**EX 1 STUDY OF NECESSARY HEADER FILES WITH**

**RESPECT TO SOCKET PROGRAMMING**

**Package java.net Description**

Provides the classes for implementing networking applications.

The java.net package can be roughly divided in two sections:

* A Low Level API, which deals with the following abstractions:
  + Addresses, which are networking identifiers, like IP addresses.
  + Sockets, which are basic bidirectional data communication mechanisms.
  + Interfaces, which describe network interfaces.
* A High Level API, which deals with the following abstractions:
  + URIs, which represent Universal Resource Identifiers.
  + URLs, which represent Universal Resource Locators.
  + Connections, which represents connections to the resource pointed to by URLs.

**Sockets**

Sockets are means to establish a communication link between machines over the network. The java.net package provides 4 kinds of Sockets:

* [Socket](https://docs.oracle.com/javase/7/docs/api/java/net/Socket.html) is a TCP client API, and will typically be used to [connect](https://docs.oracle.com/javase/7/docs/api/java/net/Socket.html#connect(java.net.SocketAddress)) to a remote host.
* [ServerSocket](https://docs.oracle.com/javase/7/docs/api/java/net/ServerSocket.html) is a TCP server API, and will typically [accept](https://docs.oracle.com/javase/7/docs/api/java/net/ServerSocket.html#accept()) connections from client sockets.
* [DatagramSocket](https://docs.oracle.com/javase/7/docs/api/java/net/DatagramSocket.html) is a UDP endpoint API and is used to [send](https://docs.oracle.com/javase/7/docs/api/java/net/DatagramSocket.html#send(java.net.DatagramPacket)) and [receive](https://docs.oracle.com/javase/7/docs/api/java/net/DatagramSocket.html#receive(java.net.DatagramPacket)) [datagram packets](https://docs.oracle.com/javase/7/docs/api/java/net/DatagramPacket.html).
* [MulticastSocket](https://docs.oracle.com/javase/7/docs/api/java/net/MulticastSocket.html) is a subclass of DatagramSocket used when dealing with multicast groups.

Sending and receiving with TCP sockets is done through InputStreams and OutputStreams from the [Socket.getInputStream()](https://docs.oracle.com/javase/7/docs/api/java/net/Socket.html#getInputStream()) and [Socket.getOutputStream()](https://docs.oracle.com/javase/7/docs/api/java/net/Socket.html#getOutputStream()) methods. These methods are available in the **java.io** package.

|  |  |
| --- | --- |
| **CLASS** | **DESCRIPTION** |
| [Authenticator](https://docs.oracle.com/javase/7/docs/api/java/net/Authenticator.html) | The class Authenticator represents an object that knows how to obtain  authentication for a network connection. |
| [CacheRequest](https://docs.oracle.com/javase/7/docs/api/java/net/CacheRequest.html) | Represents channels for storing resources in the ResponseCache. |
| [CacheResponse](https://docs.oracle.com/javase/7/docs/api/java/net/CacheResponse.html) | Represent channels for retrieving resources from the ResponseCache. |
| [ContentHandler](https://docs.oracle.com/javase/7/docs/api/java/net/ContentHandler.html) | The abstract class ContentHandler is the superclass of all classes that  read an Object from a URLConnection. |
| [CookieHandler](https://docs.oracle.com/javase/7/docs/api/java/net/CookieHandler.html) | A CookieHandler object provides a callback mechanism to hook up  a HTTP state management policy implementation into the HTTP  protocol handler. |
| [CookieManager](https://docs.oracle.com/javase/7/docs/api/java/net/CookieManager.html) | CookieManager provides a concrete implementation of [CookieHandler](https://docs.oracle.com/javase/7/docs/api/java/net/CookieHandler.html" \o "class in java.net),  which separates the storage of cookies from the policy surrounding  accepting and rejecting cookies. |
| [DatagramPacket](https://docs.oracle.com/javase/7/docs/api/java/net/DatagramPacket.html) | This class represents a datagram packet. |
| [DatagramSocket](https://docs.oracle.com/javase/7/docs/api/java/net/DatagramSocket.html) | This class represents a socket for sending and receiving datagram packets. |
| [DatagramSocketImpl](https://docs.oracle.com/javase/7/docs/api/java/net/DatagramSocketImpl.html) | Abstract datagram and multicast socket implementation base class. |
| [HttpCookie](https://docs.oracle.com/javase/7/docs/api/java/net/HttpCookie.html) | An HttpCookie object represents an http cookie, which carries state  information between server and user agent. |
| [HttpURLConnection](https://docs.oracle.com/javase/7/docs/api/java/net/HttpURLConnection.html) | A URLConnection with support for HTTP-specific features. |
| [IDN](https://docs.oracle.com/javase/7/docs/api/java/net/IDN.html) | Provides methods to convert internationalized domain names (IDNs) between a normal Unicode representation and an ASCII Compatible Encoding (ACE) representation. |
| [Inet4Address](https://docs.oracle.com/javase/7/docs/api/java/net/Inet4Address.html) | This class represents an Internet Protocol version 4 (IPv4) address. |
| [Inet6Address](https://docs.oracle.com/javase/7/docs/api/java/net/Inet6Address.html) | This class represents an Internet Protocol version 6 (IPv6) address. |
| [InetAddress](https://docs.oracle.com/javase/7/docs/api/java/net/InetAddress.html) | This class represents an Internet Protocol (IP) address. |
| [InetSocketAddress](https://docs.oracle.com/javase/7/docs/api/java/net/InetSocketAddress.html) | This class implements an IP Socket Address (IP address + port number)  It can also be a pair (hostname + port number), in which case an attempt will be  made to resolve the hostname. |
| [InterfaceAddress](https://docs.oracle.com/javase/7/docs/api/java/net/InterfaceAddress.html) | This class represents a Network Interface address. |
| [JarURLConnection](https://docs.oracle.com/javase/7/docs/api/java/net/JarURLConnection.html) | A URL Connection to a Java ARchive (JAR) file or an entry in a JAR file. |
| [MulticastSocket](https://docs.oracle.com/javase/7/docs/api/java/net/MulticastSocket.html) | The multicast datagram socket class is useful for sending and receiving IP multicast packets. |
| [NetPermission](https://docs.oracle.com/javase/7/docs/api/java/net/NetPermission.html) | This class is for various network permissions. |
| [NetworkInterface](https://docs.oracle.com/javase/7/docs/api/java/net/NetworkInterface.html) | This class represents a Network Interface made up of a name, and a list of IP addresses assigned to this interface. |
| [PasswordAuthentication](https://docs.oracle.com/javase/7/docs/api/java/net/PasswordAuthentication.html) | The class PasswordAuthentication is a data holder that is used by Authenticator. |
| [Proxy](https://docs.oracle.com/javase/7/docs/api/java/net/Proxy.html) | This class represents a proxy setting, typically a type (http, socks) and a socket address. |
| [ProxySelector](https://docs.oracle.com/javase/7/docs/api/java/net/ProxySelector.html) | Selects the proxy server to use, if any, when connecting to the network resource referenced by a URL. |
| [ResponseCache](https://docs.oracle.com/javase/7/docs/api/java/net/ResponseCache.html) | Represents implementations of URLConnection caches. |
| [SecureCacheResponse](https://docs.oracle.com/javase/7/docs/api/java/net/SecureCacheResponse.html) | Represents a cache response originally retrieved through secure means, such as TLS. |
| [ServerSocket](https://docs.oracle.com/javase/7/docs/api/java/net/ServerSocket.html) | This class implements server sockets. |
| [Socket](https://docs.oracle.com/javase/7/docs/api/java/net/Socket.html) | This class implements client sockets (also called just "sockets"). |
| [SocketAddress](https://docs.oracle.com/javase/7/docs/api/java/net/SocketAddress.html) | This class represents a Socket Address with no protocol attachment. |
| [SocketImpl](https://docs.oracle.com/javase/7/docs/api/java/net/SocketImpl.html) | The abstract class SocketImpl is a common superclass of all classes that actually implement sockets. |
| [SocketPermission](https://docs.oracle.com/javase/7/docs/api/java/net/SocketPermission.html) | This class represents access to a network via sockets. |
| [StandardSocketOptions](https://docs.oracle.com/javase/7/docs/api/java/net/StandardSocketOptions.html) | Defines the standard socket options. |
| [URI](https://docs.oracle.com/javase/7/docs/api/java/net/URI.html) | Represents a Uniform Resource Identifier (URI) reference. |
| [URL](https://docs.oracle.com/javase/7/docs/api/java/net/URL.html) | Class URL represents a Uniform Resource Locator, a pointer to a "resource" on the World Wide Web. |
| [URLClassLoader](https://docs.oracle.com/javase/7/docs/api/java/net/URLClassLoader.html) | This class loader is used to load classes and resources from a search path of URLs referring to both JAR files and directories. |
| [URLConnection](https://docs.oracle.com/javase/7/docs/api/java/net/URLConnection.html) | The abstract class URLConnection is the superclass of all classes that represent a communications link between the application and a URL. |
| [URLDecoder](https://docs.oracle.com/javase/7/docs/api/java/net/URLDecoder.html) | Utility class for HTML form decoding. |
| [URLEncoder](https://docs.oracle.com/javase/7/docs/api/java/net/URLEncoder.html) | Utility class for HTML form encoding. |
| [URLStreamHandler](https://docs.oracle.com/javase/7/docs/api/java/net/URLStreamHandler.html) | The abstract class URLStreamHandler is the common superclass for all  stream protocol handlers. |