

PlayFab 캐릭터

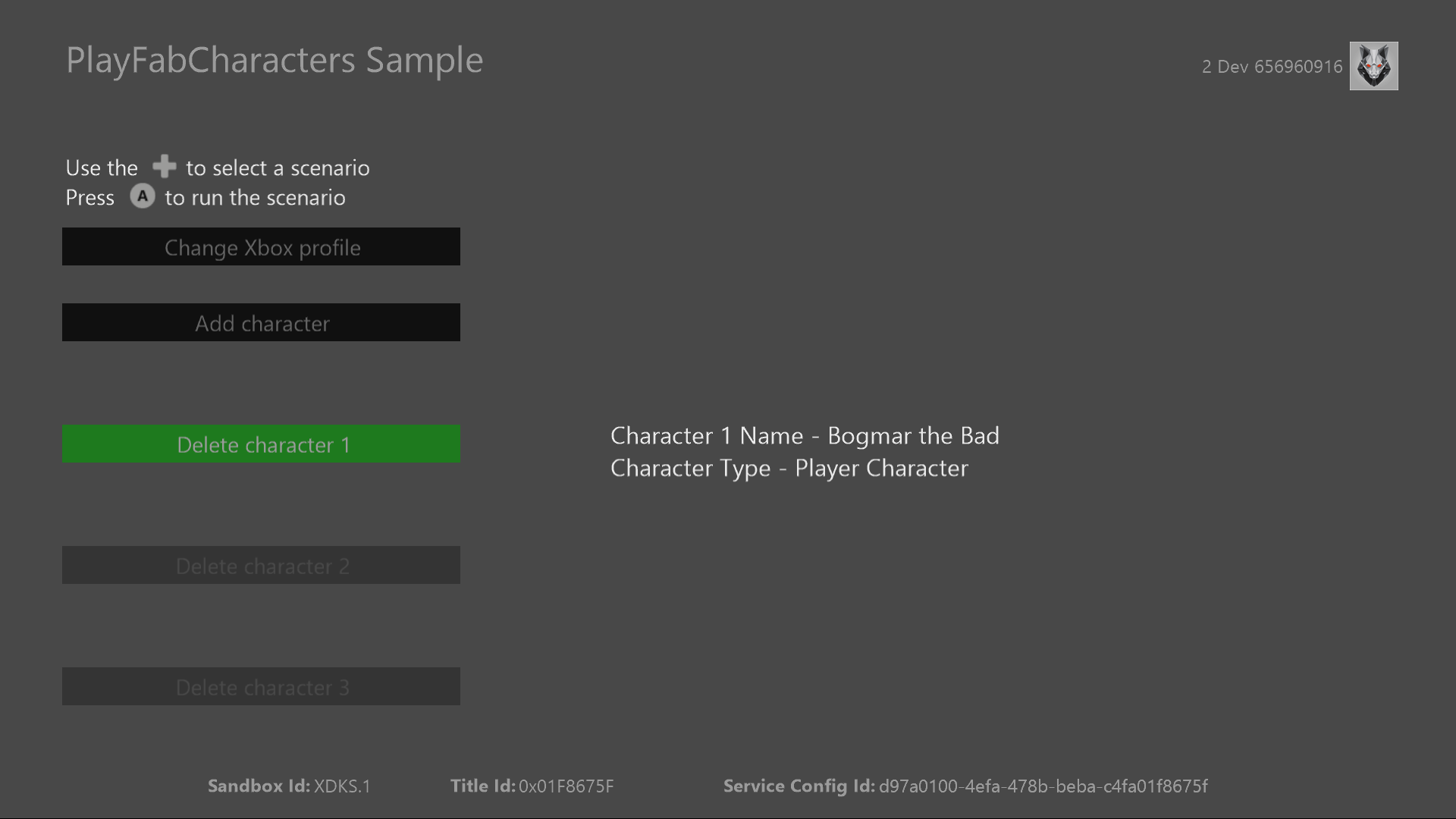
*\*이 샘플은 2018년 4월 XDK와 호환됩니다.*

# 설명

이 샘플에서는 플레이어의 계정에 문자를 부여하거나 삭제하는 방법을 보여줍니다.

# 샘플 사용하기

이 샘플은 XDKS.1 샌드 박스에서 작동하도록 구성되었습니다.



|  |  |
| --- | --- |
| 액션 | 게임 패드 |
| 작업 선택 | D 패드 |
| 액션 실행 | A 버튼 |
| 종료 | 보기 버튼 |

# 구현 정보

이 버전의 XDK에서 샘플은 Xbox One 콘솔에서 XDKS.1 **SandboxID**을 사용해야 합니다. 이것을 전환하려면 **SandboxID**, Xbox One XDK 명령 프롬프트에서 다음을 사용하십시오.

xbconfig sandboxid=XDKS.1

PlayFab 타이틀 ID는 샘플 초기화 코드에서 설정해야 합니다. 이 샘플에서는 PlayFab 제목 ID "DC0"을 사용합니다. **PlayFabSettings::titleId** 정적 필드.

# PlayFab 캐릭터

PlayFab의 캐릭터는 플레이어의 프로필을 확장한 것입니다. 각 캐릭터는 고유한 인벤토리와 자신의 통계 및 사용자 정의 데이터를 가지고 있습니다. 각 캐릭터는 인벤토리에 있는 유료 아이템과 같이 중요한 정보와 연관될 수 있기 때문에 이를 부여하고 삭제하려면 클라이언트 코드의 트랜잭션 또는 보안 서버의 실행이 필요합니다.

이 샘플에서는 간단한 PlayFab 클라우드 스크립트를 사용하여 플레이어의 문자를 문의하고 요청시 이를 부여 및 삭제합니다. 문자를 부여하고 삭제하는 데 사용되는 클라우드 스크립트는 다음과 같습니다.

handlers.grantCharacter = function (args, context) {

var request = {

PlayFabId: currentPlayerId, CharacterName: args.Name, CharacterType: "플레이어 캐릭터"

};

server.GrantCharacterToUser(request);

};

handlers.deleteCharacter = function (args, context) {

var request = {

PlayFabId: currentPlayerId, CharacterId: args.CharId, SaveCharacterInventory: false

};

server.DeleteCharacterFromUser(request);

};

플레이어에게 캐릭터를 부여하려면 플레이어의 표시 이름과 비슷한 문자 이름과 문자 유형이 필요합니다. 플레이어가 생성하려는 캐릭터의 이름을 결정하면 이름은 **Name** 인수로 스크립트에서 통과하여 **GrantCharacterToUser** 호출에 사용됩니다.

**알림** 고유한 이름은 서비스에 의해 시행되지 않습니다. 고유한 이름의 문자가 필요한 경우 제목 코드에 적용되어야 합니다.

캐릭터가 생성되면 사용자의 PlayFab ID와 유사한 고유 한 식별자를 부여합니다. 이 ID는 **DeleteCharacterFromUser**에 대하여 필수 항목입니다. 이 샘플은 플레이어가 삭제하고자 하는 캐릭터를 선택할 수 있게 한 다음 해당 캐릭터 ID가 **CharId** 인수로 클라우드 스크립트에 전달됩니다.

# 업데이트 기록

**초판:** 2018년 6월

# 개인정보처리방침

샘플을 컴파일하고 실행할 때 샘플 실행 파일의 이름이 Microsoft로 보내져 샘플 사용을 추적 할 수 있습니다. 이 데이터 수집을 거부하려면 Main.cpp에서 "샘플 사용 텔레메트리"라는 코드 블록을 제거하면 됩니다.

Microsoft의 개인 정보 취급 방침에 대한 일반적인 내용은 [Microsoft 개인 정보 취급 방침](https://privacy.microsoft.com/ko-kr/privacystatement/)을 참조하십시오.