**package** xhu.edu.socket2;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.IOException;

**import** java.io.InputStream;

**import** java.io.InputStreamReader;

**import** java.io.OutputStream;

**import** java.io.OutputStreamWriter;

**import** java.net.Socket;

**import** java.net.UnknownHostException;

/\*

\* 需求：模拟浏览器向服务器发送一个请求，并接收信息然后显示。

\* 分析：通信双方正常通信的前提，是遵循相同正确的协议，所以无论是请求还是响应都要遵循一定的格式规范。

\* 如HTTP协议：

\* 请求方：

\* 请求行：请求的方式 请求的资源路径 请求的协议以及版本号

\* 请求头

\* 空行

\* 请求体

\*

\* 响应方：

\* 响应行：响应的协议以及版本号 请求的资源的状态码 响应的状态

\* 响应头

\* 空行

\* 响应体

\*

\* 所以请求步骤：

\* 1.建立一个套接字

\* 2.开启输入输出流

\* 3.向服务端发送请求信息

\* 4.接收并显示信息

\* 5.关闭资源

\*/

**public** **class** Client {

**public** **static** **void** main(String[] args) **throws** UnknownHostException, IOException {

Socket socket=**new** Socket("www.baidu.com",80);

//方式1：使用基本流

// InputStream inputStream=socket.getInputStream();

// OutputStream outputStream=socket.getOutputStream();

// outputStream.write("GET / HTTP/1.1\n".getBytes());

// outputStream.write("Host: m.baidu.com\n".getBytes());

// outputStream.write("Connection: Close\n".getBytes());

// outputStream.write("\n".getBytes());

// byte[]b=new byte[1024];

//使用：缓冲流

BufferedReader br = **new** BufferedReader(**new** InputStreamReader(socket.getInputStream()));

BufferedWriter bw = **new** BufferedWriter(**new** OutputStreamWriter(socket.getOutputStream()));

bw.write("GET / HTTP/1.1\n\n");

bw.flush();

**while**(**true**) {

String string=br.readLine();

**if**(string==**null**) {

**break**;

}

System.***out***.println(string);

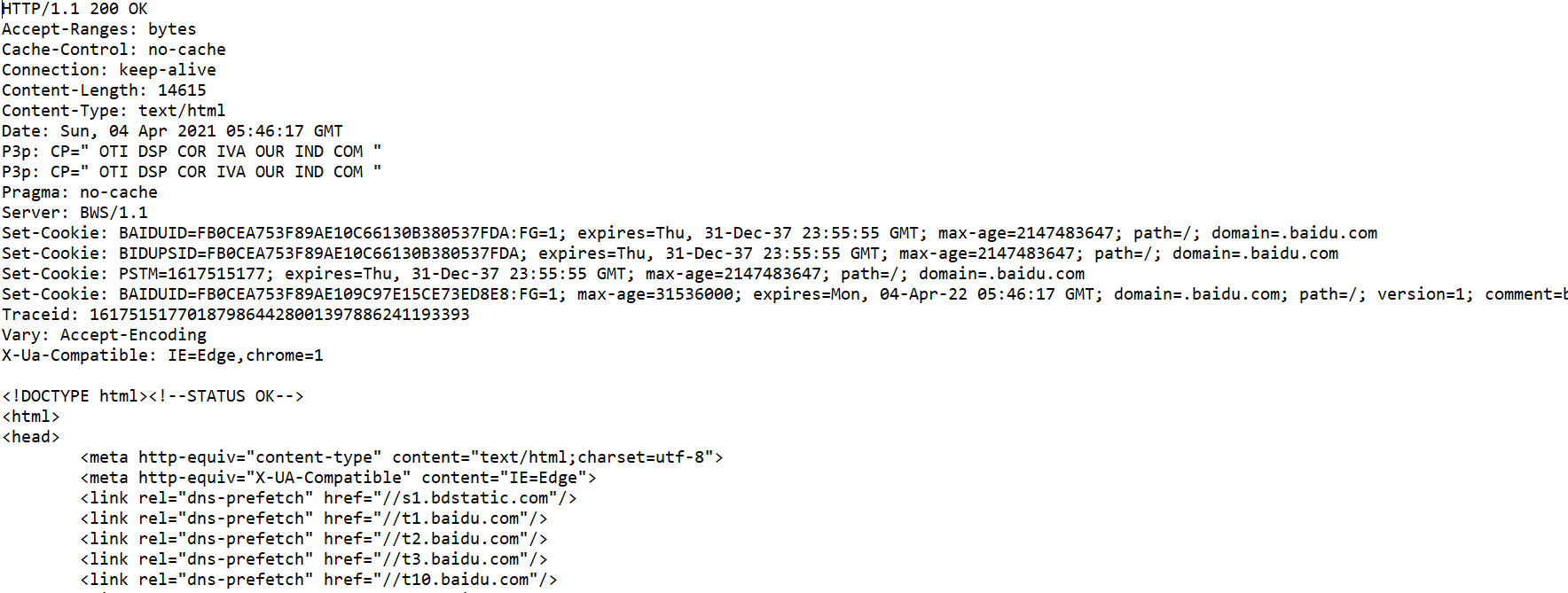
// System.out.println(new String(b));

}

}

}

运行截图：



总结：

通过输出流发出请求，请求的格式一定要对，大小写需要区分，注意：http1.0是短链接，1.1是长连接。缓冲流一定要进行flush.请求的资源路径，当有中文时需要进行URL编码。