

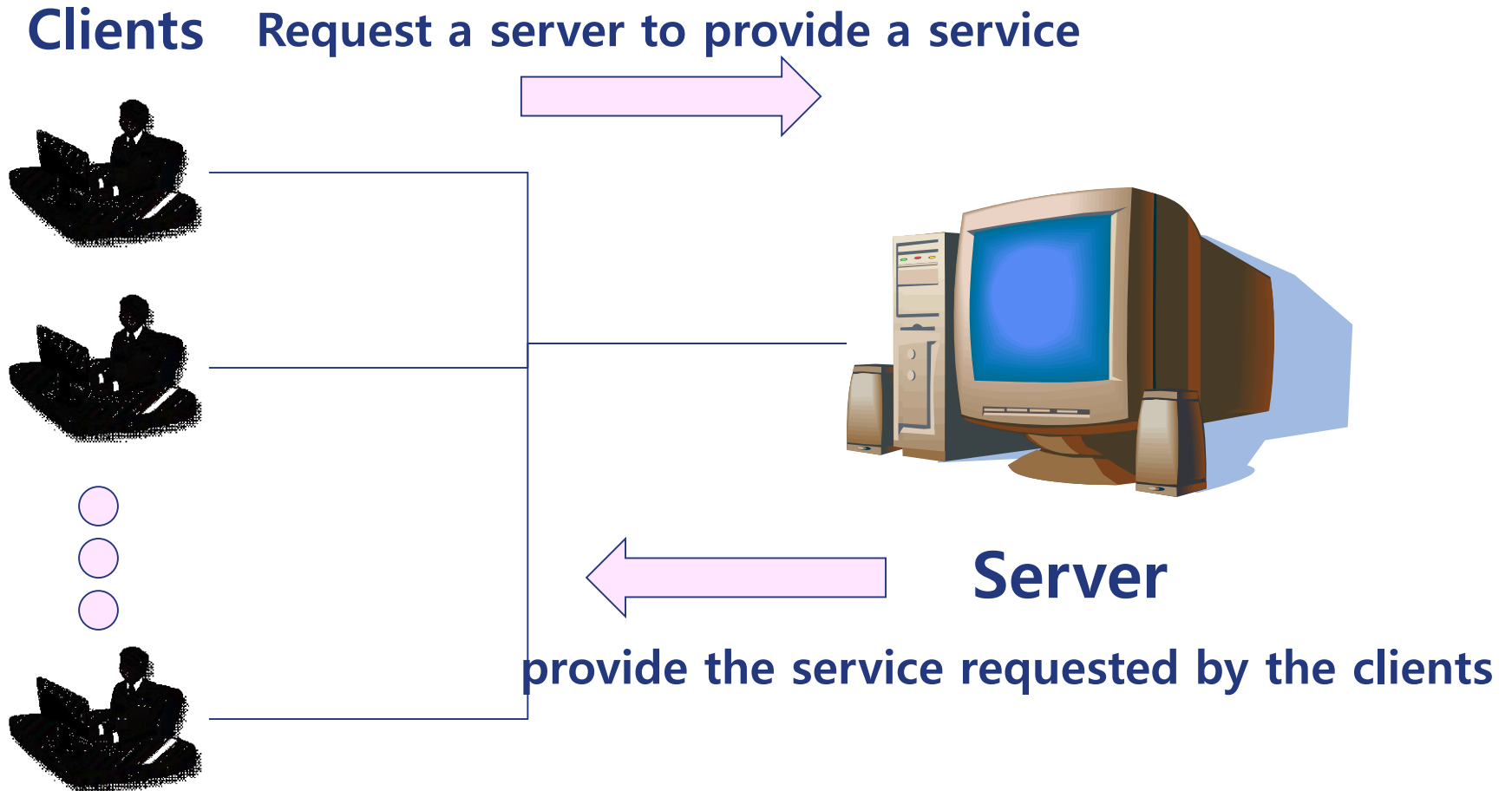
# Network Programming

- ❖ Client-Server Model
- ❖ Host and Port
- ❖ Socket
  
- ❖ Implementing Client
- ❖ Implementing Server
- ❖ Implementing Server for Multiple Clients
  
- ❖ Getting Web Data



# Client-Server Model

---



# Client-Server Model: Examples

---

- ❖ Many services are provided in the form of Client-Server
  - Web-based Services
  - Games

# Web-based Services

학적 Register	교육과정 Curriculum	수업 Classes	성적 Grades	장학 Scholarships	등록 Enrollment	졸업 Graduation	타대교류 Exchange	대학생활 Campus Life	비교과 Extracurriculum	PASS PASS	포트폴리오 e-Portfolio
----------------	--------------------	---------------	--------------	--------------------	------------------	------------------	------------------	---------------------	------------------------	--------------	----------------------



## LOGIN



로그인

☐ ID 저장    · 아이디 찾기    · 비밀번호 찾기

- [2018/11/30 ~ 2018/12/14] 제1학기 재입학신청
- [2018/12/03 ~ 2018/12/04] 겨울계절수업 수강정정
- [2018/12/03 ~ 2018/12/04] 겨울계절수업 수강대상자 복학신청
- [2018/12/06 ~ 2018/12/10] 겨울계절수업 등록금납부

## 학생지원시스템



[교양교육원] 일반졸업영어(영어졸업인증 대체과목) 수강신청 안내

## 공지사항

## Q&A

## FAQ



- 2019학년도 학생회선거 안내
-  [학사과] 2018학년도 겨울계절수업 1차 폐강강좌 통보 및 1차 수강정정 안내
-  [학사과] 2018학년도 겨울계절수업 개설강좌 확정 및 수강신청 안내
- [교양교육원] 일반졸업영어(영어졸업인증 대체과목) 수강신청 안내

2018-11-27  
2018-11-20  
2018-11-05  
2018-11-05

## 개인화 바로가기 서비스




- 로그인 하셔야만 사용할 수 있습니다.

# Web-based Services

네이버를 시작해보시나요? | 홈네이버 | 네이버

**NAVER**

메일 카페 블로그 지식iN 쇼핑 Pay TV 사진 뉴스 증권 부동산 지도 영화 뮤직 책 웹툰 더보기 ▾ 9 볼타는 청춘 ▾



연합뉴스 > "얼차 타기 두려운 시경"...코메일 안전관리 '동체적' 난국

뉴스스탠드 > 전체 언론사 | MY 뉴스

YTN	OSEN	세계일보	KJD	전자신문	파이낸셜뉴스
일간스포츠	NEWSIS	SBS	머니투데이	한국경제TV	노컷뉴스
조인뉴스	밸스조선	국민일보	ECONOMY	과학동아	부산일보

TV연예 뮤직 영화 책문화 함께IN 여행+ 디자인 경제M JOB& 과학 중국 < >

네이버를 더 편리하고 편리하게 이용하세요.

**NAVER 로그인**

아이디·비밀번호 찾기 회원가입

11.28. (수) 부산시 구서2동 > 1/6 < >

8°C  
9°/16°  
오늘 오전  
16°C  
오늘 오후

날씨 실시간 기상 정보

다음을 시작해보시나요?

**Daum**


도깨비 세명 논란 이모미 이연 열애 무정사례가 파는곳 마켓 군야버지 이영현 부모님

카페 메일 뉴스 지도 증권 쇼핑 카카오TV 웹툰 블로그 브런치 사진 게임 별틀레이 같이가자 더보기 ▾


11. 28. (수) 경기도 5°C ▾

뉴스 >>> 연예 스포츠 영화 자동차 부동산


1/8 < >



518도 세월호도 희생자 가족 무력화 앞장선...



바닷바람이 좋다구요? 서해 섬에 가봤더니...



국내 오염물질에 황사까지. 출근길 공기... 통합대기

"현대차 온다" 땅값 집값 "들쭉"...임금 줄 것" 우려도

이재명 휴대전화 비밀번호 '답변' 잠금. 기소여부 내주 결론

한국당, 사립유치원 사유재산 인정' 가덕. 논란 예상

BMW 협력업체 줄도산 위기... '취어왔다'

문 대통령 사무장 출신 GKL 상임이사 선임... '낙하산' 논란

지하 5m의 비밀. 과연 화성에서 인간은 살 수 있나?

'별'은 멀고 '남연 보복'은 가까웠다. '피명' 얼굴 못 본 척

EU 집행위 보고서 "EU 수출, 한국 내 일자리 16만2천개 지..."

실시간 이슈 검색어

- 1 사의 참미
- 2 노정철
- 3 반민정
- 4 홍현희
- 5 제이슨
- 6 김주혁
- 7 조목제
- 8 윤심덕
- 9 이해정
- 10 아내의 맛

다우존스 24,540.61 ▼ 99.63

로그인 상태 유지 IP보안 ON

아이디  비밀번호

로그인

회원가입 아이디·비밀번호 찾기 카카오 로그인

**Klud** 1월 10까지 하는 요즘. 안샀으면 어쩔뻔!

클릭 미니 마사지 지금 50% 할인

1/18 < >

쇼핑 > 실시간 쇼호 스토어 롯데닷컴

G마켓 · 11번가 · 옥션 · GS상 · Cmall · 위메프

하프클럽 · 롯데닷컴 · 패션플러스 · NS홈쇼핑

헤지스 · 아이소이 · 제이에스티나 · 시즌기획전

1/18 < >

5



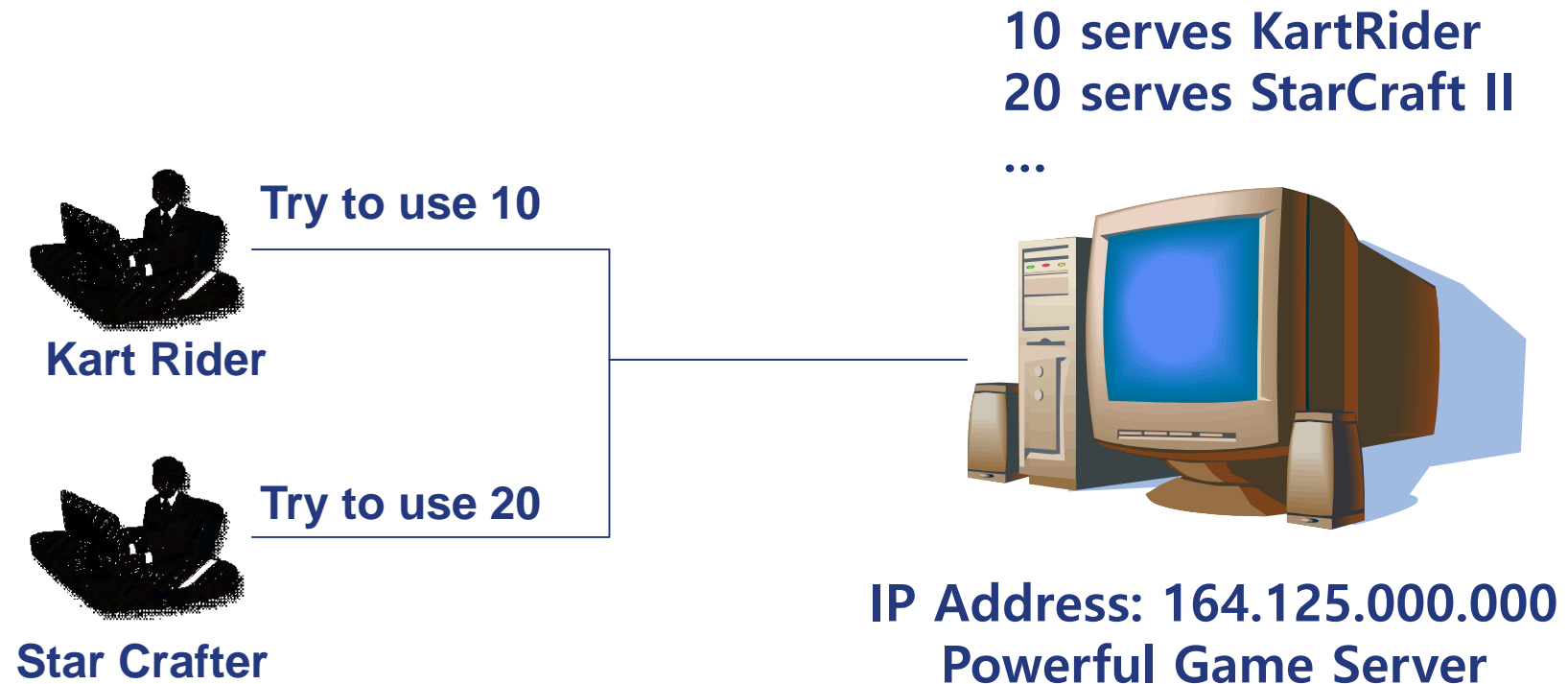
# Games



# Host and Port

---

- ❖ Host is used to address the server.
- ❖ Port is used to specify a particular service of a server



# Well-known Ports

---

- ❖ Many port numbers are reserved for standard services

Port	Service
7	Echo
13	DayTime
20, 21	FTP
22	Secure Shell
23	Telnet
25	SMTP
80	HTTP
111	Sun RPC
...	

- ❖ For a complete list, visit Port Numbers by The Internet Assigned Numbers Authority (IANA) (<http://www.iana.org/assignments/port-numbers>)



# DayTime Service at 13

---

Microsoft Telnet> op time-A.nist.gov 13

연결 대상 time-A.nist.gov...

59123 20-10-01 16:56:35 32 0 0 844.2 UTC(NIST) \*

호스트에 대한 연결을 잃었습니다.

계속 진행하려면 아무 키나 누르십시오...

# Socket

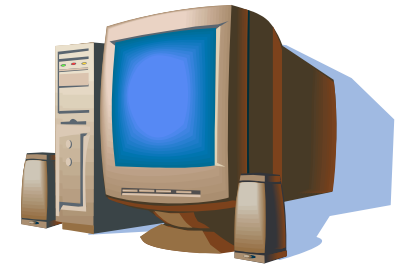
---

- ❖ Socket indicates the end point of communication between client and server
- ❖ Socket consists of host and port



## Client Socket

- host: client host
- Port: client port



## Server Socket

- Host: time-A.timefreq.bldrdoc.gov
- Port: 13

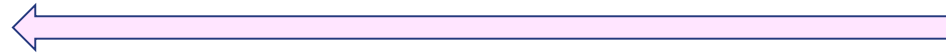
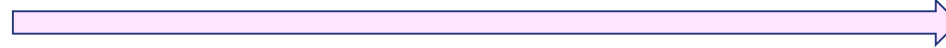
# DayTime Service

---

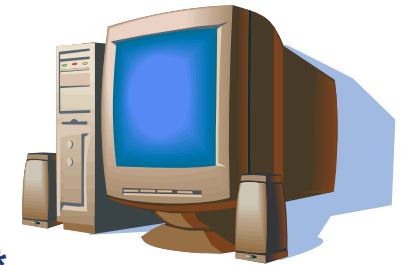
- ❖ DayTime service returns the time.



`connect(time-A.timefreq.bldrdoc.gov, 13)`



`59123 20-10-01 16:56:35 32 0 0 844.2 UTC(NIST) *`



**Client Socket**

- host: client host
- Port: client port

**Server Socket**

- Host: time-A.timefreq.bldrdoc.gov
- Port: 13

```
import java.io.IOException;
import java.net.Socket;
import java.util.Scanner
```

## Client for DayTime Service

```
public class DayTimeClient {
    public static void main(String[] args) {
        try { // Creates a stream socket and connects it to the port on the host.
            Socket socket = new Socket("time-A.timefreq.bldrdoc.gov", 13);
            // Socket(String host, int port) throws UnknownHostException, IOException
            System.out.println(socket.getInetAddress() + " " + socket.getPort());
            System.out.println(socket.getLocalAddress().getHostAddress() + " "
                + socket.getLocalPort());

            try ( Scanner in = new Scanner(socket.getInputStream()) ) {
                while ( in.hasNextLine() ) {
                    String line = in.nextLine();
                    System.out.println(line);
                }
            }
            finally { socket.close(); }
        }
        catch ( IOException e ) { e.printStackTrace(); }
        // UnknownHostException is a subclass of IOException
    }
}
```

```
time-A.timefreq.bldrdoc.gov/132.163.96.1 13
164.125.34.145 1120
```

```
59123 20-10-01 16:56:35 32 0 0 844.2 UTC(NIST) *
```



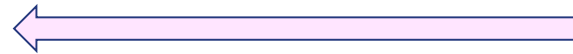
# Echo Service

---

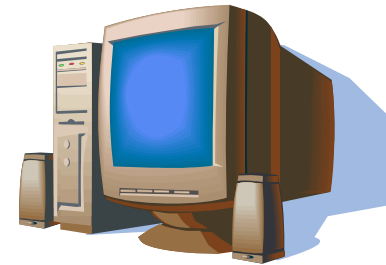


**Echo Client**

Hello, Good Afternoon !



Hello, Good Afternoon !



**Echo Server**

# Implementing Echo Server

```
import java.io.*;
import java.net.*;
import java.util.*;
public class EchoServer {
    public static void main(String[] args ) {
        try {
            ServerSocket s = new ServerSocket(8189);
            // Creates a server socket, bound to the specified port.
            // should avoid the conflict with the well-known ports
            // ServerSocket wait for a connection from a client at 8189

            Socket incoming = s.accept( );
            // Listens for a connection to be made to this socket and accepts it.
            // This method blocks the current thread until the connection is made

            InetAddress clientAddress = incoming.getInetAddress() ;
            System.out.printf("Request from %s[%s] has been accepted!\n",
                clientAddress.getHostName(), clientAddress.getHostAddress()) ;
```



```
try {  
    InputStream inStream = incoming.getInputStream();  
    OutputStream outStream = incoming.getOutputStream();  
  
    Scanner in = new Scanner(inStream);  
    PrintWriter out = new PrintWriter(outStream, true /* autoFlush */);  
  
    out.println( "Hello! Enter BYE to exit." );  
    // echo client input  
    boolean done = false;  
    while ( !done && in.hasNextLine() ) {  
        String line = in.nextLine(); // read a line of text from the client  
        out.println("Echo: " + line); // echo it back to the client  
        if ( line.trim().equals("BYE") ) done = true;  
    }  
}  
finally { incoming.close(); }  
}  
catch (IOException e) { e.printStackTrace(); }  
}
```



# Serving Multiple Clients

```
import java.io.*;
import java.net.*;
import java.util.*;
public class ThreadedEchoServer {
    public static void main(String[] args ) {
        try {
            int i = 1;
            ServerSocket s = new ServerSocket(8189);

            while (true) {
                Socket incoming = s.accept();
                System.out.println("Spawning " + i);
                Runnable r = new ThreadedEchoHandler(incoming, i);
                Thread t = new Thread(r); t.start();
                i++;
            }
        }
        catch (IOException e) { e.printStackTrace(); }
    }
}
```

```
class ThreadedEchoHandler implements Runnable {  
    private Socket incoming;  
    private int counter;  
    public ThreadedEchoHandler(Socket i, int c) { incoming = i; counter = c; }  
    public void run() {  
        try {  
            try {  
                InputStream inStream = incoming.getInputStream();  
                OutputStream outStream = incoming.getOutputStream();  
                Scanner in = new Scanner(inStream);  
                PrintWriter out = new PrintWriter(outStream, true /* autoFlush */);  
                out.println( "Hello! Enter BYE to exit." );  
                boolean done = false;  
                while ( !done && in.hasNextLine() ) {  
                    String line = in.nextLine();  
                    out.println("Echo: " + line);  
                    if ( line.trim().equals("BYE") ) done = true;  
                }  
            }  
            finally { incoming.close(); }  
        }  
        catch ( IOException e ) { e.printStackTrace(); }  
    }  
}
```



# Socket Timeouts

```
Socket(String host, int port)
// can block indefinitely until an initial connection is established

Socket s = new Socket() ;
s.connect(new InetSocketAddress(host, port), timeout) ; // SocketTimeoutException
// Connects this socket to the server with a specified timeout value.
// A timeout of zero is interpreted as an infinite timeout.
```

```
Socket s = new Socket( ... ) ;
s.setSoTimeout(10000) ; // time out after 10 seconds
// all subsequent read and write operations throw SocketTimeoutException
// when the timeout has been reached
try {
    Scanner in = new Scanner(s.getInputStream()) ;
    String line = in.nextLine() ;
    ...
}
catch ( SocketTimeoutException e ) { time out occurred }
```

# Getting Web Data

```
public class URLConnectionTest {
    public static void main(String[] args) {
        try {
            URL url = new URL("https://www.pusan.ac.kr/kor");
            URLConnection connection = url.openConnection();
            connection.connect();
            // print header fields
            Map<String, List<String>> headers = connection.getHeaderFields();
            for (Map.Entry<String, List<String>> entry : headers.entrySet())
                System.out.println(entry.getKey() + ": " + entry.getValue());
            // print first ten liens of the contents
            try (Scanner in = new Scanner(connection.getInputStream())) {
                for (int n = 1; in.hasNextLine() && n <= 10; n++)
                    System.out.println(in.nextLine());
                if (in.hasNextLine()) System.out.println(". . .");
            }
        } catch (IOException e) { e.printStackTrace(); }
    }
}
```

# Getting Web Data

```
Transfer-Encoding: [chunked]
Keep-Alive: [timeout=10, max=100]
null: [HTTP/1.1 200 OK]
Connection: [Keep-Alive]
Pragma: [no-cache]
Date: [Thu, 01 Oct 2020 12:53:27 GMT]
X-Frame-Options: [ALLOW-FROM https://dv.pusan.ac.kr/]
Cache-Control: [no-store, no-cache, must-revalidate, post-check=0, pre-check=0]
Set-Cookie: [JSESSIONID=0000HbvjyMHxOXgSs2wyJP22dXS:-1; Path=/; HttpOnly]
Expires: [Sat, 31 Aug 2019 13:12:12 KST]
Content-Language: [ko]
Content-Type: [text/html; charset=UTF-8]
X-Powered-By: [Servlet/3.0]
```

```
<!doctype html>
<html lang="ko">
```

```
<head>
```

```
<!-- don touch!! -->
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1;">
...
```



# Q&A

---