice;
12     [SerializeField]
13     private GameObject executivesOption;
14     [SerializeField]
15     private TextMeshProUGUI textMemberCount;
16
17     [SerializeField]
18     private GameObject memberPrefab;
19     [SerializeField]
20     private Transform parentContent;
21
22     private string guildName = string.Empty;           // 길드 이름
23     private MemoryPool memoryPool;
24
25     private void Awake()
26     {
27         memoryPool = new MemoryPool(memberPrefab, parentContent);
28     }
29 }
```



# 내 길드 정보 조회

## □ GuildPage Script 수정 (계속)

```
30 public void Setup(string guildName, bool isMaster=false)
31 {
32     notice.Setup(isMaster);
33     executivesOption.SetActive(isMaster);
34
35     gameObject.SetActive(true);
36
37     textGuildName.text = guildName;
38     this.guildName = guildName;
39     textMemberCount.text = $"길드 인원 {backendGuildSystem.myGuildData.memberCount}/100";
40
41     backendGuildSystem.GetGuildMemeberList(backendGuildSystem.myGuildData.guildInDate);
42 }
43
44 public void Activate(GuildMemberData member)
45 {
46     GameObject item = memoryPool.ActivatePoolItem();
47     item.GetComponent<GuildMember>().Setup(member);
48 }
49
50 public void Deactivate(GameObject member)
51 {
52     memoryPool.DeactivatePoolItem(member);
53 }
54
55 public void DeactivateAll()
56 {
57     memoryPool.DeactivateAllPoolItems();
58 }
59
60 public void OnClickApplyGuild()...
64 }
```

Activate() 메소드의 이름을  
Setup()으로 변경



# 내 길드 정보 조회

- guildPage.Activate() 메소드를 guildPage.Setup()으로 변경
  - GuildDefaultPage Script 수정

```
15 public void OnClickSearchGuild()  
16 {  
17     string guildName = inputFieldGuildName.text;  
18  
19     if ( guildName.Trim().Equals("") ) ...  
23  
24     inputFieldGuildName.text = "";  
25  
26     string inDate = backendGuildSystem.GetGuildInfoBy(guildName);  
27  
28     if ( inDate.Equals(string.Empty) ) ...  
32     else  
33     {  
34         // 길드 팝업 페이지 활성화  
35         guildPage.Setup(guildName);  
36     }  
37 }  
38  
39 public void OnClickMyGuildInfo() ...  
43  
44 public void SuccessMyGuildInfo()  
45 {  
46     bool isMaster = UserInfo.Data.nickname.Equals(backendGuildSystem.myGuildData.master.nickname);  
47     guildPage.Setup(backendGuildSystem.myGuildData.guildName, isMaster);  
48 }  
49 }
```



# 내 길드 정보 조회

- PopupGuild 오브젝트의 "GuildPage" 컴포넌트 변수 설정

The screenshot displays the Unity Inspector for a **PopupGuild** object. The **Inspector** panel shows the **Inspector** tab with the following configuration:

- Script:** GuildPage
- Backend Guild System:** BackendSystem (Backend Guild)
- Text Guild Name:** Title (Text Mesh Pro UGUI)
- Notice:** Notice (Notice)
- Executives Option:** ExecutivesOption
- Text Member Count:** GuildMemberCount (Text Mesh Pro UGUI)
- Member Prefab:** Guild\_Member
- Parent Content:** Content (Rect Transform)

The **Project** window on the right shows the **Assets > Prefabs** folder containing various prefabs, with **Guild\_Member** highlighted. The **Hierarchy** window on the left shows the **PopupGuild** object selected, with **Content** highlighted under its **Content** property.





# 내 길드 정보 조회

- 길드원 정보 조회에 성공하면 guildPage.Activate()로 길드원 UI 활성화
  - BackendGuildSystem Script 수정

```
1  +using ...
5
6  -public class BackendGuildSystem : MonoBehaviour
7  {
8      [SerializeField]
9      private FadeEffect_TMP      textLog;
10     [SerializeField]
11     private GuildDefaultPage     guildDefaultPage;
12     [SerializeField]
13     private GuildCreatePage      guildCreatePage;
14     [SerializeField]
15     private GuildApplicantsPage  guildApplicantsPage;
16     [SerializeField]
17     private GuildPage            guildPage;
18
19     public GuildData              myGuildData { private set; get; } = new GuildData();
20
21     + public void CreateGuild(string guildName, int goodsCount=1)...
41
42     + public void ApplyGuild(string guildName)...
60
61     + public void GetApplicants()...
137
```



# 내 길드 정보 조회

## BackendGuildSystem Script 수정 (계속)

```
238 public void GetGuildMemeberList(string guildInDate)
239 {
240     Backend.Guild.GetGuildMemberListV3(guildInDate, callback =>
241     {
242         if ( !callback.IsSuccess() ) ...
243     }
244     try
245     {
246         LitJson.JsonData memberJson = callback.GetFlattenJSON()["rows"];
247
248         if ( memberJson.Count <= 0 ) ...
249
250         guildPage.DeactivateAll();
251
252         foreach ( LitJson.JsonData member in memberJson )
253         {
254             GuildMemberData guildMember = new GuildMemberData();
255
256             guildMember.position = member["position"].ToString();
257             guildMember.nickname = member["nickname"].ToString();
258             guildMember.goodsCount = int.Parse(member["totalGoods1Amount"].ToString());
259             guildMember.lastLogin = member["lastLogin"].ToString();
260
261             guildPage.Activate(guildMember);
262         }
263     }
264     catch ( Exception e )
265     {
266         Debug.LogError(e);
267     }
268 });
269
270
271
272
273
274
275
276
277
```





# 내 길드 정보 조회

- BackendSystem 오브젝트의 "BackendGuildSystem" 컴포넌트 변수 설정

The screenshot displays the Unity Hierarchy and Inspector panels. In the Hierarchy panel, the 'BackendSystem' object is selected under the 'Lobby\*' hierarchy. The Inspector panel shows the 'Backend Guild System (Script)' component with the following variables:

- Script: BackendGuildSystem
- Text Log: TextLog (Fade Effect\_TMP)
- Guild Default Page: DefaultPage (Guild Default Page)
- Guild Create Page: PopupCreateGuildPage (Guild C)
- Guild Applicants Page: PopupApplicants (Guild Applicar)
- Guild Page: PopupGuild (Guild Page)

A red box highlights the 'Backend Guild System (Script)' component and its variables. A red arrow points from the 'PopupGuild' component in the Hierarchy panel to the 'Guild Page' variable in the Inspector panel.



# 내 길드 정보 조회

## ■ 결과 화면

고박사 2

35 / 30 6936 10721

Enter text... 쿠폰등록

유니티노트

길드 공지사항 공지사항 등록 가입 요청 목록

안녕하세요.  
유니티노트 길드입니다.  
잘부탁드립니다.

길드 인원 2/100

길드원	닉네임	0	1시간 전
길드마스터	고박사	0	0시간 전

가입 신청

게임시작

Icons: Shop, Backpack, Sword, Leaderboard, Profile

# 특정 길드 정보 조회

- 특정 길드 정보 조회



# 특정 길드 정보 조회

## ■ 특정 길드 정보 조회

### ■ 특정 길드 정보 조회 메소드

```
GetGuildInfoV3(string guildIndate);
```

guildindate 길드 정보 조회

#### Error Cases

\* guildIndate가 존재하지 않는 경우 (404 / guild not found, ..)

\* Old guild를 조회한 경우 (412 / guild's version is different ..)



# 특정 길드 정보 조회

## ■ 특정 길드 정보 JsonData

```
{
  "guild":
  {
    "memberCount": "22" // 길드원 수
    "notice": "공지사항입니다." // 길드 메타 데이터
    "viceMasterList": // 부 길드 마스터 목록
    {
      "inDate": "2023-06-26T02:22:22.022Z" // 부 길드 마스터 1 inDate
      "nickname": "고박사" // 부 길드 마스터 1 닉네임
    },
    {
      "inDate": "2023-06-26T02:22:22.022Z" // 부 길드 마스터 2 inDate
      "nickname": "이름없음" // 부 길드 마스터 2 닉네임
    }
    "masterNickname": "DoctorKo" // 길드 마스터 닉네임
    "inDate": "2023-06-26T02:22:22.022Z" // 길드 inDate
    "guildName": "유니티노트" // 길드 이름
    "goodsCount": "1" // 길드 굿즈 개수
    "masterInDate": "2023-06-26T02:22:22.022Z" // 길드 마스터 inDate
    "_immediateRegistration": "true" // 즉시 가입 여부. SetRegistrationValueV3를 통해 설정한 경우에만 존재
    "_countryCode": "KR" // 국가 코드 (국가 코드를 추가한 경우에만 존재함)
  }
}
```



# 특정 길드 정보 조회

- guildInDate 길드의 정보를 조회하는 GetGuildInfo() 메소드 정의
  - BackendGuildSystem Script 수정

```
1  +using ...
5
6  public class BackendGuildSystem : MonoBehaviour
7  {
8      [SerializeField]
9      private FadeEffect_TMP      textLog;
10     [SerializeField]
11     private GuildDefaultPage     guildDefaultPage;
12     [SerializeField]
13     private GuildCreatePage      guildCreatePage;
14     [SerializeField]
15     private GuildApplicantsPage  guildApplicantsPage;
16     [SerializeField]
17     private GuildPage            guildPage;
18
19     public GuildData              myGuildData      { private set; get; } = new GuildData();
20     public GuildData              otherGuildData  { private set; get; } = new GuildData();
21
22     + public void CreateGuild(string guildName, int goodsCount=1) ...
42
43     + public void ApplyGuild(string guildName) ...
61
62     + public void GetApplicants() ...
138
139    + public void ApproveApplicant(string gamerInDate) ...
```





# 특정 길드 정보 조회

- BackendGuildSystem Script 수정 (계속)

```
280 public void GetGuildInfo(string guildInDate)
281 {
282     Backend.Guild.GetGuildInfoV3(guildInDate, callback =>
283     {
284         if ( !callback.IsSuccess() )
285         {
286             ErrorLog(callback.GetMessage(), "Guild_Failed_Log", "GetGuildInfo");
287             return;
288         }
289
290         try
291         {
292             LitJson.JsonData guildJson = callback.GetFlattenJSON()["guild"];
293
294             if ( guildJson.Count <= 0 )
295             {
296                 Debug.LogWarning("불러온 길드 데이터가 없습니다.");
297                 return;
298             }
299
```



# 특정 길드 정보 조회

## BackendGuildSystem Script 수정 (계속)

```
300         otherGuildData.guildName      = guildJson["guildName"].ToString();
301         otherGuildData.guildInDate    = guildJson["inDate"].ToString();
302         otherGuildData.notice         = guildJson["NOTICE"].ToString();
303         otherGuildData.memberCount    = int.Parse(guildJson["memberCount"].ToString());
304
305         otherGuildData.master          = new GuildMemberData();
306         otherGuildData.master.nickname = guildJson["masterNickname"].ToString();
307         otherGuildData.master.inDate   = guildJson["masterInDate"].ToString();
308
309         otherGuildData.viceMasterList = new List<GuildMemberData>();
310
311         LitJson.JsonData viceJson = guildJson["viceMasterList"];
312         for ( int i = 0; i < viceJson.Count; ++ i )
313         {
314             GuildMemberData vice = new GuildMemberData();
315             vice.nickname         = viceJson[i]["nickname"].ToString();
316             vice.inDate           = viceJson[i]["inDate"].ToString();
317
318             otherGuildData.viceMasterList.Add(vice);
319         }
320
321         // 길드 정보 불러오기 완료 후 처리
322         guildDefaultPage.SuccessGuildInfo();
323     }
324     catch ( Exception e )
325     {
326         Debug.LogError(e);
327     }
328 });
329 }
```



# 특정 길드 정보 조회

- 길드를 검색했을 때 backendGuildSystem.GetGuildInfo()로 길드 정보 조회
  - GuildDefaultPage Script 수정

```
1  +using ...
3
4  public class GuildDefaultPage : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private TMP_InputField inputFieldGuildName;
10     [SerializeField]
11     private GuildPage guildPage;
12     [SerializeField]
13     private FadeEffect_TMP textLog;
14
15     public void OnClickSearchGuild()
16     {
17         string guildName = inputFieldGuildName.text;
18
19         if ( guildName.Trim().Equals("") ) ...
23
24         inputFieldGuildName.text = "";
25
26         string inDate = backendGuildSystem.GetGuildInfoBy(guildName);
27
28         if ( inDate.Equals(string.Empty) ) ...
32         else
33         {
34             // 길드 램업 페이지 활성화
35             // guildPage.Setup(guildName);
36             backendGuildSystem.GetGuildInfo(inDate);
37         }
38     }
39 }
```



# 특정 길드 정보 조회

- GuildDefaultPage Script 수정 (계속)

```
40 public void OnClickMyGuildInfo()...
44
45 public void SuccessMyGuildInfo()...
50
51 public void SuccessGuildInfo()
52 {
53     bool isMaster = UserInfo.Data.nickname.Equals(backendGuildSystem.otherGuildData.master.nickname);
54     guildPage.Setup(backendGuildSystem.otherGuildData.guildName, isMaster, true);
55 }
56 }
```





# 특정 길드 정보 조회

- 본인이 소속된 길드 여부에 따라 길드 정보 출력
  - GuildPage Script 수정

```
30 public void Setup(string guildName, bool isMaster=false, bool isOtherGuild=false)
31 {
32     notice.Setup(isMaster, isOtherGuild);
33     executivesOption.SetActive(isMaster);
34
35     gameObject.SetActive(true);
36
37     textGuildName.text = guildName;
38     this.guildName = guildName;
39
40     if ( isOtherGuild == true )
41     {
42         textMemberCount.text = $"길드 인원 {backendGuildSystem.otherGuildData.memberCount}/100";
43         backendGuildSystem.GetGuildMemeberList(backendGuildSystem.otherGuildData.guildInDate);
44     }
45     else
46     {
47         textMemberCount.text = $"길드 인원 {backendGuildSystem.myGuildData.memberCount}/100";
48         backendGuildSystem.GetGuildMemeberList(backendGuildSystem.myGuildData.guildInDate);
49     }
50 }
51
```



# 특정 길드 정보 조회

- 본인이 소속된 길드 여부에 따라 공지사항 출력

- Notice Script 수정

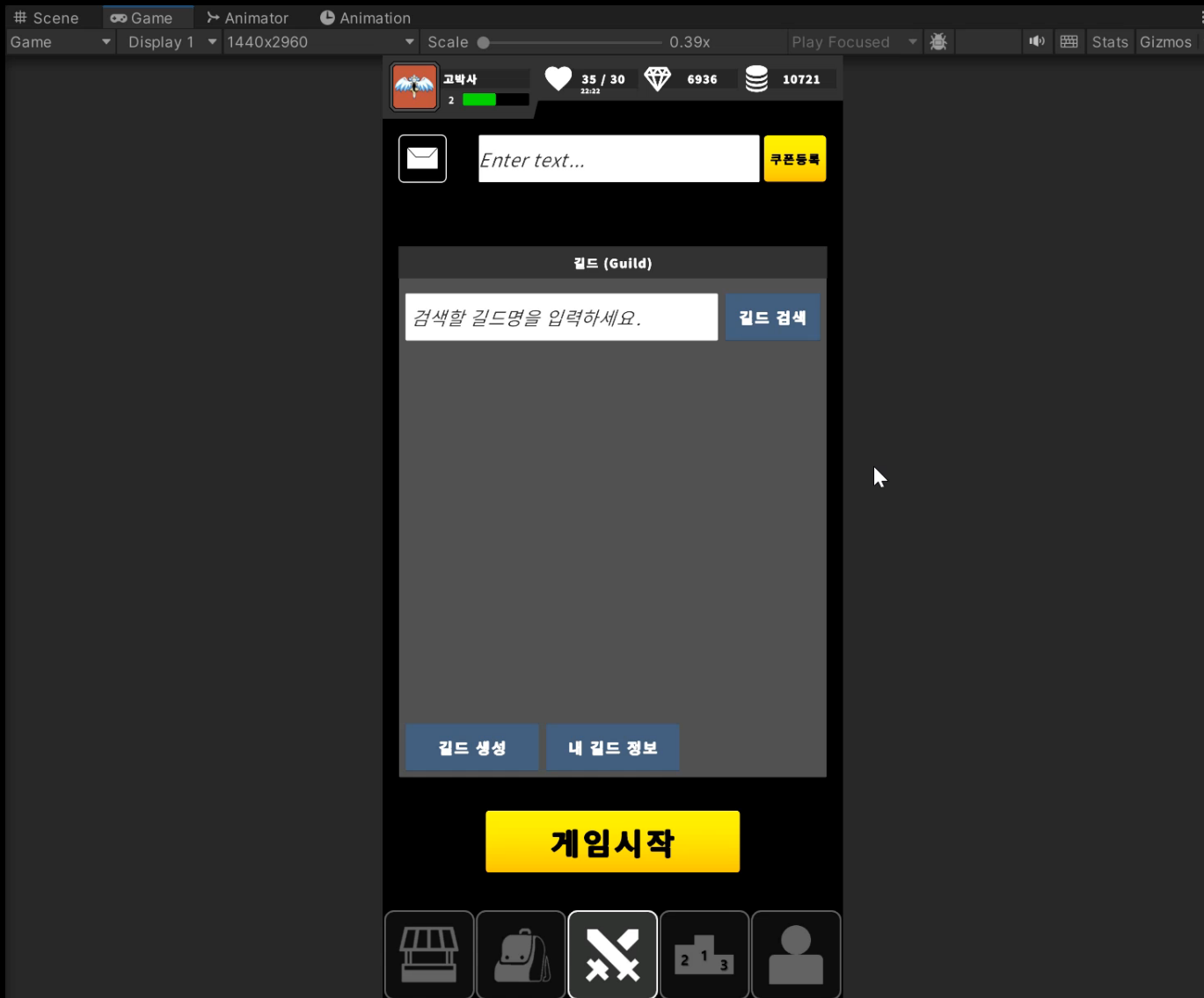
```
1  +using ...
3
4  public class Notice : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private GameObject noticeBackground; // 길드원에게 보이는 오브젝트
10     [SerializeField]
11     private TextMeshProUGUI textNotice; // 길드원에게 보이는 공지사항
12     [SerializeField]
13     private TMP_InputField inputFieldNotice; // 길드마스터에게 보이는 공지사항 오브젝트
14     [SerializeField]
15     private FadeEffect_TMP textLog;
16
17     public void Setup(bool isMaster=false, bool isOtherGuild=false)
18     {
19         if ( isOtherGuild == true )
20         {
21             textNotice.text = backendGuildSystem.otherGuildData.notice;
22             inputFieldNotice.text = backendGuildSystem.otherGuildData.notice;
23         }
24         else
25         {
26             textNotice.text = backendGuildSystem.myGuildData.notice;
27             inputFieldNotice.text = backendGuildSystem.myGuildData.notice;
28         }
29
30         noticeBackground.SetActive(!isMaster);
31         inputFieldNotice.gameObject.SetActive(isMaster);
32     }
33
34     public void OnClickNoticeRegistration()...
46 }
```





# 특정 길드 정보 조회

## ■ 결과 화면



# 랜덤 길드 정보 조회

- 랜덤 조회 생성 (Backend Console)
- 랜덤 길드 정보 등록 및 조회
- 랜덤 길드 목록 UI 제작 및 연동
- 길드 목록 새로고침
- 길드 정보 탐색



# 랜덤 길드 정보 조회

## ■ 랜덤 조회 생성 (Backend Console)

### ■ 길드 랜덤 조회 생성

#### □ 랜덤 조회 - "랜덤 조회 생성"

The screenshot shows the Backnd Console interface for 'ProjectA'. The left sidebar contains a navigation menu with the following items: 푸시 관리, 쿠폰 관리, 차트 관리, 확률 관리, 로그 관리, 영수증 검증, 길드 관리, 캐시 관리, 버전 관리, 공지사항, 이벤트, 일대일 문의, 정책 관리, and 유저 정보 찾기. The '랜덤 조회' (Random Query) item at the bottom of the sidebar is highlighted with a red box. The main content area shows the '랜덤 조회' (Random Query) page, where the '랜덤 조회 생성' (Generate Random Query) button is also highlighted with a red box. Below the button, there is a table with columns: 번호 (Number), 랜덤 조회명 (Random Query Name), UUID, 분류 (Category), and 생성일 (Creation Date). The table is currently empty, displaying the message '등록된 랜덤 조회가 없습니다.' (No registered random queries).



# 랜덤 길드 정보 조회

## □ 분류, 랜덤 조회 명 설정

The screenshot shows the Backnd Console interface. On the left is a sidebar with navigation items like '푸시 관리', '쿠폰 관리', '차트 관리', etc. The main area displays '랜덤 조회' (Random Query) management. A modal dialog titled '랜덤 조회 생성' (Create Random Query) is open. It contains the following elements:

- 분류 (Category):** A dropdown menu with '유저' (User) and '길드' (Guild) options. '길드' is selected and highlighted with a red box.
- 랜덤 조회 명\* (Random Query Name):** A text input field containing '랜덤 길드 조회' (Random Guild Query), also highlighted with a red box.
- Buttons:** '확인' (Confirm) and '취소' (Cancel) buttons at the bottom right. The '확인' button is highlighted with a red box.



# 랜덤 길드 정보 조회

## □ 생성 완료된 랜덤 조회

뒤끝 콘솔

console.thebackend.io/project/1ea3f14d34e89530ea88b3245bc82dc17d5f52ce1554049f19fce9219a847cfce18bb8891bcaf994bc66efb9a3b7878c6bc5513c1dd300af990c59a6edd3f2b94037...

Backnd Console ProjectA 관리자

푸시 관리  
쿠폰 관리  
차트 관리  
확률 관리  
로그 관리  
영수증 검증  
길드 관리  
캐시 관리  
버전 관리  
공지사항  
이벤트  
일대일 문의  
정책 관리  
유저 정보 찾기  
랜덤 조회  
실시간  
뒤끝챗  
realtimeChatting Preview  
뒤끝매치

★ 랜덤 조회 랜덤 조회 생성 삭제 SDK 문서 콘솔 가이드

번호	랜덤 조회명	UUID	분류	생성일
1	랜덤 길드 조회	6a675fc0-3084-11ee-99ec-b1646f9340c5	길드	2023.08.02 00:59



# 랜덤 길드 정보 조회

## ■ 랜덤 길드 정보 등록 및 조회

### ■ 길드 랜덤 조회의 UUID 변수 선언

#### □ Constants Script 수정

```
1 public static class Constants
2 {
3     public static readonly string USER_DATA_TABLE = "USER_DATA";
4     public static readonly string DAILY_RANK_UUID = "0fa14490-085a-11ee-b506-7d5eaba96eff";
5     public static readonly int MAX_RANK_LIST = 20;
6
7     public static readonly string LEVEL_CHART = "82683";
8
9     public static readonly string GOODS_CHART_NAME = "재화차트";
10
11     public static readonly int EXPIRATION_DAYS = 3;
12
13     public static readonly string RANDOM_GUILD_UUID = "6a675fc0-3084-11ee-99ec-b1646f9340c5";
14 }
```





# 랜덤 길드 정보 조회

## ■ 랜덤 그룹 데이터 삽입 메소드

```
SetRandomData(RandomType randomType, string uuid, int value);
```

```
SetRandomData(RandomType randomType, string uuid, long value);
```

```
SetRandomData(RandomType randomType, string uuid, float value);
```

```
SetRandomData(RandomType randomType, string uuid, double value);
```

랜덤 조회 그룹에 데이터 삽입.

하나의 랜덤 조회 그룹에 각 유저(길드) 당 하나의 데이터만 존재할 수 있다.

스코어 외에 추가적인 값의 삽입은 불가능하다.

길드 유형 랜덤 조회의 경우 길드에 가입된 유저만 삽입이 가능하며 미가입 유저가 삽입을 시도하면 에러 발생

### Error Cases

\* 입력한 uuid가 null 혹은 string.Empty일 경우 (400 / uuid is null or empty)

\* 랜덤 조회의 유형이 다를 경우 (400 / bad type is not match, ..)

\* (랜덤 조회 유형 : 길드) 길드에 가입되어 있지 않을 경우 (404 / guild not found, ..)

\* 존재하지 않는 uuid일 경우 (404 / randomPool not found, ..)



# 랜덤 길드 정보 조회

## ■ 랜덤 그룹 데이터 삽입 메소드 (계속)

Value	Type	설명
randomType	RandomType	랜덤 조회 유형 (유저 : RandomType.User, 길드 : RandomType.Guild)
uuid	string	랜덤 조회의 uuid
value	int/long/float/double	삽입하고자 하는 값 (소수점, 음수도 가능)(long.MaxValue 이상일 경우 조회시 에러 발생)



# 랜덤 길드 정보 조회

## ■ 랜덤 그룹 데이터 조회 메소드

`GetRandomData(RandomType randomType, string uuid, int pivot, int gap, int count);`

해당 uuid에 존재하는 데이터들을 조건(pivot, gap, count)에 맞게 랜덤으로 불러옴

- 랜덤 조회에 등록된 점수가 10000, 1000, 100, 10, 1 순으로 되어있고 pivot이 100, gap이 1이면, 99~101점인 데이터를 랜덤 검색하는 것이 아니라, 100점이 속한 3등을 기준으로 2등(1000점)과 4등(10점) 사이에서 랜덤으로 검색합니다.
- 만약 조건에 만족하는 데이터가 count만큼 존재하지 않을 경우, count보다 적은 데이터를 불러올 수 있습니다.
- gap은 최소 0에서 최대 10까지 입력할 수 있으며, 해당 범위를 넘어갈 경우 최소/최대값이 재설정되어 검색됩니다.
- count는 최소 1에서 최대 10까지 입력할 수 있으며, 해당 범위를 넘어갈 경우 최소/최대값이 재설정되어 검색됩니다.
- pivot은 long의 최댓값(9223372036854775807, 1.0E+17 미만)까지 입력할 수 있으며, 해당 값을 넘어갈 경우 최소/최대값이 재설정되어 검색됩니다.
- 닉네임은 제공되지 않으며, 유저/길드의 inDate만 제공됩니다.

### Error Cases

- \* 입력한 uuid가 null 혹은 string.Empty일 경우 (400 / uuid is null or empty)
- \* 랜덤 조회의 유형이 다를 경우 (400 / bad type is not match, ..)
- \* 존재하지 않는 uuid일 경우 (404 / randomPool not found, ..)



# 랜덤 길드 정보 조회

## ■ 랜덤 그룹 데이터 조회 메소드 (계속)

Value	Type	설명	min	max
randomType	RandomType	랜덤 조회 유형 (유저 : RandomType.User, 길드 : RandomType.Guild)		
uuid	string	랜덤 조회의 uuid		
pivot	int/long/float/double	중심이 되는 값	-long.MaxValue	long.MaxValue
gap	int	pivot에 일치하거나 제일 가까운 값의 순위를 기준으로 위아래 등수의 범위	0	10
count	int	불러올 개수	1	10



# 랜덤 길드 정보 조회

- 랜덤 그룹 데이터 조회 JsonData (RandomType.User)

```
{
  "rows":
  [
    {
      "gamerInDate": "2020-02-22T02:22:22.022Z" // 유저 InDate
      "score": "22" // 유저 점수
    },
    {
      "gamerInDate": "2020-02-22T02:22:22.022Z" // 유저 InDate
      "score": "38" // 유저 점수
    },
    ..
  ]
}
```





# 랜덤 길드 정보 조회

- 랜덤 그룹 데이터 조회 JsonData (RandomType.Guild)

```
{
  "rows":
  [
    {
      "guildName": "유니티노트"           // 길드 이름
      "guildInDate": "2020-02-22T02:22:22.022Z" // 길드 InDate
      "score": "22"                       // 길드 점수
    },
    {
      "guildName": "unitynote"           // 길드 이름
      "guildInDate": "2020-02-22T02:22:22.022Z" // 길드 InDate
      "score": "38"                       // 길드 점수
    },
    ..
  ]
}
```



# 랜덤 길드 정보 조회

- 길드를 생성할 때 랜덤 조회 테이블에 길드 등록, 조회
  - BackendGuildSystem Script 수정

```
22 public void CreateGuild(string guildName, int goodsCount=1)
23 {
24     Backend.Guild.CreateGuildV3(guildName, goodsCount, callback =>
25     {
26         if ( !callback.IsSuccess() ) ...
32
33         Debug.Log($"길드가 생성되었습니다. : {callback}");
34
35         // 길드를 생성할 때 "길드 공지사항입니다."를 공지사항으로 기본 설정
36         SetNotice("길드 공지사항입니다.");
37
38         // 길드 생성에 성공했을 때 호출
39         guildCreatePage.SuccessCreateGuild();
40
41         Backend.RandomInfo.SetRandomData(RandomType.Guild, Constants.RANDOM_GUILD_UUID, 0, callback =>
42         {
43             if ( !callback.IsSuccess() )
44             {
45                 ErrorLog(callback.GetMessage(), "Guild_Failed_Log", "CreateGuild():SetRandomData");
46
47                 return;
48             }
49
50             Debug.Log($"생성한 길드를 길드 랜덤 조회 목록에 추가하였습니다. : {callback}");
51         });
52     });
53 }
54
```



# 랜덤 길드 정보 조회

- BackendGuildSystem Script 수정 (계속)

```
343 public void GetRandomGuildList()
344 {
345     Backend.RandomInfo.GetRandomData(RandomType.Guild, Constants.RANDOM_GUILD_UUID, 0, 0, 20, callback =>
346     {
347         if ( !callback.IsSuccess() )
348         {
349             ErrorLog(callback.GetMessage(), "Guild_Failed_Log", "GetRandomGuildList");
350
351             return;
352         }
353
354         try
355         {
356             LitJson.JsonData guildJson = callback.GetFlattenJSON()["rows"];
357
358             if ( guildJson.Count <= 0 )
359             {
360                 Debug.LogWarning("불러온 랜덤 길드 데이터가 없습니다.");
361                 return;
362             }
363
```



# 랜덤 길드 정보 조회

## BackendGuildSystem Script 수정 (계속)

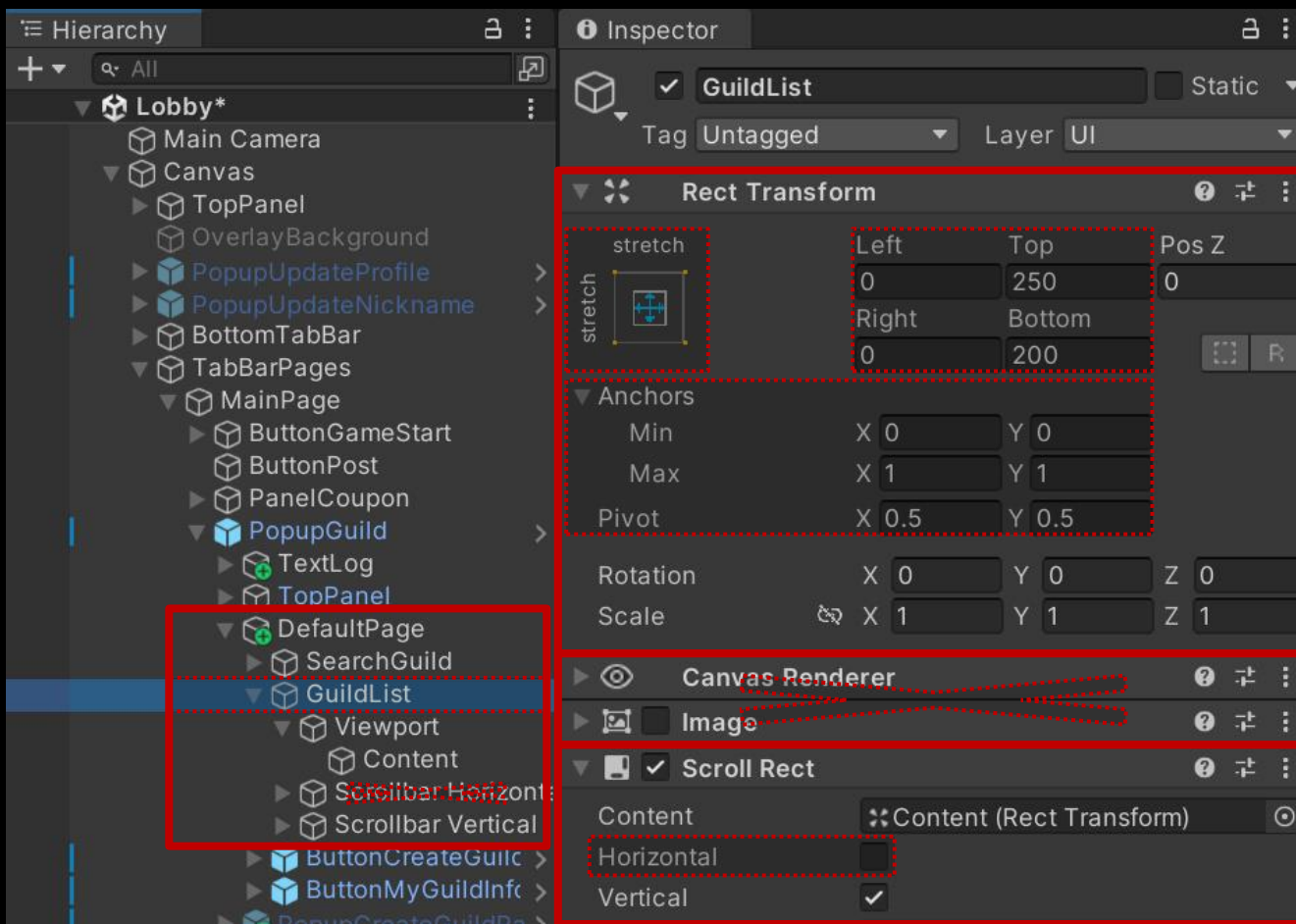
```
364     foreach ( LitJson.JsonData guild in guildJson )
365     {
366         Backend.Guild.GetGuildInfoV3(guild["guildInDate"].ToString(), callback =>
367         {
368             if ( !callback.IsSuccess() )
369             {
370                 ErrorLog(callback.GetMessage(), "Guild_Failed_Log", "GetRandomGuildList():GetGuildInfoV3");
371                 return;
372             }
373
374             LitJson.JsonData guildJson = callback.GetFlattenJSON()["guild"];
375
376             if ( guildJson.Count <= 0 )
377             {
378                 Debug.LogWarning("불러온 길드 데이터가 없습니다.");
379                 return;
380             }
381
382             GuildData guildData = new GuildData();
383             guildData.guildName = guildJson["guildName"].ToString();
384             guildData.master = new GuildMemberData();
385             guildData.master.nickname = guildJson["masterNickname"].ToString();
386             guildData.memberCount = int.Parse(guildJson["memberCount"].ToString());
387         });
388     }
389 }
390 catch ( Exception e )
391 {
392     Debug.LogError(e);
393 }
394 });
395 }
```



# 랜덤 길드 정보 조회

## ■ 랜덤 길드 목록 UI 제작 및 연동

- 가입 가능한 임의의 길드 목록을 출력하는 Scroll View UI 생성 및 설정
  - GameObject - UI - Scroll View







# 랜덤 길드 정보 조회

- Scrollbar가 보이지 않도록 Width를 0으로 설정

The image shows the Unity development environment. On the left is the Hierarchy panel, and on the right is the Inspector panel.

**Hierarchy Panel:** Shows a tree view of the scene hierarchy. The 'Lobby\*' folder is expanded, showing various UI elements. The 'Scrollbar Vertical' component is highlighted with a red box.

**Inspector Panel:** Shows the properties of the selected 'Scrollbar Vertical' component. The 'Rect Transform' component is expanded, and the 'Width' property is highlighted with a red dashed box, showing a value of 0. Other properties like 'Pos X', 'Top', 'Pos Z', 'Bottom', 'Anchors', 'Pivot', 'Rotation', and 'Scale' are also visible.

Property	Value
Pos X	0
Top	0
Pos Z	0
Width	0
Bottom	17
Min X	1
Min Y	0
Max X	1
Max Y	1
Pivot X	1
Pivot Y	1
Rotation X	0
Rotation Y	0
Rotation Z	0
Scale X	1
Scale Y	1
Scale Z	1



# 랜덤 길드 정보 조회

- Content 오브젝트에 컴포넌트 추가 및 설정

The screenshot displays the Unity Inspector interface for a 'Content' object. The Hierarchy panel on the left shows the object's position within the scene hierarchy, with 'Content' highlighted. The Inspector panel on the right shows the following configuration:

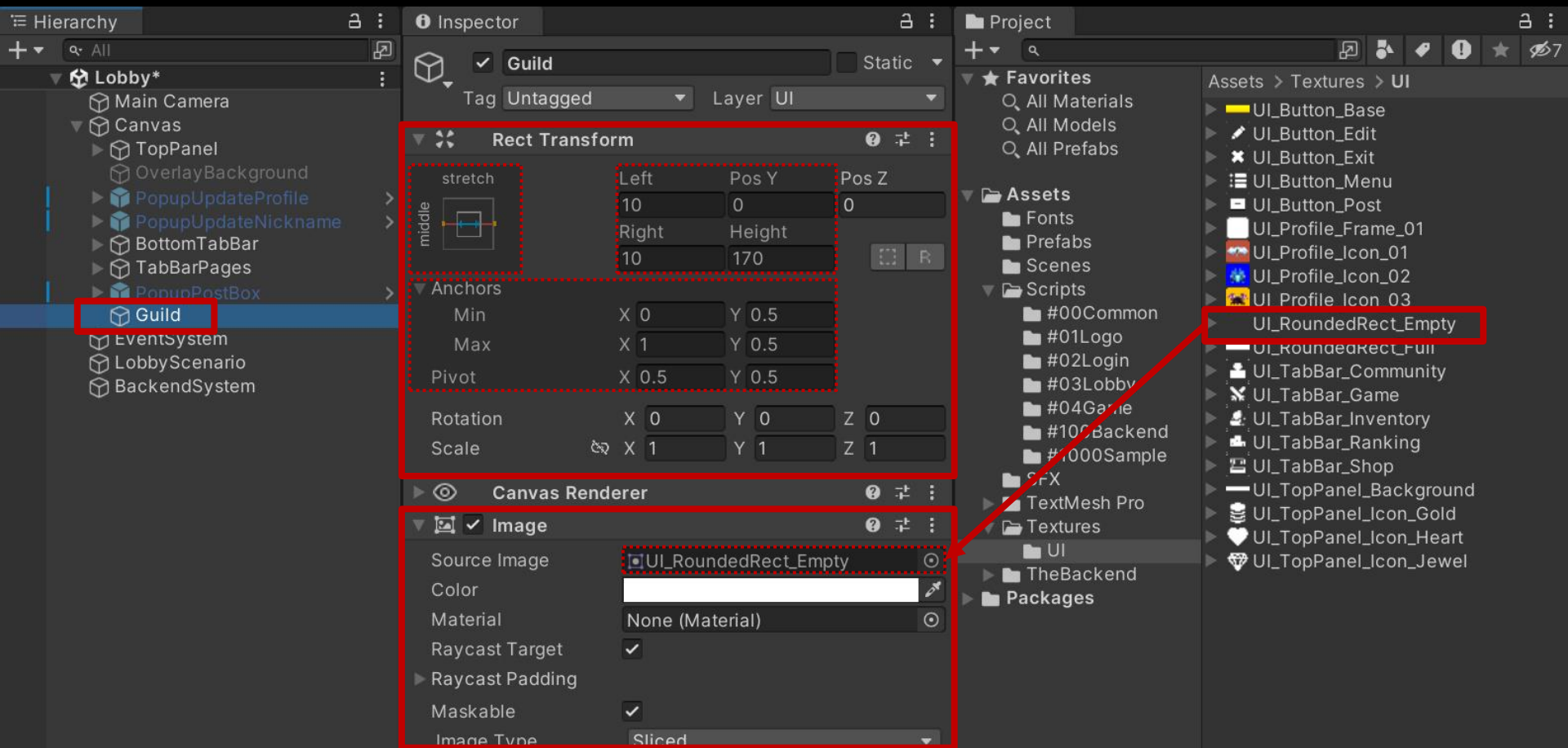
- Content** component: Tag: Untagged, Layer: UI.
- Rect Transform** component: Expanded to show **Vertical Layout Group** settings:
  - Padding: Left (20), Right (20), Top (20), Bottom (20), Spacing (20).
  - Child Alignment: Upper Center.
  - Reverse Arrangement:
  - Control Child Size:  Width,  Height.
  - Use Child Scale:  Width,  Height.
  - Child Force Expand:  Width,  Height.
- Content Size Fitter** component: Horizontal Fit: Unconstrained, Vertical Fit: Preferred Size.

An 'Add Component' button is visible at the bottom of the Inspector panel.



# 랜덤 길드 정보 조회

- 길드 정보를 출력하는 Image UI 생성 및 설정
  - GameObject - UI - Image





# 랜덤 길드 정보 조회

- 길드 이름을 출력하는 "Text - TextMeshPro" UI 생성 및 설정
  - GameObject - UI - "Text - TextMeshPro"

Hierarchy: Lobby\* > Guild > GuildName

Inspector: GuildName (Static, Tag: Untagged, Layer: UI)

Rect Transform:

left	Pos X	Top	Pos Z
stretch	10	10	0
	Width	Bottom	
	500	10	

Anchors:

Min	X	Y
	0	0
Max	X	Y
	0	1

Pivot: X 0, Y 0.5

Rotation: X 0, Y 0, Z 0

Scale: X 1, Y 1, Z 1

Canvas Renderer

TextMeshPro - Text (UI)

NotoSansKR-Bold SDF Material (Material)

Shader: TextMeshPro/Distance Fi

Add Component

Text Input: Enable RTL Editor

Text Style: Normal

Main Settings:

Font Asset: NotoSansKR-Bold SDF (TMP\_Fc)

Material Preset: NotoSansKR-Bold SDF Material

Font Style: B I U S ab AB SC

Font Size: 50

Auto Size:

Vertex Color:

Color Gradient:

Override Tags:

Spacing Options (em): Character 0, Word 0, Line 0, Paragraph 0

Alignment:  Center

Wrapping: Enabled

Overflow: Overflow

Horizontal Mapping: Character

Vertical Mapping: Character



# 랜덤 길드 정보 조회

- 길드마스터 닉네임을 출력하는 "Text - TextMeshPro" UI 생성 및 설정
  - GuildName 오브젝트를 Ctrl+D로 복제

Hierarchy: Lobby\* > Canvas > TopPanel > OverlayBackground > PopupUpdateProfile > PopupUpdateNickname > BottomTabBar > TabBarPages > PopupPostBox > Guild > GuildName > MasterNickname

Inspector: MasterNickname (Static) | Tag: Untagged | Layer: UI

Rect Transform: left, stretch, Pos X: 520, Top: 10, Pos Z: 0, Width: 300, Bottom: 10

Anchors: Min (X: 0, Y: 0), Max (X: 0, Y: 1), Pivot (X: 0, Y: 0.5)

Rotation: X: 0, Y: 0, Z: 0

Scale: X: 1, Y: 1, Z: 1

Canvas Renderer

TextMeshPro - Text (UI)

NotoSansKR-Bold SDF Material (Material) | Shader: TextMeshPro/Distance F...

Add Component

Text Input: Enable RTL Editor [ ]

고박사

Text Style: Normal

Main Settings

Font Asset: NotoSansKR-Bold SDF (TMP\_Fc)

Material Preset: NotoSansKR-Bold SDF Material

Font Style: B I U S ab AB SC

Font Size: 50

Auto Size: [ ]

Vertex Color: [ ]

Color Gradient: [ ]

Override Tags: [ ]

Spacing Options (em): Character 0, Word 0, Line 0, Paragraph 0

Alignment: [Left] [Center] [Right] [Justified] [Full]

Wrapping: Enabled

Overflow: Overflow

Horizontal Mapping: Character

Vertical Mapping: Character





# 랜덤 길드 정보 조회

- 길드 인원수를 출력하는 "Text - TextMeshPro" UI 생성 및 설정
  - GuildName 오브젝트를 Ctrl+D로 복제

Hierarchy

- Lobby\*
  - Main Camera
  - Canvas
    - TopPanel
    - OverlayBackground
    - PopupUpdateProfile
    - PopupUpdateNickname
    - BottomTabBar
    - TabBarPages
    - PopupPostBox
    - Guild**
      - GuildName
      - MasterNickname
      - MemberCount
    - EventSystem
    - LobbyScenario
    - BackendSystem

Inspector

MemberCount

Tag Untagged Layer UI

Rect Transform

right	Pos X	Top	Pos Z
	-170	10	0
stretch	Width	Bottom	
	280	10	

Anchors

Min	X 1	Y 0
Max	X 1	Y 1
Pivot	X 1	Y 0.5

Rotation

X	0	Y 0	Z 0
---	---	-----	-----

Scale

X	1	Y 1	Z 1
---	---	-----	-----

Canvas Renderer

TextMeshPro - Text (UI)

NotoSansKR-Bold SDF Material (Material)

Shader TextMeshPro/Distance Fi

Add Component

Text Input

Enable RTL Editor

1/100

Text Style Normal

Main Settings

Font Asset NotoSansKR-Bold SDF (TMP\_Fc)

Material Preset NotoSansKR-Bold SDF Material

Font Style B I U S ab AB SC

Font Size 50

Auto Size

Vertex Color

Color Gradient

Override Tags

Spacing Options (em) Character 0 Word 0 Line 0 Paragraph 0

Alignment

Wrapping Enabled

Overflow Overflow

Horizontal Mapping Character

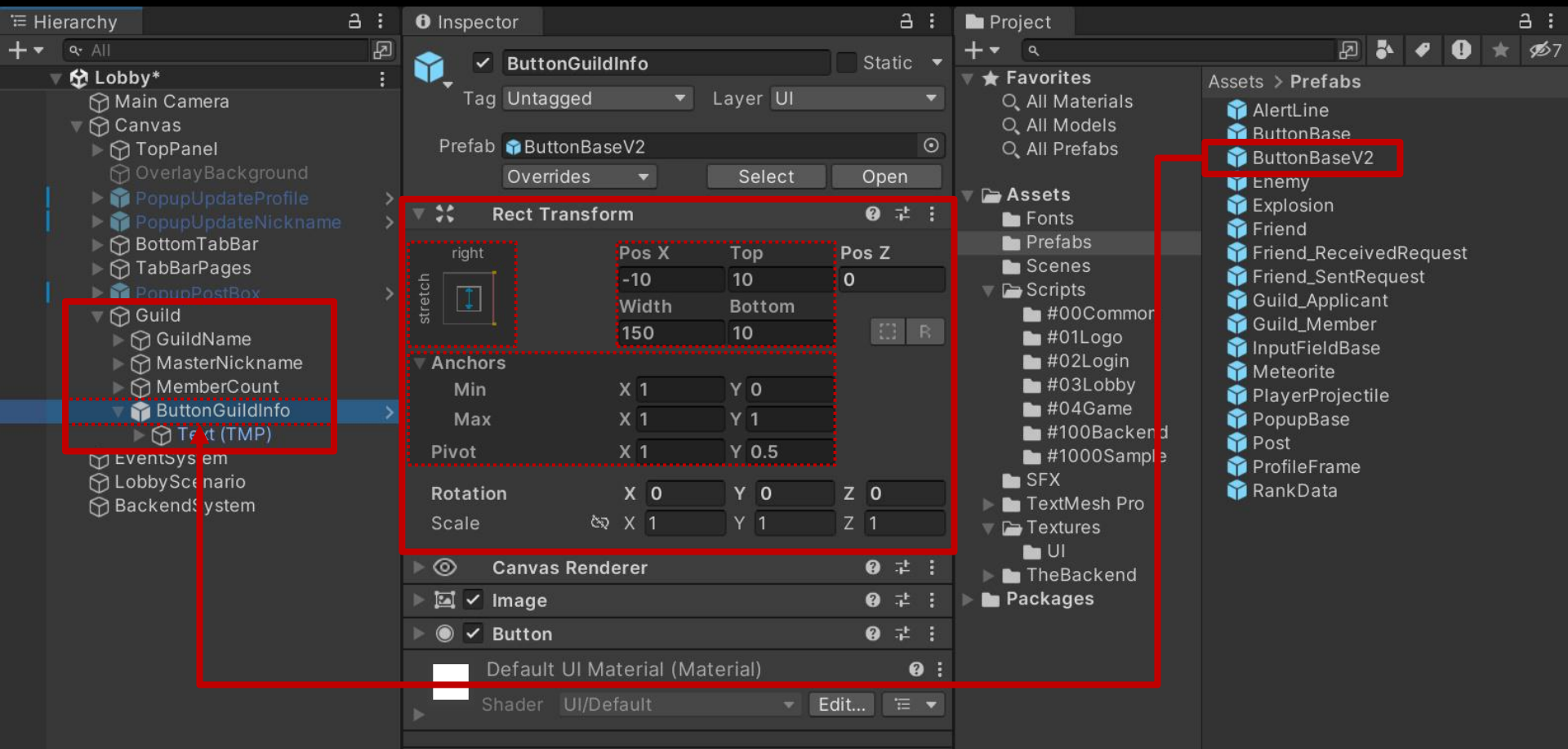
Vertical Mapping Character





# 랜덤 길드 정보 조회

- “길드정보” Button UI 생성 및 설정
  - ButtonBaseV2 프리팹을 Hierarchy View로 Drag & Drop





# 랜덤 길드 정보 조회

- “길드정보” Button UI 생성 및 설정

The image shows the Unity development environment. On the left is the Hierarchy panel, and on the right is the Inspector panel.

**Hierarchy Panel:**

- Expanded: Lobby\*
- Expanded: Canvas
  - Expanded: Guild
    - Expanded: ButtonGuildInfo
      - Text (TMP)** (Selected)

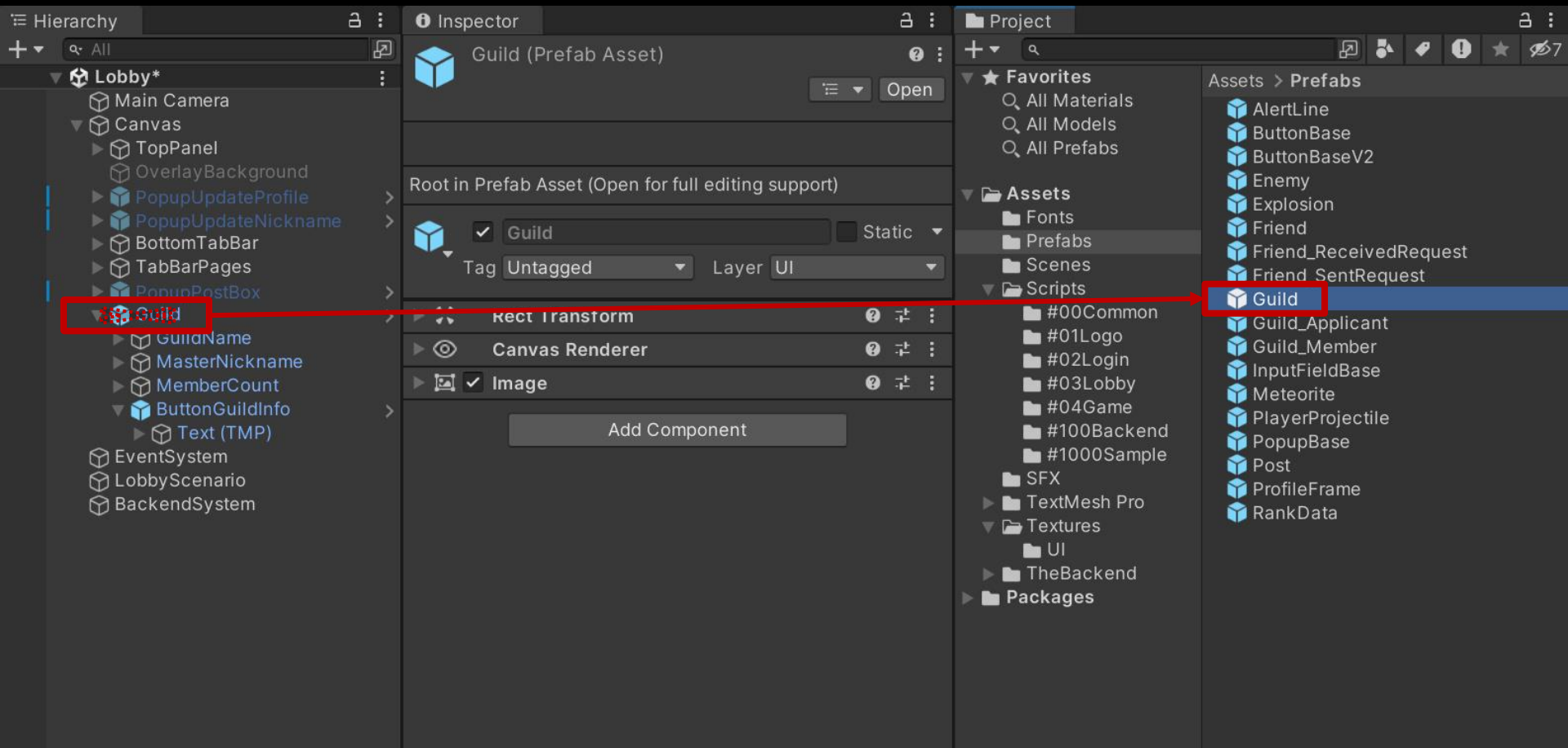
**Inspector Panel:**

- Component: **Text (TMP)** (Static: unchecked)
- Tag: Untagged, Layer: UI
- Component: **TextMeshPro - Text (UI)** (Selected)
- Text Input: Enable RTL Editor (unchecked)
- Text: 길드 정보
- Text Style: Normal
- Main Settings
  - Font Asset: NotoSansKR-Bold SDF (TMP\_Fc)
  - Material Preset: NotoSansKR-Bold SDF Material
  - Font Style: B I U S ab AB SC
  - Font Size: 40** (Selected)
  - Auto Size: unchecked
  - Vertex Color: [Color Picker]
  - Color Gradient: unchecked



# 랜덤 길드 정보 조회

- Guild 오브젝트 Prefab 생성
  - Hierarchy View의 "Guild" 오브젝트를 Project View로 Drag & Drop
  - Hierarchy View에 있는 "Guild" 오브젝트 삭제





# 랜덤 길드 정보 조회

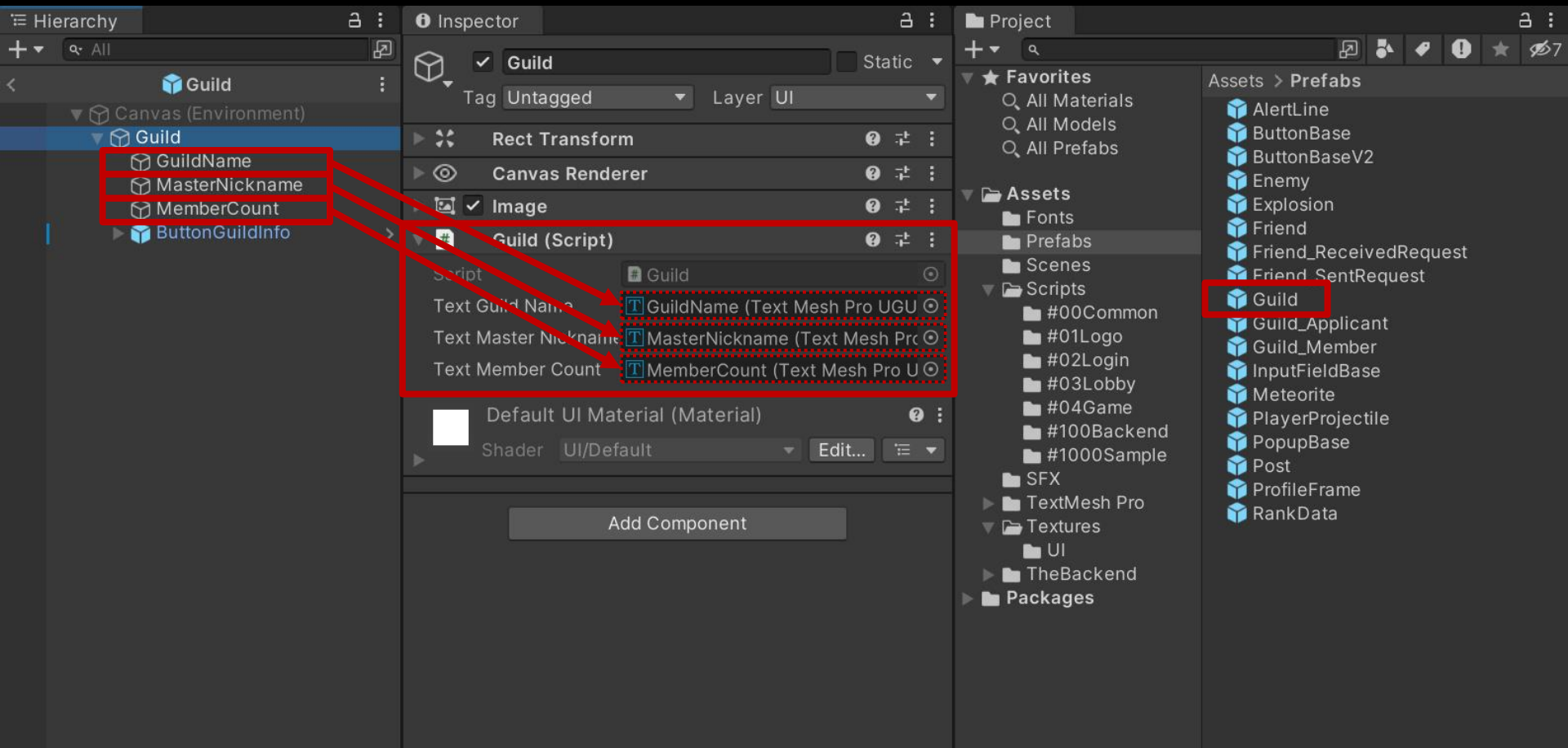
- 길드 정보 출력을 제어하는 스크립트 생성 및 작성
  - C# Script 생성 후 스크립트의 이름을 "Guild"로 변경

```
1  using UnityEngine;
2  using TMPro;
3
4  public class Guild : MonoBehaviour
5  {
6      [SerializeField]
7      private TextMeshProUGUI    textGuildName;
8      [SerializeField]
9      private TextMeshProUGUI    textMasterNickname;
10     [SerializeField]
11     private TextMeshProUGUI    textMemberCount;
12
13     public void Setup(GuildData guildData)
14     {
15         textGuildName.text      = guildData.guildName;
16         textMasterNickname.text = guildData.master.nickname;
17         textMemberCount.text    = $"{guildData.memberCount}/100";
18     }
19 }
```



# 랜덤 길드 정보 조회

- Guild 프리팹에 "Guild" 컴포넌트 추가 및 설정







# 랜덤 길드 정보 조회

- 랜덤 길드 정보를 조회하고, UI로 출력하는 메소드 정의
  - GuildDefaultPage Script 수정

```
1  using ...
3
4  public class GuildDefaultPage : MonoBehaviour
5  {
6      [SerializeField]
7      private BackendGuildSystem backendGuildSystem;
8      [SerializeField]
9      private TMP_InputField inputFieldGuildName;
10     [SerializeField]
11     private GuildPage guildPage;
12     [SerializeField]
13     private FadeEffect_TMP textLog;
14
15     [SerializeField]
16     private GameObject guildPrefab;
17     [SerializeField]
18     private Transform parentContent;
19
20     private MemoryPool memoryPool;
21
22     private void Awake()
23     {
24         memoryPool = new MemoryPool(guildPrefab, parentContent);
25
26        OnClickRefresh();
27     }
28
```





# 랜덤 길드 정보 조회

## □ GuildDefaultPage Script 수정 (계속)

```
29      + public void OnClickSearchGuild()...
51
52      + public void OnClickMyGuildInfo()...
56
57      + public void SuccessMyGuildInfo()...
62
63      + public void SuccessGuildInfo()...
68
69      - public void OnClickRefresh()
70      {
71          backendGuildSystem.GetRandomGuildList();
72      }
73
74      - public void Activate(GuildData guild)
75      {
76          GameObject item = memoryPool.ActivatePoolItem();
77          item.GetComponent<Guild>().Setup(guild);
78      }
79
80      - public void DeactivateAll()
81      {
82          memoryPool.DeactivateAllPoolItems();
83      }
84      }
```



# 랜덤 길드 정보 조회

- DefaultPage 오브젝트의 "GuildDefaultPage" 컴포넌트 변수 설정

The screenshot displays the Unity Inspector window for the 'DefaultPage' object. The 'Inspector' panel shows the 'Guild Default Page (Script)' component with the following variable assignments:

- Script: GuildDefaultPage
- Backend Guild System: BackendSystem (Backend Guild)
- Input Field Guild Name: InputFieldBase (TMP\_Input Field)
- Guild Page: PopupGuild (Guild Page)
- Text Log: TextLog (Fade Effect\_TMP)
- Guild Prefab: Guild (highlighted with a red dashed box)
- Parent Content: Content (Rect Transform) (highlighted with a red dashed box)

Red arrows point from the 'Guild' and 'Content (Rect Transform)' entries in the Inspector to their respective locations in the Hierarchy and Project panels. In the Hierarchy panel, the 'DefaultPage' object and its 'Content' child are highlighted with red boxes. In the Project panel, the 'Assets > Prefabs' folder is expanded, and the 'Guild' prefab is highlighted with a red box.



# 랜덤 길드 정보 조회

## ■ 길드 랜덤 정보를 조회했을 때 길드 UI 활성화

### □ BackendGuildSystem Script 수정

```
343 public void GetRandomGuildList()
344 {
345     Backend.RandomInfo.GetRandomData(RandomType.Guild, Constants.RANDOM_GUILD_UUID, 0, 0, 20, callback =>
346     {
347         if ( !callback.IsSuccess() )...
353
354         try
355         {
356             LitJson.JsonData guildJson = callback.GetFlattenJSON()["rows"];
357
358             if ( guildJson.Count <= 0 )...
363
364             guildDefaultPage.DeactivateAll();
365
366             foreach ( LitJson.JsonData guild in guildJson )
367             {
368                 Backend.Guild.GetGuildInfoV3(guild["guildInDate"].ToString(), callback =>
369                 {
370                     if ( !callback.IsSuccess() )...
375
376                     LitJson.JsonData guildJJson = callback.GetFlattenJSON()["guild"];
377
378                     if ( guildJJson.Count <= 0 )...
383
384                     GuildData guildData = new GuildData();
385                     guildData.guildName = guildJJson["guildName"].ToString();
386                     guildData.master = new GuildMemberData();
387                     guildData.master.nickname = guildJJson["masterNickname"].ToString();
388                     guildData.memberCount = int.Parse(guildJJson["memberCount"].ToString());
389
390                     guildDefaultPage.Activate(guildData);
391                 });
392             }
393         }
394         catch ( Exception e )
395         {
396             Debug.LogError(e);
397         }
398     });
399 }
```



# 랜덤 길드 정보 조회

## ■ 결과 화면


user04
30 / 30
0
0



Enter text...

쿠폰등록

길드 (Guild)

검색할 길드명을 입력하세요. 길드 검색

유니티노트	고박사	1/100	길드 정보
unitynote	닉네임	1/100	길드 정보
unity	user03	1/100	길드 정보

길드 생성
내 길드 정보

게임시작



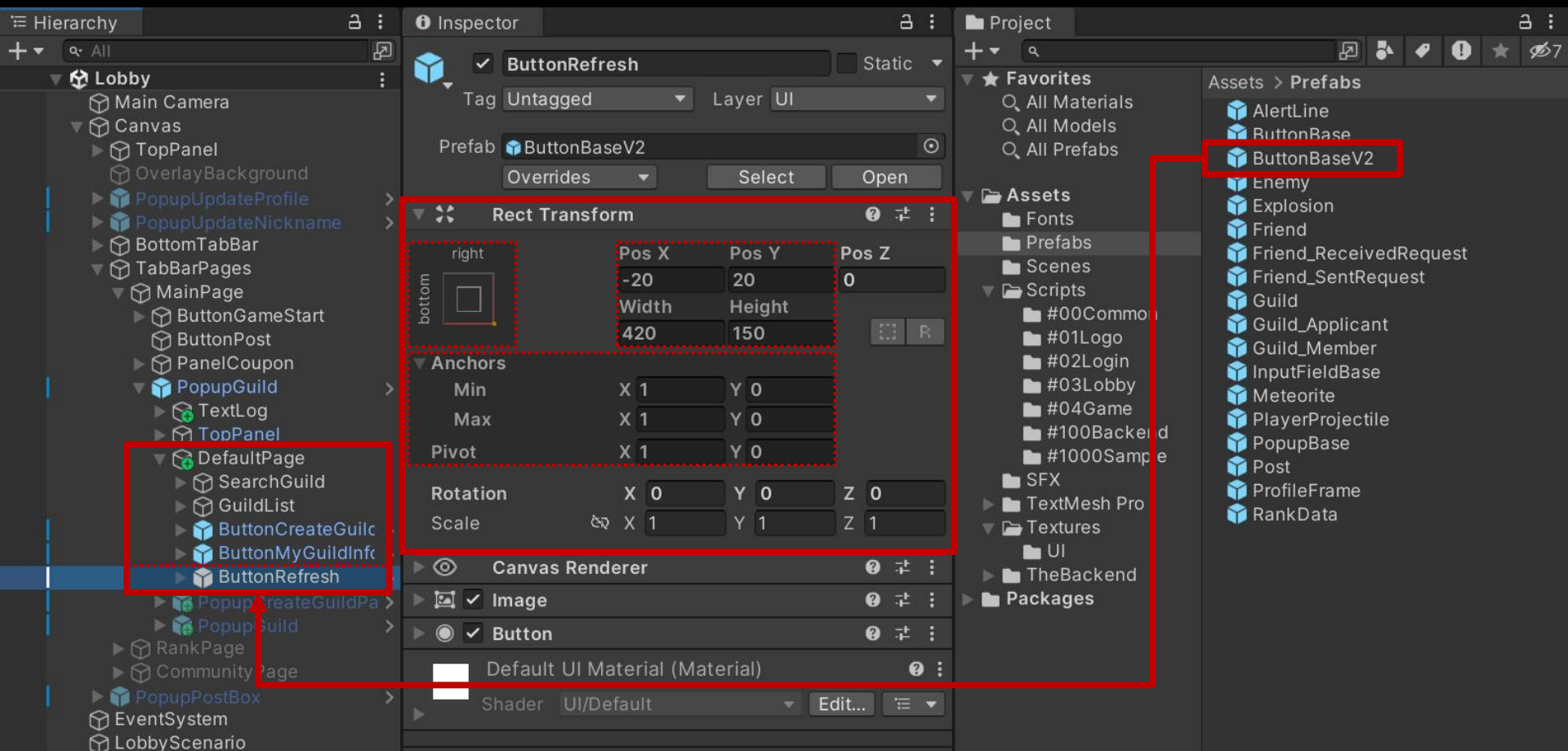


# 랜덤 길드 정보 조회

## ■ 길드 목록 새로고침

### ■ 새로고침 Button UI 생성 및 설정

#### □ ButtonBaseV2 프리팹을 Hierarchy View로 Drag & Drop







# 랜덤 길드 정보 조회

- 새로그침 Button UI 생성 및 설정 (계속)

The image shows the Unity development environment. On the left is the Hierarchy panel, and on the right is the Inspector panel.

**Hierarchy Panel:** Shows a tree view of the scene objects. The 'Lobby\*' folder is expanded, showing 'Main Camera', 'Canvas', and 'TabBarPages'. Under 'Canvas', there are several UI elements. Under 'TabBarPages', there is a 'MainPage' folder. Inside 'MainPage', there are several buttons and panels. The 'Text (TMP)' object is highlighted with a red box.

**Inspector Panel:** Shows the properties of the selected 'Text (TMP)' object. The 'TextMeshPro - Text (UI)' component is selected. The text input field contains the Korean text '새로그침'. The 'Text Style' is set to 'Normal'. The 'Main Settings' section shows the 'Font Asset' is 'NotoSansKR-Bold SDF (TMP\_Fc)', the 'Material Preset' is 'NotoSansKR-Bold SDF Material', the 'Font Style' is 'B I U S ab AB SC', and the 'Font Size' is '50'. The 'Vertex Color' is set to white.





# 랜덤 길드 정보 조회

- ButtonRefresh 오브젝트의 "Button" 컴포넌트 onClick() 이벤트 등록

The image shows the Unity Hierarchy and Inspector panels. The Hierarchy panel on the left displays the scene structure, with 'Lobby\*' expanded to show various UI elements. 'ButtonRefresh' is selected under 'MainPage'. The Inspector panel on the right shows the properties of the selected 'Button' component. The 'On Click ()' event list at the bottom of the Inspector panel is highlighted with a red box, showing a single event: 'Runtime Only' with a dropdown menu set to 'GuildDefaultPage.OnClickRefres' and a target object 'DefaultPa'. A red arrow points from the 'DefaultPage' object in the Hierarchy panel to the 'DefaultPa' target in the event list.



# 랜덤 길드 정보 조회

## ■ 결과 화면

길드 (Guild)

검색할 길드명을 입력하세요. 길드 검색

유니티노트	고박사	1/100	길드 정보
-------	-----	-------	-------

길드 생성    내 길드 정보    새로고침

길드 (Guild)

검색할 길드명을 입력하세요. 길드 검색

unitynote	닉네임	1/100	길드 정보
-----------	-----	-------	-------

길드 생성    내 길드 정보    새로고침

길드 (Guild)

검색할 길드명을 입력하세요. 길드 검색

unity	user03	1/100	길드 정보
-------	--------	-------	-------

길드 생성    내 길드 정보    새로고침

“새로고침” 버튼을 누를 때마다 현재 길드 랜덤 조회에 등록되어 있는 3개의 길드 중 임의의 길드가 출력된다.

길드가 3개 밖에 없기 때문에 GetRandomData()의 count를 1로 설정하고 테스트함

게임시작

게임시작

게임시작





# 랜덤 길드 정보 조회

## ■ 길드 정보 탐색

- 길드정보 버튼을 눌러 길드를 탐색하기 위해 guildInDate 저장

- BackendGuildSystem Script 수정

```
343 public void GetRandomGuildList()
344 {
345     Backend.RandomInfo.GetRandomData(RandomType.Guild, Constants.RANDOM_GUILD_UUID, 0, 0, 20, callback =>
346     {
347         if ( !callback.IsSuccess() ) {...
353
354         try
355         {
356             LitJson.JsonData guildJson = callback.GetFlattenJSON()["rows"];
357
358             if ( guildJson.Count <= 0 ) {...
363
364             guildDefaultPage.DeactivateAll();
365
366             foreach ( LitJson.JsonData guild in guildJson )
367             {
368                 Backend.Guild.GetGuildInfoV3(guild["guildInDate"].ToString(), callback =>
369                 {
370                     if ( !callback.IsSuccess() ) {...
375
376                     LitJson.JsonData guildJson = callback.GetFlattenJSON()["guild"];
377
378                     if ( guildJson.Count <= 0 ) {...
383
384                     GuildData guildData = new GuildData();
385                     guildData.guildName = guildJson["guildName"].ToString();
386                     guildData.guildInDate = guildJson["inDate"].ToString();
387                     guildData.master = new GuildMemberData();
388                     guildData.master.nickname = guildJson["masterNickname"].ToString();
389                     guildData.memberCount = int.Parse(guildJson["memberCount"].ToString());
```



# 랜덤 길드 정보 조회

- “길드정보” 버튼을 눌렀을 때 호출하는 메소드 정의

- Guild Script 수정

```
1  +using ...
3
4  public class Guild : MonoBehaviour
5  {
6      [SerializeField]
7      private TextMeshProUGUI    textGuildName;
8      [SerializeField]
9      private TextMeshProUGUI    textMasterNickname;
10     [SerializeField]
11     private TextMeshProUGUI    textMemberCount;
12
13     private BackendGuildSystem  backendGuildSystem;
14     private GuildData           guildData;
15
16     public void Setup(BackendGuildSystem guildSystem, GuildData guildData)
17     {
18         backendGuildSystem    = guildSystem;
19         this.guildData        = guildData;
20
21         textGuildName.text    = guildData.guildName;
22         textMasterNickname.text = guildData.master.nickname;
23         textMemberCount.text  = $"{guildData.memberCount}/100";
24     }
25
26     public void OnClickGuildInfo()
27     {
28         backendGuildSystem.GetGuildInfo(guildData.guildInDate);
29     }
30 }
```



# 랜덤 길드 정보 조회

- Guild.Setup() 메소드 매개변수로 backendGuildSystem 전달
  - GuildDefaultPage Script 수정

```
22  + private void Awake()...
28
29  + public void OnClickSearchGuild()...
51
52  + public void OnClickMyGuildInfo()...
56
57  + public void SuccessMyGuildInfo()...
62
63  + public void SuccessGuildInfo()...
68
69  + public void OnClickRefresh()...
73
74  - public void Activate(GuildData guild)
75  {
76      GameObject item = memoryPool.ActivatePoolItem();
77      item.GetComponent<Guild>().Setup(backendGuildSystem, guild);
78  }
79
80  + public void DeactivateAll()...
84  }
```





# 랜덤 길드 정보 조회

- ButtonGuildInfo 오브젝트의 "Button" 컴포넌트 onClick 이벤트 설정

The screenshot displays the Unity Inspector interface for a **ButtonGuildInfo** object. The **Button** component is selected, and its **On Click** event is configured as follows:

- Runtime Only** (dropdown menu)
- Guild.OnClickGuildInfo** (script reference dropdown menu)
- Guild (Gui)** (object reference dropdown menu)

Red boxes and arrows highlight the **Button** component and the **On Click** event configuration. The **Assets > Prefabs** panel on the right shows the **Guild** prefab selected.





# 랜덤 길드 정보 조회

## ■ 결과 화면

The screenshot shows a game interface with two panels. The left panel displays search results for a guild named 'unity'. The right panel shows a guild join request for 'unity' from 'user04'.

**Left Panel: Guild Search Results**

Header: **길드 (Guild)**

Search bar: *검색할 길드명을 입력하세요.* **길드 검색**

unity	user03	1/100	길드 정보
유니티노트	고박사	1/100	길드 정보
unitynote	닉네임	1/100	길드 정보

Buttons: **길드 생성** **내 길드 정보** **새로고침**

**게임시작**

**Right Panel: Guild Join Request**

Header: **unity**

Buttons: **길드 공지사항** **공지사항 등록** **가입 요청 목록**

Text: *길드 공지사항입니다.*

Header: **길드 가입 요청 목록**

	<b>user04</b>	수락	거절
	Lv. 1		

Button: **가입 신청**

**게임시작**

Bottom navigation bar: