



Menu

On this page

## 사용자 로그 남기기

사용자의 행동과 요소 노출을 로깅해서 분석할 수 있어요. 페이지 간 이동 로그는 자동으로 기록되니까 따로 설정할 필요가 없어요.

## 클릭 이벤트 로깅

사용자가 버튼을 클릭할 때 같은 상호작용을 기록하는 방법이에요.

javascript

```
import { Analytics } from "@apps-in-toss/web-framework";

// 'myButton' 버튼을 클릭하면 'my_button' 이벤트를 로깅해요.
document.getElementById("myButton").addEventListener("click", function () {
  Analytics.click({ button_name: "my_button" });
  // 클릭 후 실행할 추가 동작을 여기에 작성해요.
});
```

### 설명

- `Analytics.click` 메서드는 클릭 이벤트를 로깅해요.
- `button_name` 파라미터는 어떤 버튼인지 식별할 이름이에요. (예: 'my\_button')
- 클릭한 뒤 추가로 필요한 동작이 있다면 주석 아래에 코드를 작성하면 돼요.

## 요소 노출 이벤트 로깅

사용자 화면에 특정 요소가 보일 때 이벤트를 기록하는 방법이에요. `IntersectionObserver` API를 사용해서 구현해요.

```
import { Analytics } from "@apps-in-toss/web-framework";

const target = document.getElementById("impressionItem");

const observer = new IntersectionObserver(
  ([entry]) => {
    if (entry.isIntersecting) {
      Analytics.impression({ item_id: target.dataset.itemId });
      observer.disconnect();
    }
  },
  { threshold: 0.1 }
);

observer.observe(target);
```

## 설명

- `IntersectionObserver` 는 `target` 요소가 화면에 10% 이상 노출될 때 콜백을 실행해요.
- `Analytics.impression` 메서드는 노출 이벤트를 로깅해요.
- `item_id` 파라미터는 어떤 아이템이 노출됐는지 식별할 ID예요. (예: '1234')
- 노출을 한 번만 기록하기 위해 `observer.disconnect()` 로 감시를 중단해요.

## HTML 예시

```
<div id="impressionItem" data-item-id="1234">노출을 감지할 요소</div>
```

---

## 유의 사항

- 로깅 메서드는 사용자가 의미 있는 상호작용을 할 때만 호출하는 게 좋아요. 불필요한 호출은 분석 품질을 떨어뜨릴 수 있어요.
- 추가 파라미터는 구체적이고 명확하게 설정해요. 예를 들어 버튼 이름이나 아이템 ID를 잘 구분할 수 있도록 지정하면 이벤트 분석이 훨씬 쉬워져요.

Previous page

[인앱 결제 연동하기](#)

Next page

[이미지 불러오기에 실패했을 때 처리하기](#)