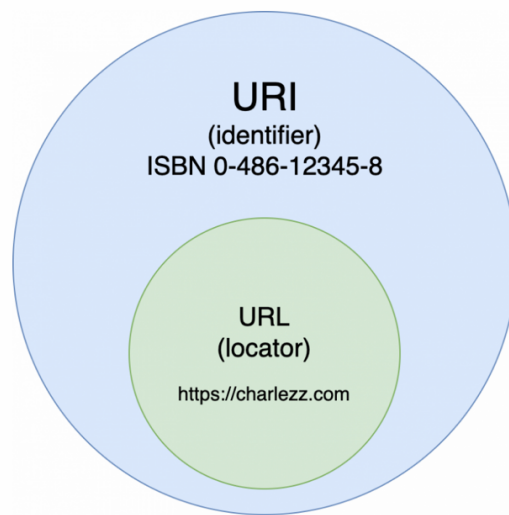


URI (Uniform Resource Identifier)

URI는 특정 리소스를 식별하는 **통합 자원 식별자(Uniform Resource Identifier)**를 의미한다. 웹 기술에서 사용하는 논리적 또는 물리적 리소스를 식별하는 고유한 문자열 시퀀스다.

URL (Uniform Resource Locator)

URL은 흔히 웹 **주소**라고도 하며, 컴퓨터 네트워크 상에서 리소스가 **어디 있는지 알려주기** 위한 규약이다. URI의 서브셋이다.



비록 URL은 URI의 서브셋이지만 URI와 URL의 가장 큰 차이점은 바로 아래와 같다

~~URI~~ URI는 식별하고, URL은 위치를 가르킨다.

우선

URL은 Uniform Resource Locator

URI는 Uniform Resource Identifier

이다.

그냥 단어의 뜻대로 해석하자면 예전에는 URL이 가리키는게 **자원(파일) 위치** 였는데

요즘은 Rewrite 등의 Apache , IIS, Tomcat 핸들러 때문에 **자원 식별자** 이라고 부른다.

즉 요즘에는 웹사이트 주소가 (<http://test.com/company/location>) 라고 했을 때

(요청하는 주소가 파일의 위치를 말하는 것(Location)이기 보다는)구분자(Identifier)로 보는 것이다.

실제로 해당 웹사이트의 [company/location](#) 라는 파일은 없다. (아마도 **company** 클래스의 **location** 메소드를 호출할 것이다. 이렇게 **구분자(Identifier)**로 보는 것이 URI 이다.)

요약하자면 URL 은 다음과 같다.

<http://test.com/work/sample.pdf>

[test.com](#) 서버에서 [work](#) 폴더안의 [sample.pdf](#) 를 요청하는 URL.

URI(통합 자원 식별자) 의 예는 다음과 같다.

1) **rewrite** 기술을 사용하여 만든 의미있는 식별자

`http://test.com/company/location`

2) **REST 서비스 (url로 실행되는 서비스)**

`http://service.com/tv/turn/on`

3) **Web-oriented architecture** (웹 기반의 구조문법)

`kakaotalk://sendmsg?text=hello!` (이 uri는 `kakaotalk` 프로토콜을 해석할 수 있는 프로그램이 핸들링한다. 해당프로그램은 `sendmsg` 라는 식별자를 해석하고 동작한다.)

`facebookmsg://like?url=mysite.com` (이 uri는 `facebookmsg` 프로토콜을 해석할 수 있는 프로그램이 핸들링한다. 해당프로그램은 `like` 라는 식별자를 해석하고 동작한다.)

