# Google Analytics(GA)

웹사이트의 트래픽과 사용자 행동을 분석하는 도구로, 웹사이트 소유자가 사이트의

성과를 추적하고 이해할 수 있도록 도와줌.

## 기능 및 분석 항목(Real-Time Reports)

**현재 방문자 수**: 사이트에 현재 접속 중인 사용자 수를 실시간으로 확인할 수 있음.

**페이지뷰**: 실시간으로 조회되는 페이지와 그에 대한 사용자 수를 추적.

**트래픽 소스**: 실시간으로 사용자가 어떤 소스(직접 방문, 검색 엔진, 소셜 미디어 등)에서

왔는지 확인할 수 있음.

**사용자 위치 및 기기**: 사용자의 지리적 위치와 사용 중인 디바이스 종류를 실시간으로 파악.

## 청중 분석(Audience Analysis)

**인구 통계**: 방문자의 나이, 성별 등의 정보를 제공.

**기술적 속성**: 사용자가 사용하는 브라우저, 운영 체제, 기기 유형 등을 분석.

**사용자 행동**: 사용자 재방문율, 세션 기간 등을 통해 방문자 유지 및 재방문 패턴을 파악

## 획득(Acquisition)

**채널**: 트래픽이 들어오는 채널을 분류하여 분석함. (예: 오가닉 검색, 직접, 유료 광고, 소셜 미디어 등)

**캠페인 추적**: UTM 파라미터를 사용해 특정 캠페인의 효과를 측정할 수 있음..

## 행동(Behavior)

**사이트 콘텐츠**: 가장 많이 조회된 페이지와 사용자 흐름을 분석함.

**사이트 속도**: 페이지 로딩 시간과 그에 따른 사용자 이탈률을 분석함.

**검색 분석**: 사이트 내 검색 데이터를 분석해 사용자들이 원하는 정보를 파악함.

## 전환(Conversion)

**목표 설정**: 특정 목표(예: 구매 완료, 회원 가입 등)를 설정하고, 그 목표 달성률을 추적 할 수 있음.

**전자상거래 분석**: 제품별 매출, 평균 주문 금액, 구매 전환율 등을 분석할 수 있음.

## 고급 기능 및 사용자 정의

**세그먼트**: 특정 사용자 그룹을 정의하고 그들의 행동을 분석할 수 있음.

**커스텀 보고서**: 필요에 따라 맞춤형 보고서를 생성하여 특정 지표를 깊이 분석할 수 있음.

**API 연동**: GA 데이터를 외부 애플리케이션과 연동하여 더욱 심층적인 분석과 활용이 가능.

# CORS(Cross-Origin Resource Sharing)

## 기본 원리

Same-Origin Policy: 웹 브라우저의 보안 모델로, 동일한 출처에서만 리소스를 요청할 수 있도록 제한함.

출처는 스키마(프로토콜), 호스트(도메인), 포트의 조합으로 정의됨.

이 정책은 XSS(Cross-Site Scripting)과 같은 보안 위협을 방지하는 데 중요한 역할을 함.

## CORS의 역할

CORS 헤더: 서버는 HTTP 응답 헤더를 사용해 어떤 출처에서 리소스를 요청할 수 있는지를 명시.

주요 헤더는 다음과 같음.

Acess-Control-Allow-Origin: 허용된 출처를 명시. *은 모든 출처를 허용함을 의미.

Acess-Control-Allow-Methods: 허용된 HTTP 메서드 (예: GET, POST, PUT 등).

Acess-Control-Allow-Headers: 허용된 커스텀 헤더

Acess-Control-Allow-Credentials: 자격 증명(쿠키 등)을 포함할 수 있는지 여부를 결정.

## 프리플라이트 요청(Preflight Request)

복잡한 요청(예: 메서드가 GET, POST 외, 커스텀 헤더 사용 등) 전에 브라우저는 서버에 OPTIONS 메서드로 사전 요청을 보내어 해당 요청이 허용되는지 확인함.

서버가 허용할 경우 응답에 적절한 CORS 헤더를 포함하여 브라우저에 이를 알림.

## CORS 설정 예시

예를 들어, 다음과 같은 헤더를 서버 응답에 추가하여 특정 출처에서만 리소스를 접근할 수 있도록 할 수 있음.

```
Access-Control-Allow-Origin: https://example.com
Access-Control-Allow-Methods: GET, POST
Access-Control-Allow-Headers: Content-Type
```

이는 'https://example.com' 도메인에서 오는 GET 및 POST 요청을 허용하며, 요청 시 Content-Type 헤더를 사용할 수 있음을 나타냄.

## 보안 고려 사항

신중한 허용 설정: 특정 출처만을 허용하여 허용되지 않은 출처에서의 불법 접근을 방지함.

자격 증명 관리: Access-Control-Allow-Credentials 헤더를 사용할 때 주의가 필요하며, 반드시 신뢰할 수 있는 출처에서만 허용해야 함.

CORS는 웹 보안의 중요한 요소로, 올바르게 설정하지 않으면 CSRF(Cross-Site Request Forgery)와 같은 공격에 노출될 수 있음. 따라서, 서버 측의 리소스 접근을 관리할 때 신중하게 구성해야 함.