HTTP Request

2016-12-08

이승진

**학습목표**

웹서버에 요청(HTTP request)을 보내서 웹페이지 내용을 받아 온다.

받아 온 웹페이지 내용에서 제목과 분문을 추출한다.

받아 온 게시글을 파일에 저장한다.

**목차**

[1. 배경지식 2](#_Toc468967626)

[1) HTTP Request and Response 2](#_Toc468967627)

[2) HttpURLConnection 클래스 2](#_Toc468967628)

[3) 인코딩이란? 2](#_Toc468967629)

[4) Java 문자열의 특수 문자 3](#_Toc468967630)

[2. 게시글 제목 추출 예제 코드 4](#_Toc468967631)

[1) Example01 4](#_Toc468967632)

[2) Example02 5](#_Toc468967633)

[3) Example03 6](#_Toc468967634)

[3. 게시글 목록 받아 오기 7](#_Toc468967635)

[1) Article.java 7](#_Toc468967636)

[2) Example04 8](#_Toc468967637)

[4. 게시글 내용 받아 오기 9](#_Toc468967638)

[1) 게시글 내용 웹페이지 9](#_Toc468967639)

[2) 게시글 내용 태그 분석 10](#_Toc468967640)

[3) Example05 11](#_Toc468967641)

[4) Example06 12](#_Toc468967642)

[5) Example07 13](#_Toc468967643)

[6) Example08 14](#_Toc468967644)

[7) Example09 16](#_Toc468967645)

# 배경지식

## HTTP Request and Response

웹브라우저의 주소창에 URL을 입력하고 엔터키를 누르면, 웹브라우저는 웹서버에 HTTP Request를 전송한다.

HTTP Request에는 요청하는 것을 가르키는 URL과 파라미터가 들어있다.

HTTP Request를 받은 웹서버는 요청받은 내용을 실행한 다음에, 실행 결과를 웹브라우저에 전송한다.

웹브라우저의 HTTP Request의 결과로 웹서버가 전송하는 내용을 HTTP Response라고 부른다.

HTTP Response에는 웹브라우저에 출력될 내용(contents)이 담겨있다. (HTML 태그, 이미지 등)



## HttpURLConnection 클래스

웹브라우저 뿐만 아니라 Java 코드도 웹서버에 HTTP Request를 전송하여 HTTP Response를 받아올 수 있다.

HttpURLConnection 클래스를 이용하여 웹서버에 HTTP Request를 전송한다.

## 인코딩이란?

컴퓨터 내부에서 데이터는 이진수로 저장된다.

숫자 뿐만이 아니라 문자도 역시 이진수로 저장된다.

그래서 'a','b','c','가','나','다' 문자 각각 마다 이진수 숫자가 하나씩 정해져 있다.

컴퓨터 내부에 문자를 저장할 때 그 문자에 부여된 이진수 숫자로 저장된다.

문자에 부여된 이진수 숫자를 그 문자의 코드(Character Code)라고 부른다.

문자 각각에 이진수 숫자를 부여하는 방식을 문자 인코딩(Character Encoding)이라 부른다.

그냥 줄여서 인코딩이라고도 한다.

그런데 이 문자 인코딩 방식이 하나가 아니고 여럿이다.

여러 문자 인코딩 방식들에서 알파벳은 대부분 동일한 이진수 숫자가 부여되지만,

한글 문자에 부여되는 이진수 숫자는 인코딩 방식 마다 서로 다르다.

예를 들어 문자 '가'에 부여된 문자 코드 숫자는 다음과 같다.

|  |  |
| --- | --- |
| 인코딩 방식 | 문자 코드 |
| EUC-KR | 0xB0 0xA1 |
| UNICODE | 0xAC 0x00 |
| UTF-8 | 0xEA 0xB0 0x80 |

문자 'A'에 부여된 문자 코드 숫자는 다음과 같다.

|  |  |
| --- | --- |
| 인코딩 방식 | 문자 코드 |
| EUC-KR | 0x41 |
| UNICODE | 0x41 0x00 |
| UTF-8 | 0x41 |

문자를 컴퓨터 내부에 이진수 숫자로 저장할 때의 인코딩과

저장된 이진수 숫자를 읽어서 문자로 해석하여 회면에 출력할 때의 인코딩이 다르면

원래 저장했던 문자가 아니라 이상한 문자로 해석되어 출력될 것이다.

한글 문자를 다룰 때 이 인코딩 불일치 문제를 조심해야 한다.

웹에서 주로 사용하는 문자 인코딩 두개는 다음과 같다.

(1) UTF-8 인코딩

(2) EUC-KR 인코딩 (MS949, ANSI 인코딩이라고도 부른다)

웹서버에서 내용을 받아올 때 그 내용의 인코딩이 무엇인지 확인해야 한다.

## Java 문자열의 특수 문자

### " 문자

Java 문자열에 포함된 " 문자는 \" 형태로 입력해야 한다.

예)

System.out.println("value=3");

위 코드의 출력은 다음과 같다.

value=3

System.out.println("value='3'");

위 코드의 출력은 다음과 같다.

value='3'

System.out.println("value=\"3\"");

위 코드의 출력은 다음과 같다.

value="3"

### \ 문자

Java 문자열에 포함된 \ 문자는 \\ 형태로 입력해야 한다.

예)

System.out.println("c:\\temp\\data.txt");

위 코드의 출력은 다음과 같다.

c:\temp\data.txt

\n 문자

Java 문자열에 포함된 줄바꿈 문자는 \n 형태로 입력해야 한다.

예)

System.out.println(" hello\n world");

위 코드의 출력은 다음과 같다.

hello

world

# 게시글 제목 추출 예제 코드

## Example01

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  public class Example01 {  public static void main(String[] args) throws Exception {  String listUrl = "http://www.skhu.ac.kr/board/boardlist.aspx?curpage=1&bsid=10004&searchBun=51";  String html = getHttpResponse(listUrl);    System.out.println(html);  }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |

Example01 예제 코드는

성공회대학교 홈페이지 웹서버에 URL을 요청하고, 그 결과를 받아서,

받아 온 웹페이지 내용 전체를 화면에 출력한다.

아래의 URL은 성공회대학교 홈페이지 게시판의 URL 이다.

"http://www.skhu.ac.kr/board/boardlist.aspx?curpage=1&bsid=10004&searchBun=51"

String getHttpReqponse(String url) 메소드는

웹서버에 URL을 요청하고, 그 결과를 받아서 문자열로 리턴하는 메소드이다.

## Example02

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Example02 {  public static void main(String[] args) throws Exception {  String listUrl = "http://www.skhu.ac.kr/board/boardlist.aspx?curpage=1&bsid=10004&searchBun=51";  String html = getHttpResponse(listUrl);    String regex = "<a href='boardread.aspx[^']+'>([^<]+)</a>";  Pattern pattern = Pattern.compile(regex);  Matcher matcher = pattern.matcher(html);  while (matcher.find())  System.out.println(matcher.group(1));  }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |

Example02 예제 코드는

성공회대학교 홈페이지 웹서버에 URL을 요청하고, 그 결과를 받아서,

받아 온 웹페이지 내용 중에서 게시글 제목을 찾아서 화면에 출력한다.

아래의 정규식은 게시글 제목을 찾기위한 정규식이다.

"<a href='boardread.aspx[^']+'>([^<]+)</a>"

## Example03

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Example03 {  public static void main(String[] args) throws Exception {  String listUrl = "http://www.skhu.ac.kr/board/boardlist.aspx?curpage=1&bsid=10004&searchBun=51";  String html = getHttpResponse(listUrl);    String regex = "<a href='boardread.aspx\\?idx=([0-9]+)[^>]+>([^<]+)</a>";  Pattern pattern = Pattern.compile(regex);  Matcher matcher = pattern.matcher(html);  while (matcher.find())  System.out.printf("%s %s\n", matcher.group(1), matcher.group(2));  }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |

Example03 예제 코드는

성공회대학교 홈페이지 웹서버에 URL을 요청하고, 그 결과를 받아서,

받아 온 웹페이지 내용 중에서 게시글 아이디(idx)와 제목을 찾아서 화면에 출력한다.

아래의 정규식은 게시글 아이디(idx)와 제목을 찾기위한 정규식이다.

"<a href='boardread.aspx\\?idx=([0-9]+)[^>]+>([^<]+)</a>"



# 게시글 목록 받아 오기

## Article.java

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41 | package net.skhu.http;  public class Article {  String idx;  String title;  String body;  String date;  public String getIdx() {  return idx;  }  public void setIdx(String idx) {  this.idx = idx;  }  public String getTitle() {  return title;  }  public void setTitle(String title) {  this.title = title;  }  public String getBody() {  return body;  }  public void setBody(String body) {  this.body = body;  }  public String getDate() {  return date;  }  public void setDate(String date) {  this.date = date;  }    } |

웹서버에서 받아온 게시글 내용을 채우기 위한 클래스이다.

|  |  |
| --- | --- |
| idx | 게시글 아이디 |
| title | 게시글 제목 |
| body | 게시글 내용 |
| date | 게시글 작성일 |

## Example04

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  import java.util.ArrayList;  import java.util.List;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Example04 {    public static void main(String[] args) throws Exception {  String listUrl = "http://www.skhu.ac.kr/board/boardlist.aspx?curpage=1&bsid=10004&searchBun=51";  List<Article> list = getArticleList(listUrl);  for (Article article : list)  System.out.printf("%s %s\n", article.getIdx(), article.getTitle());  }    static List<Article> getArticleList(String url) throws IOException {  List<Article> list = new ArrayList<Article>();  String html = getHttpResponse(url);  String regex = "<a href='boardread.aspx\\?idx=([0-9]+)[^>]+>([^<]+)</a>";  Pattern pattern = Pattern.compile(regex);  Matcher matcher = pattern.matcher(html);  while (matcher.find()) {  Article article = new Article();  article.setIdx(matcher.group(1));  article.setTitle(matcher.group(2));  list.add(article);  }  return list;  }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |

getArticleList 메소드는,

성공회대학교 홈페이지 웹서버에 URL을 요청하고, 그 결과를 받아서,

받아 온 웹페이지의 내용 중에서 게시글 아이디(idx)와 제목을 찾아서,

찾은 내용을 Article 객체에 채워서,

Article 객체 목록을 리턴하는 메소드이다.

# 게시글 내용 받아 오기

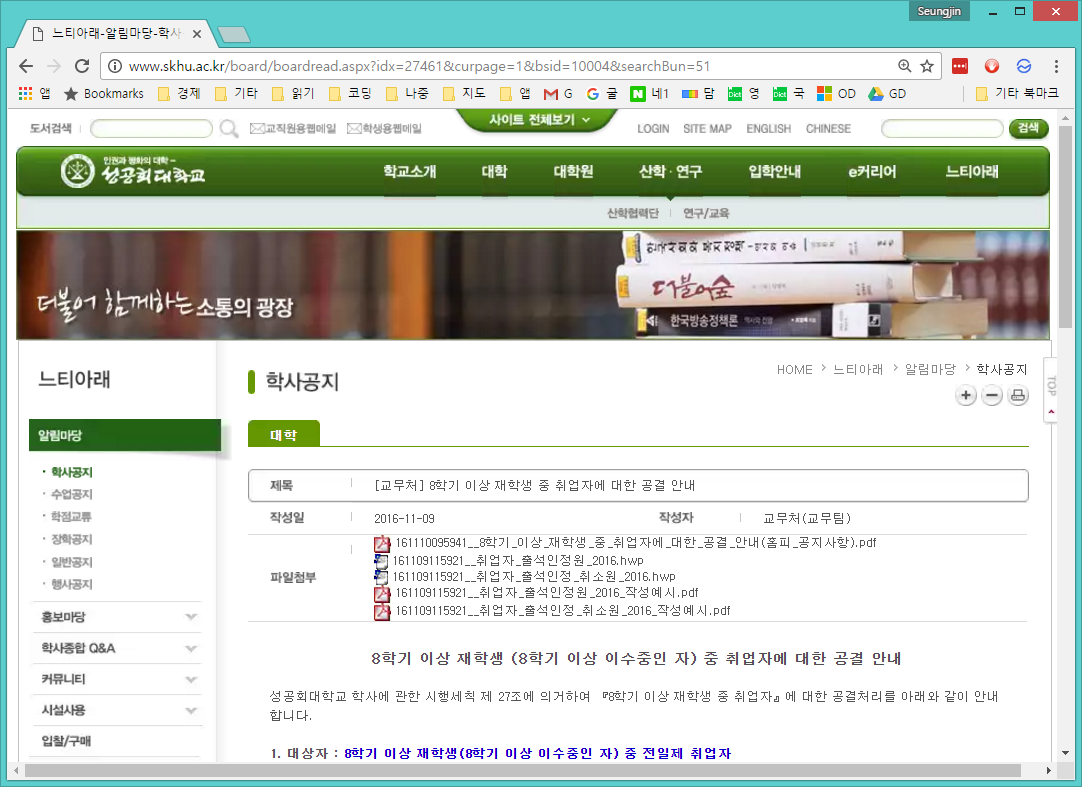
## 게시글 내용 웹페이지

아래의 URL은 게시글 내용 웹페이지의 URL 이다.

http://www.skhu.ac.kr/board/boardread.aspx?idx=27461&curpage=1&bsid=10004&searchBun=51

위 URL에서 idx=27461 부분은 게시글의 아이디이다.

내용을 보고 싶은 게시글의 아이디 값이 URL에 포함되어야 한다.



## 게시글 내용 태그 분석

|  |
| --- |
| ... 생략 ...  <td colspan="4" class="board\_view\_con">  ... 게시글 내용 ...  </td>  </tr>  </tbody>  </table>  <table summary="이전글, 다음글 구분하여 이전다음글 목록을 보여줍니다." class="prev\_next">  ... 생략 ... |

게시글의 시작 부분은 다음과 같다.

<td colspan="4" class="board\_view\_con">

... 게시글 내용 ...

이 부분을 찾기 위한 정규식은 다음과 같다.

<td colspan="4" class="board\_view\_con">

정규식 특수 문자가 포함되어 있지 않기 때문에, 찾을 문자열과 정규식이 동일하다.

위 정규식을 Java 문자열로 표현하면 다음과 같다.

"<td colspan=\"4\" class=\"board\_view\_con\">"

Java 문자열에 포함된 " 문자는 \" 형태이어야 한다.

게시글의 끝 부분은 다음과 같다.

... 게시글 내용 ...

</td>

</tr>

</tbody>

</table>

<table summary="이전글, 다음글 구분하여 이전다음글 목록을 보여줍니다." class="prev\_next">

이 부분을 찾기 위한 정규식은 다음과 같다.

</td>\n </tr>\n </tbody>\n</table>\n<table summary="이전글

줄바꿈 문자만 \n 문자로 바뀌었고 나머지는 정규식 특수 문자가 아니라서 찾을 문자열과 동일하다.

위 정규식을 Java 문자열로 표현하면 다음과 같다.

" </td>\n </tr>\n </tbody>\n</table>\n<table summary=\"이전글"

Java 문자열에 포함된 " 문자는 \" 형태이어야 한다.

게시글 보기 웹페이지에서 게시글 내용을 찾기 위한 정규식을 Java 문자열로 표현하면 다음과 같다.

<td colspan="4" class="board\_view\_con">(.+)\n </td>\n </tr>\n </tbody>\n</table>\n<table summary="이전글

위 정규식을 Java 문자열로 표현하면 다음과 같다.

"<td colspan=\"4\" class=\"board\_view\_con\">(.+)\n </td>\n </tr>\n </tbody>\n</table>\n<table summary=\"이전글"

위 Java 문자열을 알아보기 쉽게 다음과 같이 나눌 수 있다.

"<td colspan=\"4\" class=\"board\_view\_con\">(.+)\n" +

" </td>\n" +

" </tr>\n" +

" </tbody>\n" +

"</table>\n" +

"<table summary=\"이전글"

## Example05

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  public class Example05 {  public static void main(String[] args) throws Exception {  String articleUrl =  "http://www.skhu.ac.kr/board/boardread.aspx?idx=27461&curpage=1&bsid=10004&searchBun=51";  String html = getHttpResponse(articleUrl);  System.out.println(html);  }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |

Example05 예제 코드는

성공회대학교 홈페이지 웹서버에 URL을 요청하고, 그 결과를 받아서,

받아 온 웹페이지 내용 전체를 화면에 출력한다.

아래의 URL은 성공회대학교 홈페이지 게시판의 게시글 보기 URL 이다.

"http://www.skhu.ac.kr/board/boardread.aspx?idx=27461&curpage=1&bsid=10004&searchBun=51"

idx=27461 부분은 게시글의 아이디 이다.

## Example06

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Example06 {  public static void main(String[] args) throws Exception {  String articleUrl =  "http://www.skhu.ac.kr/board/boardread.aspx?idx=27461&curpage=1&bsid=10004&searchBun=51";  String html = getHttpResponse(articleUrl);    String regex = "<td colspan=\"4\" class=\"board\_view\_con\">(.+)\n" +  " </td>\n" +  " </tr>\n" +  " </tbody>\n" +  "</table>\n" +  "<table summary=\"이전글";  Pattern pattern = Pattern.compile(regex, Pattern.DOTALL);  Matcher matcher = pattern.matcher(html);  if (matcher.find())  System.out.println(matcher.group(1));    }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |

Example06 예제 코드는 게시글 보기 웹페이지에서

게시글 내용 부분만 찾아서 화면에 출력한다.

Pattern pattern = Pattern.compile(regex, Pattern.DOTALL);

정규식에서 . 특수 문자가 줄바꿈 문자(\n)도 일치할 수 있으려면 위와 같은 코드가 필요하다.

## Example07

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Example07 {  public static void main(String[] args) throws Exception {  String body = getArticleBody("27461");  System.out.println(body);  }    static String getArticleBody(String idx) throws IOException {  String articleUrl =  "http://www.skhu.ac.kr/board/boardread.aspx?idx=9999&curpage=1&bsid=10004&searchBun=51";  articleUrl = articleUrl.replace("9999", idx);  String html = getHttpResponse(articleUrl);    String regex = "<td colspan=\"4\" class=\"board\_view\_con\">(?<body>.+)\n" +  " </td>\n" +  " </tr>\n" +  " </tbody>\n" +  "</table>\n" +  "<table summary=\"이전글";  Pattern pattern = Pattern.compile(regex, Pattern.DOTALL);  Matcher matcher = pattern.matcher(html);  return matcher.find() ? matcher.group("body") : null;  }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |

getArticleBody 메소드

이 메소드의 파라미터는 게시글 아이디이다.

이 아이디의 게시글을 웹서버로부터 받아서, 받은 내용 중에서, 게시글의 본문 부분을 찾아서

문자열로 리턴한다.

## Example08

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.BufferedWriter;  import java.io.FileOutputStream;  import java.io.IOException;  import java.io.InputStreamReader;  import java.io.OutputStreamWriter;  import java.net.HttpURLConnection;  import java.net.URL;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Example08 {  public static void main(String[] args) throws Exception {  String body = getArticleBody("27461");  writeTextFile("d:/temp/27461.html", body, "EUC-KR");  }    static void writeTextFile(String path, String text, String encoding) throws IOException {  try (BufferedWriter writer =  new BufferedWriter(new OutputStreamWriter(new FileOutputStream(path), encoding))) {  writer.write(text);  writer.close();  }  }    static String getArticleBody(String idx) throws IOException {  String articleUrl =  "http://www.skhu.ac.kr/board/boardread.aspx?idx=9999&curpage=1&bsid=10004&searchBun=51";  articleUrl = articleUrl.replace("9999", idx);  String html = getHttpResponse(articleUrl);    String regex = "<td colspan=\"4\" class=\"board\_view\_con\">(?<body>.+)\n" +  " </td>\n" +  " </tr>\n" +  " </tbody>\n" +  "</table>\n" +  "<table summary=\"이전글";  Pattern pattern = Pattern.compile(regex, Pattern.DOTALL);  Matcher matcher = pattern.matcher(html);  return matcher.find() ? matcher.group("body") : null;  }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |

Example08 예제 코드는 게시글 보기 웹페이지에서

게시글 내용 부분만 찾아서 텍스트 파일에 저장한다.

writeTextFile(String path, String text, String encoding) 메소드

문자열을 텍스트 파일에 저장하는 메소드이다.

path: 텍스트 파일의 경로명

text: 저장할 문자열

encoding: 텍스트 파일의 문자 인코딩

## Example09

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91 | package net.skhu.http;  import java.io.BufferedReader;  import java.io.BufferedWriter;  import java.io.FileOutputStream;  import java.io.IOException;  import java.io.InputStreamReader;  import java.io.OutputStreamWriter;  import java.net.HttpURLConnection;  import java.net.URL;  import java.util.ArrayList;  import java.util.List;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Example09 {  public static void main(String[] args) throws Exception {  String listUrl = "http://www.skhu.ac.kr/board/boardlist.aspx?curpage=1&bsid=10004&searchBun=51";  List<Article> list = getArticleList(listUrl);  for (Article article : list) {  String idx = article.getIdx();  String body = getArticleBody(idx);  article.setBody(body);  String path = "d:/temp/##.html".replace("##", idx);  writeTextFile(path, body);  }  }    static void writeTextFile(String path, String text) throws IOException {  try (BufferedWriter writer =  new BufferedWriter(new OutputStreamWriter(new FileOutputStream(path), "EUC-KR"))) {  writer.write(text);  writer.close();  }  }    static List<Article> getArticleList(String url) throws IOException {  List<Article> list = new ArrayList<Article>();  String html = getHttpResponse(url);  String regex = "<a href='boardread.aspx\\?idx=([0-9]+)[^>]+>([^<]+)</a>";  Pattern pattern = Pattern.compile(regex);  Matcher matcher = pattern.matcher(html);  while (matcher.find()) {  Article article = new Article();  article.setIdx(matcher.group(1));  article.setTitle(matcher.group(2));  list.add(article);  }  return list;  }    static String getArticleBody(String idx) throws IOException {  String articleUrl =  "http://www.skhu.ac.kr/board/boardread.aspx?idx=9999&curpage=1&bsid=10004&searchBun=51";  articleUrl = articleUrl.replace("9999", idx);  String html = getHttpResponse(articleUrl);    String regex = "<td colspan=\"4\" class=\"board\_view\_con\">(?<body>.+)\n" +  " </td>\n" +  " </tr>\n" +  " </tbody>\n" +  "</table>\n" +  "<table summary=\"이전글";  Pattern pattern = Pattern.compile(regex, Pattern.DOTALL);  Matcher matcher = pattern.matcher(html);  return matcher.find() ? matcher.group("body") : null;  }  static String getHttpResponse(String url) throws IOException {  URL urlObj = new URL(url);  HttpURLConnection connection = (HttpURLConnection)urlObj.openConnection();  connection.setRequestMethod("GET");  connection.setRequestProperty("User-Agent", "Mozilla/5.0");  String contentType = connection.getContentType();  String encoding = (contentType.toUpperCase().indexOf("UTF-8") >= 0) ? "UTF-8" : "EUC-KR";  StringBuffer builder = new StringBuffer();  try (BufferedReader input =  new BufferedReader(new InputStreamReader(connection.getInputStream(), encoding))) {  String s;  while ((s = input.readLine()) != null) {  builder.append(s);  builder.append('\n');  }  }  return builder.toString();  }  } |