

Tiến trình URL trong Java

URL là viết tắt của Uniform Resource Locator và biểu diễn một nguồn trên World Wide Web, ví dụ như một trang Web hoặc thư mục FTP.

Để hiểu sâu hơn các khái niệm được trình bày trong chương này, mời bạn tham khảo loạt bài: [Ví dụ về Lập trình mạng \(Networking\) trong Java](#).

Chương này hướng dẫn bạn cách viết các chương trình Java mà giao tiếp với một URL. Một URL có thể được phân chia thành các phần như sau:

```
protocol://host:port/path?query#ref
```

Ví dụ về Protocol bao gồm HTTP, HTTPS, FTP, và File. path cũng được xem như là filename, và host cũng được gọi như là authority.

Sau đây là một URL tới một trang Web mà protocol là HTTP:

```
http://www.amrood.com/index.htm?language=en#j2se
```

Bạn chú ý rằng URL này không xác định một cổng (port), trong trường hợp này cổng mặc định cho protocol được sử dụng. Với HTTP, cổng mặc định là 80.

Các phương thức của lớp URL trong Java

Lớp **java.net.URL** biểu diễn một URL và có tập hợp đầy đủ các phương thức để thao tác URL trong Java.

Lớp URL có một số constructor để tạo các URL, bao gồm:

STT	Phương thức và Miêu tả
1	public URL(String protocol, String host, int port, String file) throws MalformedURLException. Tạo một URL bằng việc đặt các phần đã cho cùng với nhau
2	public URL(String protocol, String host, String file) throws MalformedURLException

	Giống constructor trước, ngoại trừ việc cổng mặc định cho protocol được sử dụng
3	public URL(String url) throws MalformedURLException Tạo một URL từ String đã cho
4	public URL(URL context, String url) throws MalformedURLException Tạo một URL bởi việc parse các tham số URL và String cùng nhau

Lớp URL chứa nhiều phương thức để truy cập các phần khác nhau của URL đang được biểu diễn.

Dưới đây là một số phương thức trong lớp URL:

STT	Phương thức và Miêu tả
1	public String getPath() Trả về path của URL đó
2	public String getQuery() Trả về phần query của URL đó
3	public String getAuthority() Trả về authority của URL đó
4	public int getPort() Trả về port của URL đó
5	public int getDefaultPort() Trả về port mặc định cho protocol của URL đó
6	public String getProtocol() Trả về protocol của URL đó
7	public String getHost() Trả về host của URL đó

8	public String getFile() Trả về filename của URL đó
9	public String getRef() Trả về phần reference của URL đó
10	public URLConnection openConnection() throws IOException Mở một kết nối tới URL, cho phép một client giao tiếp với nguồn

Ví dụ:

Chương trình URLEDemo sau minh họa các phần khác nhau của một URL. Một URL được nhập trên command line, và chương trình URLEDemo này đưa ra kết quả là mỗi phần của URL đã cho.

```
// File Name : URLEDemo.java

import java.net.*;
import java.io.*;

public class URLEDemo
{
    public static void main(String [] args)
    {
        try
        {
            URL url = new URL("http://www.amrood.com/index.htm?language=en#j2se");
            System.out.println("URL is " + url.toString());
            System.out.println("protocol is "
                               + url.getProtocol());
            System.out.println("authority is "
                               + url.getAuthority());
            System.out.println("file name is " + url.getFile());
            System.out.println("host is " + url.getHost());
            System.out.println("path is " + url.getPath());
        }
    }
}
```

```
        System.out.println("port is " + url.getPort());
        System.out.println("default port is "
                           + url.getDefaultPort());
        System.out.println("query is " + url.getQuery());
        System.out.println("ref is " + url.getRef());
    } catch (IOException e)
    {
        e.printStackTrace();
    }
}
```

Chạy mẫu ví dụ trên sẽ cho kết quả sau:

```
URL is http://www.amrood.com/index.htm?language=en#j2se
protocol is http
authority is www.amrood.com
file name is /index.htm?language=en
host is www.amrood.com
path is /index.htm
port is -1
default port is 80
query is language=en
ref is j2se
```

Các phương thức lớp URLConnection trong Java

Phương thức `openConnection()` trả về một **java.net.URLConnection**, một lớp abstract mà các lớp phụ của nó biểu diễn các kiểu kết nối URL đa dạng.

Ví dụ:

- Nếu bạn kết nối tới một URL mà protocol của nó là HTTP, thì phương thức `openConnection()` trả về một đối tượng `HttpURLConnection`.
- Nếu bạn kết nối tới một URL mà biểu diễn một JAR file, thì phương thức `openConnection()` trả về một đối tượng `JarURLConnection`.

- etc...

Lớp `URLConnection` có nhiều phương thức để thiết lập và xác định thông tin về connection, bao gồm:

STT	Phương thức và Miêu tả
1	Object getContent() Thu nhận nội dung của URL connection này
2	Object getContent(Class[] classes) Thu nhận nội dung của URL connection này
3	String getContentEncoding() Trả về giá trị của trường content-encoding header
4	int getContentLength() Trả về giá trị của trường content-length header
5	String getContentType() Trả về giá trị của trường content-type header
6	int getLastModified() Trả về giá trị của trường last-modified header
7	long getExpiration() Trả về giá trị của trường expires header
8	long getIfModifiedSince() Trả về giá trị của trường ifModifiedSince của đối tượng này
9	public void setDoInput(boolean input) Truyền là true để biểu thị rằng connection sẽ được sử dụng cho input. Giá trị mặc định là true bởi vì Client đọc từ một <code>URLConnection</code> .

10	public void setDoOutput(boolean output) Truyền là true để biểu thị rằng connection sẽ được sử dụng cho output. Giá trị mặc định là false bởi vì nhiều kiểu URL không hỗ trợ để được ghi trên đó
11	public InputStream getInputStream() throws IOException Trả về Input Stream của URL connection để đọc từ nguồn
12	public OutputStream getOutputStream() throws IOException Trả về Output Stream của URL connection để ghi từ nguồn
13	public URL getURL() Trả về URL mà đối tượng URLConnection này được kết nối tới

Ví dụ:

Chương trình URLConnectionDemo sau kết nối tới một URL được nhập từ command line.

Nếu URL biểu diễn một nguồn HTTP, thì connection được ném tới HttpURLConnection, và dữ liệu trong nguồn được đọc từng dòng một tại một thời điểm.

```
// File Name : URLConnDemo.java

import java.net.*;
import java.io.*;

public class URLConnDemo
{
    public static void main(String [] args)
    {
        try
        {
            URL url = new URL("http://www.amrood.com");
            URLConnection urlConnection = url.openConnection();
            HttpURLConnection connection = null;
            if(urlConnection instanceof HttpURLConnection)
            {
```

```
        connection = (HttpURLConnection) urlConnection;
    }
    else
    {
        System.out.println("Please enter an HTTP URL.");
        return;
    }

    BufferedReader in = new BufferedReader(
        new InputStreamReader(connection.getInputStream()));
    String urlString = "";
    String current;
    while((current = in.readLine()) != null)
    {
        urlString += current;
    }
    System.out.println(urlString);
} catch(IOException e)
{
    e.printStackTrace();
}
}
```

Chạy mẫu chương trình trên sẽ cho kết quả:

```
$ java URLConnDemo

.....a complete HTML content of home page of amrood.com.....
```