JAVA 프로그래밍

Chapter 18 네트워크 프로그래밍

목차

- 네트워크 정의
- TCP 통신
- URL 통신
- UDP 통신
- Multicast 통신

네트워크 개념

- ❖ 네트워크(Network)
  - 데이터 교환을 목적으로 로컬 컴퓨터와 원격 컴퓨터 간에 데이터의 흐름을 나타내는 구조
- ❖ IP(Internet Protocol)
  - 인터넷 주소라 불리는 유일한 32비트 숫자로 구성된 주소체계.(221.214.3.102)
- ❖ 포트(PORT)
  - 프로그램에서 사용되는 논리적인 접속 장소
  - 80(HTTP), 21(FTP), 22(SSH), 23(TELNET)등이 있다.
  - 포트번호는 0~65535까지 이며, 0~1023까지는 시스템에 의해 예약된 포트번호이기 때문에 될 수 있는 한 사용하지 않는 것이 바람직하다.
- ❖ 프로토콜(PROTOCOL)
  - 클라이언트와 서버간의 통신 규약
  - 상호간의 접속이나 절단방식, 통신방식, 데이터의 형식, 전송속도 등에 대하여 정하는 것.

네트워크 개념

❖ TCP/IP(Transmission Control Protocol/Internet Protocol)

- 인터넷상에서 호스트들을 서로 연결시키는데 사용하는 통신 프로토콜로 기종이 서로 다른 컴퓨터 시스템을 서로 연결해 데이터를 전송하기 위한 통신 프로토콜.

❖ UDP(User Datagram Protocol)

- IP를 사용하는 네트워크 내에서 컴퓨터들 간에 메시지들을 교환할 때 제한된 서비스만을 제공하는 통신 프로토콜.

❖ URL(Uniform Resource Locator)

- 인터넷상에 있는 각종 정보들의 위치를 나타내는 표준이다.

❖ URI(Uniform Resource Identifier)

- 특정 자원에 접근하기 위한 형식이나 고유한 이름으로 URL보다 넓은 의미의 개념이다.

네트워크 개념

- Broadcast
  - 데이터를 여러 방향으로 동시에 전송하여 동일 IP그룹에 있는 컴퓨터라면 데이터를 수신할 수 있는 방식이다.
- Unicast
  - 특정한 대상 수신자에게만 보내는 방식이다.
- Multicast
  - 다중의 수신 대상자들에게 보내는 방식이다.
- RMI(Remote Method Invocation)
  - 자바 프로그래밍 언어와 개발 환경을 사용하여 서로 다른 컴퓨터 상에 있는 객체들이 분산 네트워크 내에서 상호 작용하는 객체지향형 프로그램을 작성할 수 있도록 해주는 방식이다. RMI는 일반적으로 RPC라고 알려진 것의 자바 버전이다.

TCP 통신을 위한 클래스들

- InetAddress 클래스
- URL, URLConnection 클래스
- Socket 클래스
- ServerSocket 클래스

InetAddress 클래스

❖ 정의: IP주소를 표현한 클래스

반환형	메서드	설명
InetAddress[]	getAllByName(String host)	매개변수 host에 대응되는 InetAddress 배열을 반환한다.
InetAddress	getByAddress(byte[] addr)	매개변수 addr에 대응되는 InetAddress 객체를 반환한다.
	getByAddress(String host, byte[] addr)	매개변수 host와 addr로 Inetaddress객체를 생성한다.
	getByName(String host)	매개변수 host에 대응되는 InetAddress 객체를 반환한다.
	getLocalHost()	로컬호스트의 InetAddress 객체를 반환한다.

반환형	메서드	설명
byte[]	getAddress()	InetAddress 객체의 실제 IP 주소를 바이트 배열로 리턴한다.
String	getHostAddress()	IP 주소를 문자열로 반환한다.
	getHostName()	호스트 이름을 문자열로 반환한다.
	toString()	IP 주소를 스트링 문자열로 오버라이딩 한 메소드

URL(Uniform Resource Locator) 통신

❖ URL 클래스

- URL : 인터넷에서 접근 가능한 자원(Resource)의 주소를 표현할 수 있는 형식.
- URL을 추상화 하여 만든 클래스

```
<protocol>://<host>:<port>/<path>?<query>#<reference>
http://www.daum.net:80/member/mem.jsp?name=sung#content
```

❖ URLConnection 클래스

- 원격자원에 접근하는데 필요한 정보를 갖고 있다.
- 원격서버의 헤더 정보, 해당 자원의 길이와 타입정보, 언어 등을 얻어 올 수 있다.
- 추상화 클래스 이므로 URL 클래스의 openConnection() 메소드를 이용 객체 생성.
- URLConnection 클래스의 connect()메소드를 호출해야 객체가 완성됨.

```
URL  url  =  new URL("http://java.sun.com");
URLConnection  urlCon  =  url.openConnection();
urlCon.connect();
```

Socket/ServerSocket 클래스

❖ Socket 클래스

- 서버 프로그램으로 연결 요청을 한다.
- 데이터 전송을 담당한다.

[표 15-7] Socket 클래스의 주요 생성자

생성자	설명
Socket(InetAddress address, int port)	InetAddress 객체와 port를 이용하여 Socket 객체를 생성한다.
Socket(String host , int port)	host와 port를 이용하여 Socket 객체를 생성한다.

Socket/ServerSocket 클래스

❖ Socket 클래스

[표 15-8] Socket 클래스의 주요 메서드

반환형	메서드	설명
void	close( )	소켓 객체를 닫는다.
InetAddress	getInetAddress( )	소켓 객체를 InetAddress 객체로 반환한다.
InputStream	getInputStream( )	소켓 객체로부터 입력할 수 있는 InputStream 객체를 반환한다.
InetAddress	getLocalAddress( )	소켓 객체의 로컬 주소를 반환한다.
int	getPort( )	소켓 객체의 포트를 반환한다.
boolean	isClosed( )	소켓 객체가 닫혀있으면 true를, 열려있으면 false를 반환한다.
	isConnected( )	소켓 객체가 연결되어 있으면 true, 연결되어 있지 않으면 false를 반환한다.
void	setSoTimeout(int timeout)	소켓 객체의 시간을 밀리 세컨드로 설정한다.

Socket/ServerSocket 클래스

❖ ServerSocket 클래스

- 서버 프로그램에서 사용하는 소켓
- 포트를 통해 연결 요청이 오기를 대기
- 요청이 오면 클라이언트와 연결을 맺고 또 다른 소켓을 만드는 일을 한다.
- 새로 만들어진 소켓은 클라이언트 소켓과 데이터를 주고 받는다.

[표 15-9] ServerSocket 클래스의 주요 생성자

생성자	설명
ServerSocket(int port)	port를 이용하여 ServerSocket 객체를 생성한다.

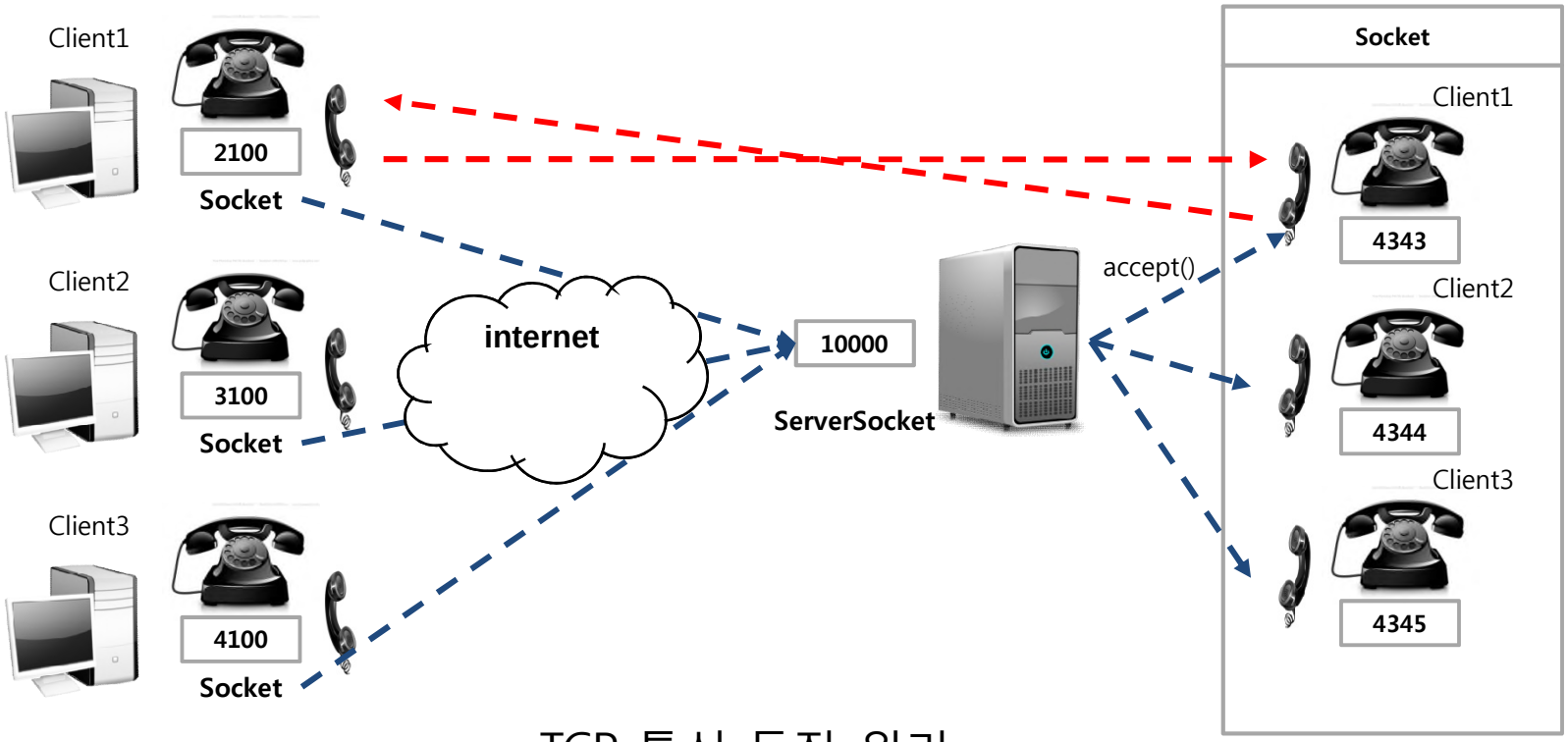
Socket/ServerSocket 클래스

❖ ServerSocket 클래스

[표 15-10] ServerSocket 클래스의 주요 메서드

반환형	메서드	설명
Socket	accept( )	클라이언트의 Socket 객체가 생성될 때까지 블로킹되는 메서드다. 클라이언트의 Socket 객체가 생성되면 서버에서 클라이언트와 통신할 수 있는 Socket 객체를 반환하게 된다.
void	close( )	ServerSocket 객체를 닫는다.
int	getLocalPort( )	ServerSocket 객체가 청취하고 있는 포트번호를 반환한다.
	getSoTimeout( )	ServerSocket 클래스의 accept( ) 메서드가 유효할 수 있는 시간을 밀리 세컨드로 반환한다. 만약, 0이면 무한대를 의미한다.
boolean	isClosed( )	ServerSocket 객체의 닫힌 상태를 반환한다.
void	setSoTimeout(int timeout)	ServerSocket 클래스의 accept( ) 메서드가 유효할 수 있는 시간을 밀리 세컨드로 설정해야 한다. 만약, 시간이 지나면 java.net.SocketTimeoutException 예외가 발생하는데, 이 예외가 발생하더라도 ServerSocket 객체는 계속 유효하다.

TCP 통신 동작 원리



TCP 통신 동작 원리

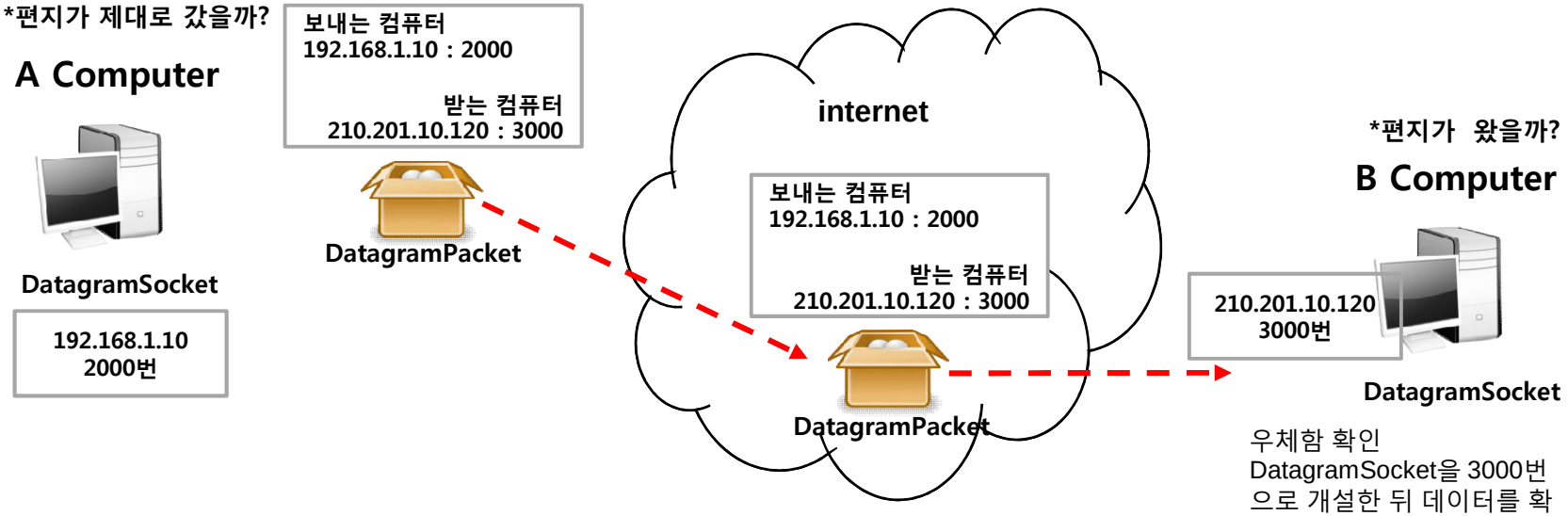
TCP 통신 동작 원리

- ServerSocket
  - ServerSocket은 클라이언트와 통신할 수 있는 서버용 Socket을 만들어 준다.
- ServerSocket의 생성과 접속 대기
  - ServerSocket ss = new ServerSocket(10000);
  - Socket socket = ss.accept();
- ServerSocket의 accept()에서 리턴된 서버용 Socket
  - 클라이언트 Socket으로 ServerSocket에 연결요청할 때 accept()가 반응한다.
  - ServerSocket이 accept()할 때 리턴된 서버용 Socket은 해당 클라이언트와 통신할 수 있는 유일한 수단이다.
  - accept()할 때 리턴된 서버용 Socket은 자동으로 포트(Port)를 할당받는다.
- 서버용 Socket의 생성 및 스트림 개설
  - ServerSocket ss = new ServerSocket(10000);
  - Socket socket = ss.accept();
  - InputStream is = socket.getInputStream();
  - OutputStream os = socket.getOutputStream();

UDP 통신

❖ UDP(User Datagram Protocol)

- 비연결지향적이다.
- 데이터의 신뢰성을 보장 하지 않는다.
- TCP에 비해 전송 속도가 빠르다.

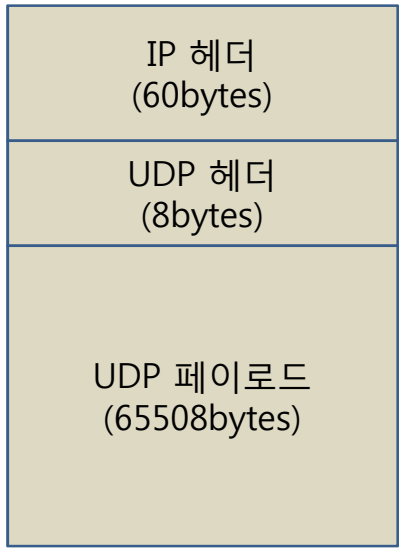


UDP 통신 동작 원리

UDP 통신

❖ UDP Datagram 구조

- IP 헤더는 패킷의 발신지와 목적지 주소, 길이, 체크섬, TTL 그리고 다른 IP 옵션들이 포함된다.
- UDP 헤더는 발신지 포트와 목적지 포트, 헤더를 포함하는 길이, 체크섬으로 구성된다.
- UDP Payload에는 실제 데이터가 전송되는 공간으로 실제 데이터가 들어가 있다. 그러나 일반적으로 512바이트로 제한하는 경우가 많다.



UDP 통신 관련 클래스

❖ DatagramPacket 클래스

- UDP 데이터그램을 추상화 한 클래스.
- 애플리케이션에서 주고받을 데이터와 관련된 클래스.
- 데이터를 송신기능과 수신 기능으로 크게 분리된다.
- 출발지 주소와 목적지 주소를 설정하거나 주소를 얻어오는 메소드 제공
- 출발지 포트와 목적지 포트를 설정하거나 포트를 얻어오는 메소드 제공

❖ DatagramSocket 클래스

- 실제 데이터의 전송을 책임진다.
- 데이터그램의 전송과 수신에서 사용할 수 있다.
- DatagramPacket을 보내거나 받을 수 있는 메소드를 제공한다.

UDP 통신 관련 클래스

❖ DatagramPacket 클래스

- DatagramPacket의 생성자는 데이터를 보내기 위한 생성자와 데이터를 받기 위한 생성자로 구분된다.

[표 15-11] DatagramPacket 클래스의 주요 생성자

생성자	설명
DatagramPacket(byte[ ] buf, int length)	데이터를 수신하기 위한 생성자로 바이트 배열 buf의 length 만큼 저장한다.
DatagramPacket(byte[ ] buf, int length, InetAddress address, int port)	데이터를 송신하기 위한 생성자로 address와 port로 바이트 배열 buf의 length만큼 저장한다.
DatagramPacket(byte[ ] buf, int offset, int length)	데이터를 수신하기 위한 생성자로 바이트 배열 buf의 offset 위치에서 length만큼 저장한다.
DatagramPacket(byte[ ] buf, int offset, intlength, InetAddress address, int port)	데이터를 송신하기 위한 생성자로 address와 port로 바이트 배열 buf의 offset 위치에서 length만큼 저장한다.

UDP 통신 관련 클래스

❖ DatagramPacket 클래스의 주요 메소드

- IP 헤더에 출발지 주소와 목적지 주소를 설정하거나 주소를 얻어오는 메서드, 출발지 포트와 목적지 포트를 설정하거나 얻어오는 다양한 메서드 제공한다.

[표 15-12] DatagramPacket 클래스의 주요 메서드

반환형	메서드	설명
InetAddress	getAddress( )	데이터그램에 대한 목적지 또는 출발지 주소를 반환한다.
byte[ ]	getData( )	버퍼에 들어있는 실제 데이터를 바이트 배열로 반환한다.
int	getLength( )	버퍼에 들어있는 실제 데이터의 길이를 반환한다.
	getOffset( )	버퍼에 들어있는 실제 데이터의 시작 위치를 반환한다
	getPort( )	데이터그램에 대한 목적지 또는 출발지 포트를 반환한다.
void	setAddress (InetAddress iaddr)	데이터그램을 보낸 호스트 주소를 설정한다.
	setData(byte[ ] buf)	버퍼에 들어있는 실제 데이터를 바이트 배열 buffer로 설정한다.
void	setData(byte[ ] buf, int offset, int length)	버퍼에 들어있는 실제 데이터를 바이트 배열 buffer의 offset 위치에서 length만큼 설정한다.
	setLength(int length)	버퍼에 들어있는 실제 데이터의 길이를 설정한다.
	setPort(int port)	데이터그램에 대한 목적지 또는 출발지 포트를 설정한다.

UDP 통신 관련 클래스

❖ DatagramSocket 클래스

- TCP 스트림 소켓과 달리 서버와 클라이언트 데이터 그램 소켓 사이에는 차이가 없으며 모든 데이터그램 소켓은 데이터그램을 전송할 뿐만 아니라 수신에서 사용할 수 있다.
- 모든 DatagramSocket 객체는 데이터그램을 수신하기 위해서 사용될 수 있기 때문에 로컬 호스트 내의 유일한 UDP 포트와 연관되어 있다.

[표 15-13] DatagramSocket 클래스의 주요 생성자

생성자	설명
DatagramSocket( )	할당된 특정한 포트번호가 중요하지 않다면 사용 가능한 임시 UDP 포트로 소켓을 생성하여 DatagramSocket 객체를 생성한다.
DatagramSocket(int port)	매개변수 포트로 소켓을 생성하여 DatagramSocket 객체를 생성한다.
DatagramSocket(int port, InetAddress iaddr)	매개변수 port와 iaddr로 소켓을 생성하여 DatagramSocket 객체를 생성한다.

UDP 통신 관련 클래스

❖ DatagramSocket 클래스

- DatagramSocket 클래스의 주요 메서드 기능은 DatagramPacket을 보내거나 받을 수 있는 메서드를 제공하는 것이다.

[표 15-14] DatagramSocket 클래스의 주요 메서드

반환형	메서드	설명
void	send(DatagramPacket dp)	UDP 데이터그램(dp)을 전송하는 메서드다.
	receive(DatagramPacket dp)	UDP 데이터그램을 받아서 이미 존재하는 DatagramPacket 객체인 dp에 저장한다.
	close( )	데이터그램 소켓이 점유하고 있는 포트를 자유롭게 놓아준다.
int	getLocalPort( )	현재 소켓이 데이터그램을 기다리고 있는 로컬 포트가 몇 번인지를 리턴한다.
void	connect(InetAddress address, int port)	DatagramSocket이 지정된 호스트의 지정된 포트하고만 패킷을 주고받을 것이라고 정한다.
	disconnect( )	현재 연결된 DatagramSocket의 연결을 끊는다. 연결이 끊기면 아무것도 하지 못하게 된다.
int	getPort( )	DatagramSocket이 연결되어 있다면 소켓이 연결되어 있는 원격지 포트번호를 반환한다.
InetAddress	getInetAddress( )	DatagramSocket이 연결되어 있다면 소켓이 연결되어 있는 원격지 주소를 반환한다.

Multicast 통신

- UDP 통신은Unicast, Broadcast, Multicast의 구분
- Multcast를 위한 자바 클래스들
  - DatagramPacket 클래스
  - MulticastSocket 클래스