

Google Real-time notification Notification

Document

Playstore 에서 발생한 이벤트를 전달받는 기능이다.

https://developer.android.com/google/play/billing/realtime_developer_notifications?hl=ko

Google Cloud Pub/Sub

Google Cloud 에서 제공하는 Pub/Sub 을 이용한 메시지 수신을 확인한다.

<https://cloud.google.com/pubsub/docs?hl=ko>

기본 개념

안내 동영상

- 토픽(주제): 게시자가 메시지를 전송하는 리소스
- 구독: 특정 토픽의 메시지가 구독 클라이언트로 전달되는 리소스
- 메시지: 게시자가 토픽에 전달하여 구독자에게 전달되는 데이터와 속성의 조합
- 메시지 속성: 게시자가 메시지에 대해 정의할 수 있는 K-V.

Subscription

구독에 대한 내용을 정리한다.

전송 횟수

메시지 저장소에 저장된 모든 메시지는 최소 1회 이상 전송된다. 단, 아래와 같은 경우에는 전송 대상에서 제외된다.

- 저장된지 7일이 지난 메시지는 메시지 저장소에서 자동 삭제되기에 전송이 불가능하다.
 - 저장 기간은 게시자가 설정이 가능하다(10분 ~ 7일)
- 구독이 설정되기 전에 생성된 메시지는 받을 수 없다.

구독 방법

게시자가 게시한 메시지를 구독하기 위한 방법은 아래와 같다.

- 가져오기 구독
- 내보내기 구독

가져오기 구독

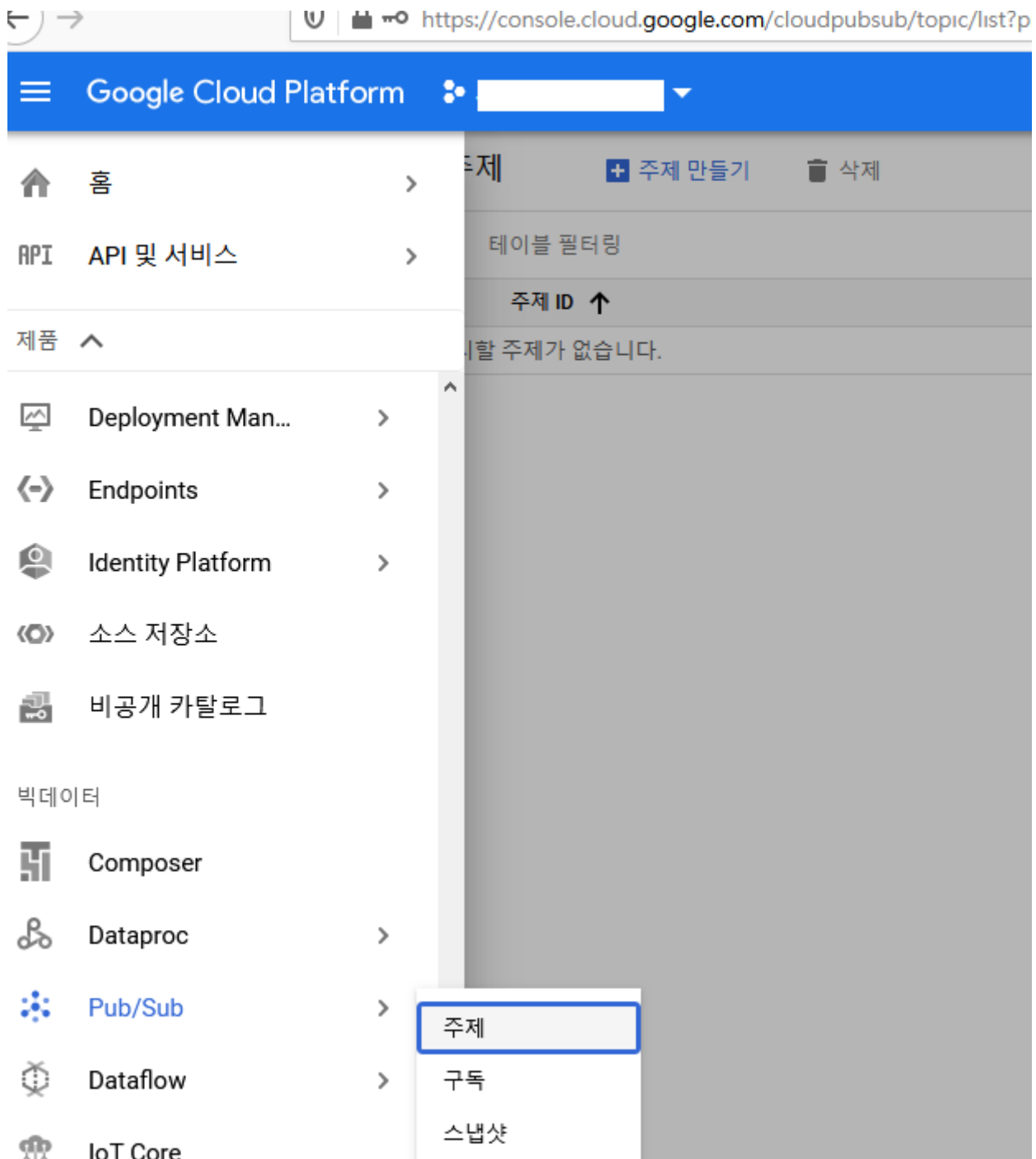
<https://cloud.google.com/pubsub/docs/pull?hl=ko>

내보내기 구독

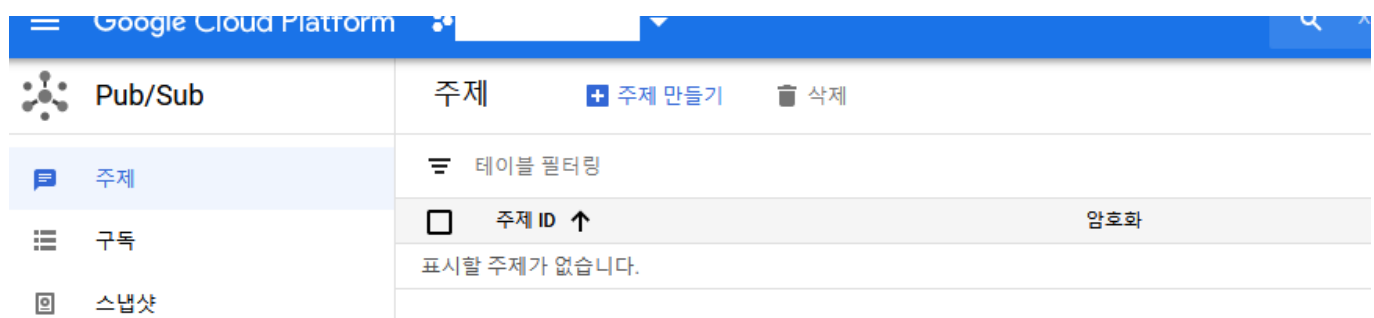
<https://cloud.google.com/pubsub/docs/push?hl=ko>

설정

Google Cloud Pub/Sub 설정



- Google Cloud Console 에 접속 후 빅데이터 > Pub/Sub 메뉴로 이동한다.



- 상단의 "주제 만들기"를 클릭하여 주제를 생성한다.

주제 만들기

주제는 게시자의 메시지를 구독자에게 전달합니다.

주제 ID *

ID는 필수 항목입니다.

암호화

☒ Google 관리 키
구성이 필요하지 않음

☐ 고객 관리 키
Google Cloud Key Management Service를 통해 관리

취소 주제 만들기

- 주제 ID 를 작성하여 하단의 "주제 만들기" 버튼을 클릭하여 생성한다.
 - 주제는 각 게임별로 생성하며 주제 ID 의 규칙은 아래와 같다.
 - iap-{game-name}

주제 + 주제 만들기 삭제 정보 패널 표시

테이블 필터링

주제 ID ↑	암호화	주제 이름	라벨
iap-sample	Google 관리	projects/	/topics/iap-sample

구독 만들기
스냅샷 만들기
다음 위치에서 가져오기
다음 위치에 내보내기
Cloud 함수 트리거
라벨 업데이트
권한 보기

BigQuery
Cloud Storage 텍스트 파일
Cloud Storage Avro 파일

- 생성된 주제의 맨 우측의 "more" 버튼을 클릭하여 권한 보기를 선택한다.

iap-sample

권한

라벨

스토리지 정책

활동

아래 권한을 수정 또는 삭제하거나 '구성원 추가'로 새 권한을 부여합니다.

+

구성원 추가

상속된 권한 표시

≡

필터 트리

?

역할/구성원 ↑

상속

▼

소유자 (1)

- "구성원 추가" 버튼을 클릭한다.

'iap-sample'에 구성원 추가

'iap-sample' 리소스에 구성원 및 역할 추가

아래에 구성원을 하나 이상 입력하고 구성원의 역할을 선택해 리소스에 대한 액세스 권한을 부여하세요. 여러 역할을 지정할 수 있습니다. [자세히 알아보기](#)

새 구성원

google-play-developer-notifications@system.gserviceaccount.com

×

?

역할

게시/구독 게시자

▼

특정 주제에 메시지를 게시합니다.

+

다른 역할 추가

☐

알림 이메일 전송

'iap-sample'의 이 역할에 대한 액세스 권한을 부여받은 구성원에게 알림 이메일이 전송됩니다.

저장

취소

- 새 구성원에 "google-play-developer-notifications@system.gserviceaccount.com" 을 지정하고, 역할에는 "게시/구독 게시자" 를 선택하고 저장한다.

☰

Google Cloud Platform

⚙️

▼

🌐

Pub/Sub

🗨️

주제

☰

구독

🖼️

스냅샷

구독

+

구독 만들기

🗑️ 삭제

☰

테이블 필터링

☐

구독 ID ↑

전송

표시할 구독이 없습니다.

- 좌측의 "구독" 메뉴를 선택 후 "구독 만들기" 버튼을 클릭한다.

←

구독 만들기

페이지(으)로 이동

구독 서비스는 구독자에게 주제에 대한 메시지를 전달합니다. 구독자에게 즉시 메시지를 푸시하거나, 구독자가 필요할 때 메시지를 가져올 수 있습니다.

구독 ID *

iap-sample

?

구독 이름: projects/ /subscriptions/iap-sample

Cloud Pub/Sub 주제 선택 *

projects, /topics/iap-sample

▼

전송 유형 ?

☐ 가져오기

☒ 푸시

엔드포인트 URL *

?

☐ 인증 사용 설정

[Learn more](#)

구독 만료 ?

☐ 다음 기간 동안 비활성 상태이면 만료(최대 365일)

열려 있는 연결, 활성 가져오기, 완료된 푸시 등의 구독자 활동이 없으면 구독이 비활성 상태가 됩니다.

☒ 만료되지 않음

활동 여부에 관계없이 구독이 만료되지 않습니다.

확인 기한 ?

기한은 10초에서 600초 사이입니다.

60

초

5 / 12

- 구독 ID 는 주제 ID 와 동일하게 설정한다.
- Cloud Pub/Sub 주제 선택
 - 앞서 생성한 주제 ID 를 선택한다.
- 전송 유형 선택
 - 푸시 선택하고 엔드포인트 URL 을 설정한다.
- 구독 만료 유형
 - 만료되지 않음 선택
- 확인 기한
 - 구독을 수신하고 수신 여부를 전달받는 시간으로 넉넉하게 60초로 설정한다.

구독 필터링 베타

필터 구문을 제공하면 구독자가 필터와 일치하는 메시지만 수신합니다.

[자세히 알아보기](#)

예: 'attributes: k' 또는 'attributes.k1 = "v" AND NOT hasPrefix(attributes.k2, "v")'

180자(영문 기준) 이하여야 합니다. 적용한 후에는 필터를 변경하거나 삭제할 수 없습니다.

메시지 보관 기간 ?

기간은 10분에서 7일 사이입니다.

일

7

▼

시간

0

▼

분

0

▼

☐ 확인한 메시지 보관 ?

사용 설정하면 확인된 메시지가 위에 지정된 메시지 보관 기간 동안 유지됩니다. 이로 인해 메시지 스토리지 수수료가 증가합니다. [자세히 알아보기](#)

데드 레터 처리

☐ 데드 레터 처리 사용

구독에 최대 전송 시도 횟수를 지정할 수 있습니다. 메시지를 전송할 수 없으면 지정된 데드 레터 주제에 다시 게시됩니다.

재시도 정책

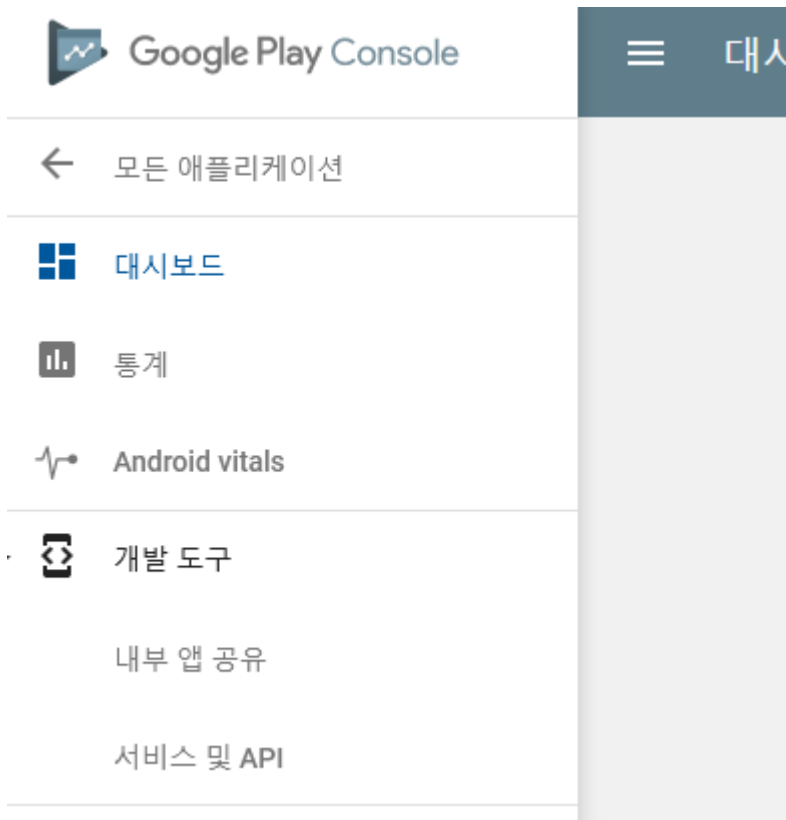
Retry policy will be triggered on NACKs or acknowledgement deadline exceeded events for a given message. [Learn more](#)

- ☒ 즉시 재시도
- ☐ 지수 백오프 지연 후 재시도

CREATE

- 메시지 보관 기간
 - 7일로 설정한다.
 - 만약 해당 메시지를 수신하였다는 응답을 받지 못하면 7일 내에 다시금 전달된다.

Play Console 설정



- Google Play Console 에 접속하고, 설정할 게임을 선택한 후 좌측의 개발 도구 > 서비스 및 API 메뉴를 선택한다.

실시간 개발자 알림

앱의 인앱 상품과 관련된 이벤트 알림을 받으려면 실시간 개발자 알림을 설정하세요.

Note: Ensure that the Cloud Pub/Sub topic exists and you have provided Google with the necessary privileges to publish push notifications on your topic. Refer to the [Configure Real-time Developer Notifications Guide](#) to learn more.

주제 이름

43/300

Enter the Cloud Pub/Sub topic name in the projects/{project_id}/topics/{topic_name} format.

테스트 알림 보내기 ②

- 하단의 "실시간 개발자 알림" 항목에 앞서 생성한 주제 이름을 입력하여 저장한다.

notification data

Real-time Notification 을 통해 전달받은 데이터는 아래와 같다.

```
{
  "message": {
    "attributes": {
      "key": "value"
    }
  },
}
```



```

    "data":
      "eyJhdmVyc2lvbiI6IHN0cm1uZywgInBhY2thZ2ZVOYW11Ijogc3RyaW5nLCAiZXZlbnRUaW11TWlzbG1zI  

      jogbG9uZywgIm9uZVRpbWVQcm9kdWN0Tm90aWZpY2F0aW9uIjogT251VGltZVByb2R1Y3R0b3RpZmljYXR  

      pb24sICJzdWJzY3JpcHRpb250b3RpZmljYXRpb24iOiBTdWJzY3JpcHRpb250b3RpZmljYXRpb24sICJ0Z  

      XN0Tm90aWZpY2F0aW9uIjogVGZvdE5vdGlmawNhGlvbiB9",
      "messageId": "136969346945"
    },
    "subscription": "projects/myproject/subscriptions/mysubscription"
  }

```

- subscription: 구독 아이디
- message.attributes: 메시지의 속성들의 집합
- message.data: Base64 로 인코딩된 메시지 내용
- message.messageId: 메시지 아이디

Base64 로 인코딩된 message.data 를 디코딩하면 아래와 같이 구매한 상품 정보를 확인할 수 있다.

One Time Product Notification

1회성 상품을 구매하였을 경우의 유형

```

{
  "version": "1.0",
  "packageName": "com.test.sample",
  "eventTimeMillis": 1595403693695,
  "oneTimeProductNotification": {
    "version": "1.0",
    "notificationType": 1,
    "purchaseToken": "...",
    "sku": "test_1"
  }
}

```

- version: 메시지의 버전
- packageName: 패키지 명
- eventTimeMillis: 이벤트가 발생한 시간
- oneTimeProductNotification.version: notification 의 버전
- oneTimeProductNotification.notificationType: notification 의 유형
 - 1: 지연 결제 상품을 성공적으로 구매함
 - 2: 지연 결제 상품을 취소함
- oneTimeProductNotification.purchaseToken: 구매 토큰(영수증)
- oneTimeProductNotification.sku: 상품 아이디

실제 전달된 데이터

데이터가 전달되지 않음

Subscription Notification

구독형 상품을 구매하였을 경우의 유형

```
{
  "version": "1.0",
  "packageName": "com.test.sample",
  "eventTimeMillis": 1595403693695,
  "subscriptionNotification": {
    "version": "1.0",
    "notificationType": 1,
    "purchaseToken": "...",
    "subscriptionId": "test_2"
  }
}
```

- version: 메시지의 버전
- packageName: 패키지 명
- eventTimeMillis: 이벤트가 발생한 시간
- subscriptionNotification.version: notification 의 버전
- subscriptionNotification.notificationType: notification 의 유형
 - 1: 결제 보류 상태에서 구독이 복구됨
 - 2: 구독이 갱신됨
 - 3: 구독이 취소됨
 - 4: 새로운 구독이 시작됨
 - 5: 결제 보류 상태
 - 6: 구독 유예 기간이 시작됨
 - 7: 사용자가 play store > 계정 > 구독 메뉴에서 구독을 활성화함
 - 8: 사용자가 구독 상품의 가격 변동을 확인함
 - 9: 구독 갱신 기한이 연장됨
 - 10: 구독이 일시 중지됨
 - 11: 구독 일시 중지 기간이 변경됨
 - 12: 구독 만료 전에 사용자에게 의해 취소됨
 - 13: 구독이 만료됨
- subscriptionNotification.purchaseToken: 구매 토큰(영수증)
- subscriptionNotification.subscriptionId: 상품 아이디

실제 전달된 데이터

```
Google Real-time Notification ==> data:
eyJ2ZXJzaW9uIjoiaMS4wIiwicGFja2FnZU5mVyc2lvdBI6IjEuMCIsIm5vdGlmawNhdGlvb1R5cGUiOjQs
InB1cmNoYXNlVG9rZW4iOiJmbm9nYnBpZWRRZWJhamZlYW9sbWpubWQuQU8tSjFPd1ZfOTFqcZJBVDVZWm
YyblA3dj1lNFpwYUxZYXNlbnpVSEluMnVWUVJlVW9uamJjSDBfazI1RXJMcHQxTjdaa2Q4bU1GRzJUNjZk
NnJZSFB4Wj12X1dEdnBFbzhnNldoY2RISC10d3lme19tNVRuMHM4YV9Ec3RIeWNjLTU3l0ZWpwN3kiLC
JzdWJzY3JpcHRpb25JZCI6ImpveXBsZV9zdWJfdGVzdF8yIn19, messageId: 1367885263800285
Google Real-time Notification ==> convert data:
{"version":"1.0","packageName":"com.test.sample","eventTimeMillis":"1595481526062",
,"subscriptionNotification":
```

```
{"version":"1.0","notificationType":4,"purchaseToken":PURCHASE_TOKEN,"subscriptionId":"test_2"}}
```

Test Notification

테스트 알림 발송 시 전달되는 유형

```
{
  "version": "1.0",
  "packageName": "com.test.sample",
  "eventTimeMillis": 1595403693695,
  "testNotification": {
    "version": "1.0"
  }
}
```

- version: 메시지의 버전
- packageName: 패키지 명
- eventTimeMillis: 이벤트가 발생한 시간
- testNotification.version: notification 의 버전

실제 전달된 데이터

```
Google Real-time Notification ==> data:
eyJ2ZXJzaW9uIjoIbW9uIiwicGFja2FnZU5mVyc2lvdG90IjEjEuMCIsIm5vdGlmawNhdGlvb1R5cGUiOjQs
InB1cmNoYXNlVG9rZW4iOiJmbm9nYnBpZWRRZWJhamZlYW9sbWpubWQuQU8tSjFPd1ZfOTFqcZJBVDVZWm
YyblA3dj1lNFpwYUxZYXNlbnpVSEluMnVWUVJlVW9uamJjSDBfazI1RXJMcHQxTjdaa2Q4bU1GRzJUNjZk
NnJZSFB4Wjl2X1dEdnBFbzhnNldoY2RISC10d3lme19tNVRuMHM4YV9Ec3RIewNjLTU3l0ZWpwN3kiLC
JzdWJzY3JpcHRpb25JZCI6ImpveXBsZV9zdWJfdGVzdF8yIn19, messageId: 1367885263800285
Google Real-time Notification ==> convert data:
{"version":"1.0","packageName":"com.test.sample","eventTimeMillis":"1595481526062"
,"subscriptionNotification":{"version":"1.0"}}
```

고려사항

구독 상품의 경우 결제 혹은 구독 갱신이 이루어지는 시점과 Pub/Sub 을 통해 notification 을 전달받는 시간이 비슷하며, 거의 동시에 동작한다고 볼 수 있다.

단, 일반 상품 구매 시 전달되는 One Time Product Notification 은 모든 일반 상품(consumable) 이 아닌 지연 결제 상품만 전달이 되는 것으로 보인다.(일반 상품 구매 시 전달이 되지 않는 것을 확인)

아래는 지연 결제 상품만 지원한다는 내용의 공식 가이드 문서이다.

https://developer.android.com/google/play/billing/billing_library_releases_notes?hl=ko#2_0_pending

대기 중인 거래

Google Play 결제 라이브러리 버전 2.0을 사용하면 자격을 부여하기 전에 추가 작업이 필요한 구매를 지원해야 합니다. 예를 들어 사용자가 오프라인 상점에서 현금으로 인앱 상품을 구매할 수도 있습니다. 이는 앱 외부에서 완료된 거래입니다. 이 시나리오에서는 사용자가 거래를 완료한 후에만 자격을 부여해야 합니다.

대기 중인 구매를 사용 설정하려면 앱 초기화의 일부인 `enablePendingPurchases()` 를 호출하세요.

`Purchase.getPurchaseState()` 를 사용하면 구매 상태가 `PURCHASED` 인지 또는 `PENDING` 인지 확인할 수 있습니다. 상태가 `PURCHASED` 인 경우에만 자격을 부여해야 합니다. 다음과 같이 `Purchase` 상태 업데이트를 확인해야 합니다.

1. 앱을 시작할 때 `BillingClient.queryPurchases()` 를 호출하여 사용자와 연관된 소비되지 않은 제품의 목록을 가져옵니다.
2. 반환된 각 `Purchase` 객체에서 `Purchase.getPurchaseState()` 를 호출합니다.
3. `onPurchasesUpdated()` 메서드를 구현하여 `Purchase` 객체의 변경에 응답합니다.

또한 Google Play Developer API에는 `PENDING` `Purchases.products` 의 상태가 포함됩니다. 정기 결제에서는 대기 중인 거래가 지원되지 않습니다.

또한 이번 출시에 새로운 실시간 개발자 알림 유형인 `OneTimeProductNotification` 이 도입됩니다. 이 알림 유형은 값이 `ONE_TIME_PRODUCT_PURCHASED` 또는 `ONE_TIME_PRODUCT_CANCELED` 인 메시지 1개를 포함합니다. 이 알림 유형은 현금과 같이 지연된 결제 방법과 관련된 구매인 경우에만 전송됩니다.

대기 중인 구매는 해당 구매가 `PENDING` 상태가 아닌 `SUCCESS` 상태인 경우에만 확인해야 합니다.

★ **참고:** 대기 중인 거래는 라이선스 테스터를 사용해 테스트할 수 있습니다. 라이선스 테스터는 2개의 테스트 신용카드 외에도 지연된 결제 방법을 위한 새로운 테스터 도구 2개를 사용할 수 있습니다. 이 도구는 몇 분 후에 자동으로 거래를 완료하거나 취소합니다. 애플리케이션을 테스트하는 동안, 이 새로운 도구 2개 중 하나를 사용하여 구매한 직후에 애플리케이션이 자격을 부여하거나 구매를 확인하지 않는지 확인해야 합니다. 자동으로 완료되는 새 테스트 도구를 사용해 구매할 때는, 구매가 완료되면 애플리케이션에서 자격을 부여하고 구매를 확인하는지 확인해야 합니다.

적용 방향

Real-time Notification 을 이용한 구매 프로세스 적용 방법은 아래와 같다.

- 구독 갱신 중 결제 오류로 인하여 본래 갱신 일자 보다 나중에 갱신되는 경우
 - `subscriptionNotification.notificationType` 이 1(결제 보류 상태에서 구독이 복구됨)인 경우를 확인하여 구독 갱신 프로세스를 진행한다.
- 지연 결제 상품의 경우
 - `oneTimeProductNotification.notificationType` 이 1(지연 결제 상품을 성공적으로 구매함)인 경우를 확인하여 구매 프로세스를 진행한다.