1 TCP

1.1 Socket (java.net)
• Konstruktoren:
public Socket (String host , **int** port) **throws** UnknownHostException, IOException
public Socket (InetAddress host , **int** port) **throws** IOException

• Methoden:
public InputStream getInputStream() **throws** IOException
public OutputStream getOutputStream() **throws** IOException
public synchronized void close() **throws** IOException
public InetAddress getInetAddress()
public int getPort()
public int getLocalPort()

1.2 ServerSocket (java.net)
• Konstruktor:
public ServerSocket(**int** port) **throws** IOException, BindException

• Methoden:
public Socket accept() **throws** IOException
public void close() **throws** IOException
public InetAddress getInetAddress()
public int getLocalPort()

2 UDP

2.1 DatagramSocket (java.net)
• Konstruktoren:
public DatagramSocket () **throws** SocketException
public DatagramSocket (int port) **throws** SocketException

• Methoden:
public void close()
public void send(DatagramPacket dp) **throws** IOException
public void receive (DatagramPacket dp) **throws** IOException

2.2 DatagramPacket (java.net)
• Konstruktor:
public DatagramPacket(**byte[]** data, **int** length)
public DatagramPacket(**byte[]** data, **int** length, InetAddress host, int port)

• Methoden:
public InetAddress getAddress ()
public int getPort ()
public int getLength ()
public byte[] getData ()
public void setLength(**int** length)

• Konvertierung des Byte-Array in einen String
String s = **new** String(**byte[]** b, **int** from, **int** len)

3 Ein- und Ausgabe

3.1 Byte-Streams ausgeben / einlesen
3.1.1 BufferedOutputStream (java.io)
• Konstruktoren:
public BufferedOutputStream(OutputStream out)

• Methoden:
public synchronized void write (**int** b) **throws** IOException
public synchronized void write (**byte[]** data, **int** offset, **int** length) **throws** IOException
public synchronized void flush( ) **throws** IOException

3.1.2 BufferedInputStream (java.io)
• Konstruktoren:
public BufferedInputStream(InputStream in)

• Methoden:
public synchronized int read() **throws** IOException
// (-1 = Dateiende)

public synchronized int read (**byte[]** input, **int** offset, **int** length) **throws** IOException
// (-1 = Dateiende)

3.2 Aus- und Eingabe
3.2.1 PrintWriter (java.io)
• Konstruktor:
public PrintWriter(OutputStream out)

• Methoden:
public void flush()
public void close()
public void print(String s)
public void println(String s)
public void write(**char[]** buf)
public void write(**int** c)

3.2.2 BufferedReader (java.io)
• Konstruktor:
public BufferedReader(Reader in)

• Methode:
public String readLine() **throws** IOExceptionPrintWriter
// (null = Stream-Ende)

3.2.3 InputStreamReader (java.io)
• Konstruktor:
public InputStreamReader(InputStream in)

3.3 Dateiverarbeitung
3.3.1 Bytes schreiben / einlesen
3.3.1.1 FileOutputStream (java.io)
• Konstruktor:
public FileOutputStream(String name) **throws** FileNotFoundException

• Methoden:
public void write(int b) **throws** IOException
public void write(byte[] b) **throws** IOException
public void flush() **throws** IOException
public void close() **throws** IOException

3.3.1.2 FileInputStream (java.io)
• Konstruktor
public FileInputStream(String name) **throws** FileNotFoundException

• Methoden:
public int read() **throws** IOException
public void close() **throws** IOException

3.3.2 Strings schreiben / einlesen
3.3.2.1 FileWriter (java.io)
• Konstruktoren:
public FileWriter(File f) **throws** IOException
public FileWriter(String fname) **throws** IOException

• Methoden:
public void write(String str) **throws** IOException
public void flush() **throws** IOException
public void close() **throws** IOException

3.3.2.2 FileReader (java.io)
• Konstruktor:
public FileReader(String fileName) **throws** FileNotFoundException

• Methoden:
public void close() **throws** IOException
public int read() **throws** IOException

4 Weitere Klassen

4.1 InetAddress (java.net)
public static InetAddress getByName(String host) **throws** UnknownHostException

public static InetAddress getLocalHost() **throws** UnknownHostException

4.2 URL (java.net)
• Konstruktor: **public** URL(String spec) **throws** MalformedURLException

• Methoden:
public String getHost()
public String getFile()

4.3 String
• Konstruktor:
public String(**byte[]** bytes, **int** offset, **int** length)

• Methoden:
public byte[] getBytes()
public boolean startsWith(String prefix)
public String[] split(String regex, **int** limit)
public int indexOf(String str)
public int indexOf(String str, **int** fromIndex)
public String substring(**int** beginIndex)
public String substring(**int** beginIndex, **int** endIndex)
// The resulting substring starts at beginIndex and
// extends to the character at (endIndex – 1)

4.4 Semaphore
• Konstruktor:
public Semaphore(int permits, boolean fair)

• Methoden:
public void acquire() **throws** InterruptedException
public void release()