

- URL이란?
- URL 클래스

제 6 장

- URL이란

```

graph LR
    URI[URI] --> S[scheme]
    S --> C1[:]
    C1 --> H[host]
    H --> U[userinfo]
    H --> AT[@]
    H --> P[port]
    H --> S2[/]
    S2 --> P1[path]
    P1 --> Q[query]
    P1 --> F[fragment]
    P1 --> QM[?]
    P1 --> HM[#]
    
```

<u>http://yhahn@www.ssc.ac.kr:80/index.html#contents</u>					
protocol	userInfo	host	port	path	ref

•2

• URL 클래스

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

```
protected String protocol;
protected String host;
protected int port;
protected String file;
protected String ref;
public String getProtocol()
public String getHost()
public int getPort()
public String getFile()
public String getRef()
public String getPath()
public String getQuery()
public String getUserInfo()
public String getAuthority()
```

URL 클래스의 필드들

필드의 값을 구하는 메소드들

•3

• URL 객체 만들기

• 4개의 생성자 메소드를 지원

```
public URL(String url) throws MalformedURLException
```

예제 6.1

URLConstructor1.java

실행결과

다양한 URL 객체를 생성하고 화면에 출력

핵심내용

URL 형식이 잘못되면 MalformedURLException 예외를 발생

•4

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

예제 6.2

URLConstructor2.java

실행결과

다양한 URL 객체를 생성하고 화면에 출력

핵심내용

프로토콜, 호스트 이름 및 파일 이름을 별도로 지정.
URL 형식이 잘못되면 MalformedURLException 예외를 발생.

*5

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

예제 6.3

URLConstructor3.java

실행결과

u2 URL 객체는 4번째 생성자 메소드를 사용해서 생성

핵심내용

4번째 생성자 메소드는 2번째 인수로 주어진 파일 이름이
첫번째 인수로 주어진 파일 이름을 대체함

*6

- 다양한 프로토콜의 지원 여부를 테스트

예제 6.4

ProtocolTester.java

실행결과

주어진 프로토콜의 지원여부를 테스트

핵심내용

지원되지 않는 프로토콜은 `MalformedURLException` 예외를 발생

*7

• 필드값 구하기

- 다양한 get 메소드를 사용해서 필드 값을 구하는데 사용

예제 6.5

GetURLFields.java

실행결과

get 메소드를 사용해서 URL 객체의 필드 값을 읽어서 출력

핵심내용

URL 형식이 잘못되면 `MalformedURLException` 예외를 발생

*8

- 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

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

예) BufferedReader br = new BufferedReader(new InputStreamReader(is));

String data = br.readLine();

•10

- URL 객체가 가리키는 파일의 내용을 읽음

예제 6.6 ReadSourceFile.java

실행결과 URL 객체가 가리키는 파일의 내용을 읽어서 화면에 출력

핵심내용 버퍼드 스트림 `br`은 URL 객체가 가리키는 파일과 연결되어 있음

•11

- `getContent()` 메소드를 사용해서 파일의 내용을 읽음

예제 6.7 GetObject.java

실행결과 `url`이 가리키는 파일의 내용을 읽어서 출력

핵심내용 `getContent()` 메소드는 객체를 반환

•12

예제 6.8

ReadServerFile.java

실행결과

텍스트 필드에서 입력한 URL이 가리키는 파일의 내용을 읽어서
텍스트 에리어에 출력

핵심내용

GUI 기반에서 실행

