제 6 장 URL 클래스를 이용하여 데이터 읽기

- URL이란?
- URL 클래스

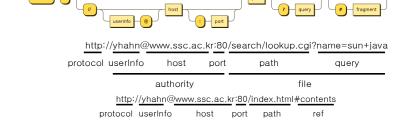
.

제 6 장

URL 클래스를 이용하여 데이터 읽기

• URL이란

• URL(Uniform Resource Locator)은 WWW 상에서 리소스를 가리키며, 통신 프로토콜, 호스트 이름, 포트 번호, 경로 및 파일이름, 그리고 질의 문자열 및 문서의 섹션으로 구성됨



http://www.microsoft.com ; HTTP 프로토콜을 사용하는 URL file:///c:\ayh\document\suwon.html ; file URL를 나타낸다.

file

authority

http://www.corevalley.com:80/doc.html ; 80번 포트를 사용하는 URL ftp://ftp.suwon-sc.ac.kr/pub/ ; ftp 프로토콜을 사용하는 URL

mailto://yhahn@mail.suwon-sc.ac.kr ; 메일 URL

•2

1

• URL 클래스

•URL를 구성하는 프로토콜, 호스트 이름, 포트 번호, 경로 및 파일이름, 그리고 문서의 섹션 번호를 저장하는 필드 및 필드의 값을 구하는 메소드들로 구성됨

protected String protocol; URL 클래스의 필드들 protected String host; protected int port; protected String file; protected String ref; public String getProtocol() 필드의 값을 구하는 메소드들 public String getHost() public int getPort() pulic String getFile() public String getRef() public String getPath() public String getQuery() public String getUserInfo() public String getAuthority()

•3

제 6 장

URL 클래스를 이용하여 데이터 읽기

• URL 객체 만들기

• 4개의 생성자 메소드를 지원

public URL(String url) throws MalformedURLException

예제6.1 URLConstructor1.java
실행결과
다양한 URL 객체를 생성하고 화면에 출력
핵심내용
URL 형식이 잘못되면 MalformedURLException 예외를 발생

제 6 장

public URL(String protocol, String host, String file) throws MalformedURLException

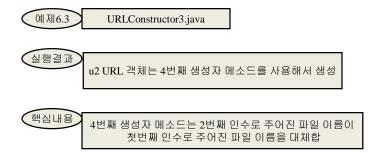


•5

제 6 장

URL 클래스를 이용하여 데이터 읽기

public URL(String protocol, String host, int port, String file) throws MalformedURLException public URL(URL u, String s) throws MalformedURLException



• 다양한 프로토콜의 지원 여부를 테스트



•7

제 6 장

URL 클래스를 이용하여 데이터 읽기

- 필드값 구하기
- 다양한 get 메소드를 사용해서 필드 값을 구하는데 사용



제 6 장

• URL로부터 데이터 읽어오기

• 3개의 메소드를 사용해서 url 객체가 가리키는 파일의 내용을 읽을 수 있음

public final InputStream openStream() throws IOException
public URLConnection openConnection() throws IOException
public final Object getContent() throws IOException, UnknownServiceException

•9

제 6 장

URL 클래스를 이용하여 데이터 읽기

public final InputStream openStream() throws IOException 메소드를 사용해서 파일의 내용을 읽어오는 과정

첫째, 상대방 호스트의 URL 객체를 생성

- 예) URL url = new URL(http://chosun.com);
- 둘째, openStream() 메소드를 이용하여 URL이 지정하는 호스트 컴퓨터와 연결하고, 데이터를 읽기 위한 InputStream 객체를 생성. InputStream 객체의 read() 메소드를 사용하여 파일의 내용을 읽음
 - 예) InputStream is = url.openStream(); // url과 연결시키는 is 스트림 객체 생성
- 셋째, 위에서 생성된 InputStream 객체를 이용하여 원하는 필터 또는 버퍼 입력 스트림을 생성하고, 상대방 호스트로부터 URL이 가리키는 파일의 원시 데이터를 읽음
 - 에) BufferedReader br = new BufferedReader(new InputStreamReader(is)); String data = br.readLine();

• URL 객체가 가리키는 파일의 내용을 읽음



•11

제 6 장

URL 클래스를 이용하여 데이터 읽기

• getContent() 메소드를 사용해서 파일의 내용을 읽음



