

Network :

- connected computing devices.

- host = end system

ex: mobile, server, workstation (Desktop), routers, switches.

OSI Model: 7 layers (طبقات)

Application layer

Presentation layer

Session layer

Transport layer

Network layer

Data link layer

Physical layer

OSI

→ Data

Http,Https,URL,URI,DNS,FTP

sokets TCP/UDP (Segment) // sokets

IP/TPsrc (Router) (Packets)

Mac(Switch), Ethernet (frames) PPP

Cable, fiber, coaxial (bit)

نحوه من جهازات الاتصال تكون
في طبقات بینهم

Application

Transport

Network

Data link

Physical

TCP/IP Model

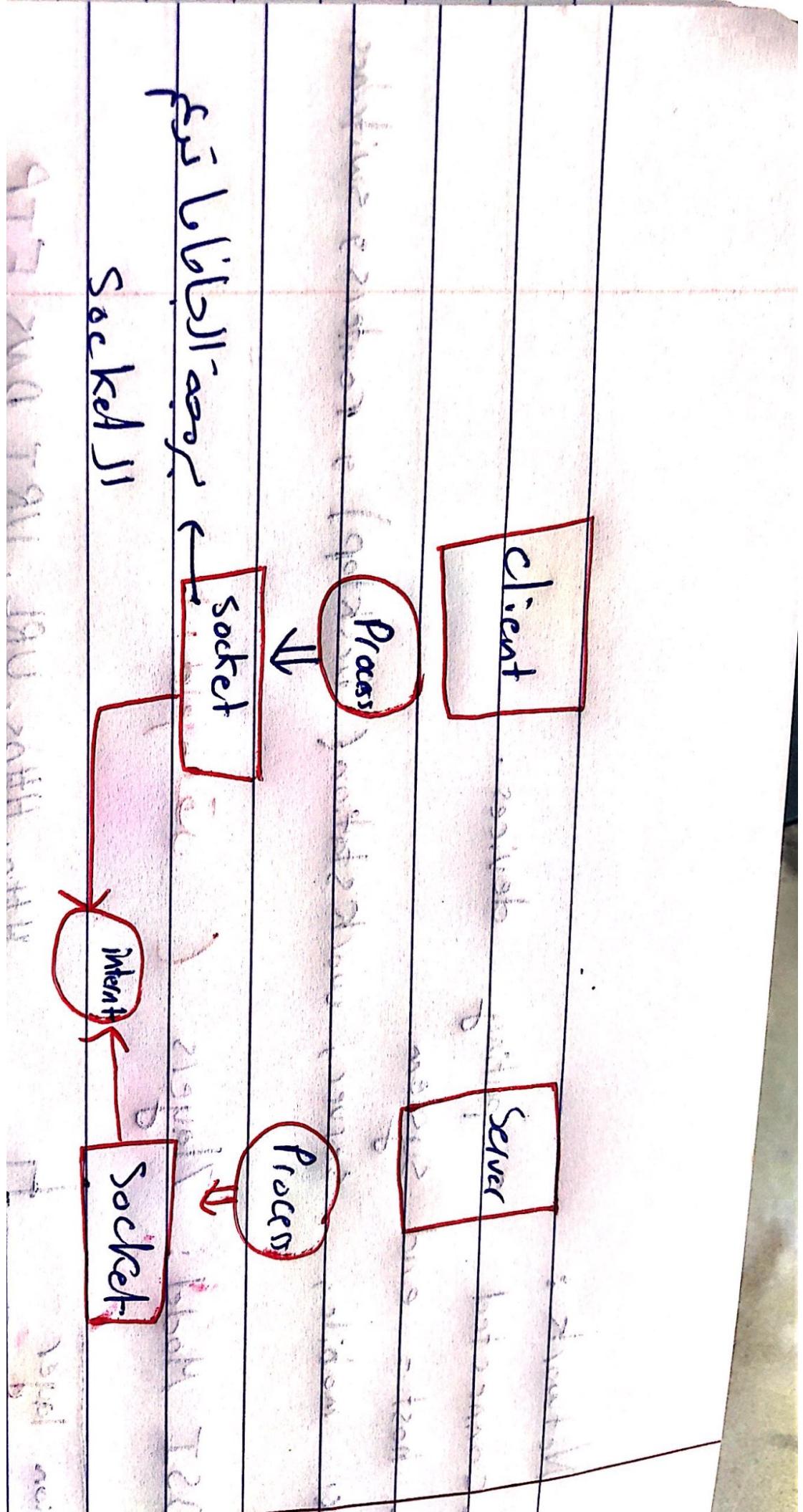
5 layers:

client - server \Rightarrow

مدعوم في Java API

Peer - Peer \Rightarrow

غير مدعوم في Java API

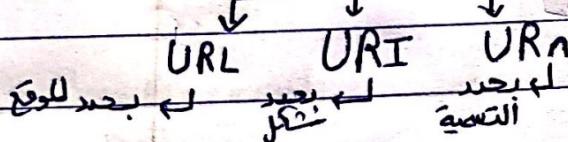


| Client | Server |
|------------|-----------|
| on/of | on |
| dynamic IP | static IP |

Soket || برمجة Java *

- Soket وظائف
- 1. choose protocol of transport
 - 2. fix parameters.

web / HTTP



URL: Uniform resource locator

URI: Uniform resource identifier

URN: Uniform resource name

الشكل الذي لا يعرّف الشكل (غير المعرف)

الشكل الذي لا يعرّف الموقع

الشكل الذي يعرّف الموقع

URI

URN

URL

http : \ thinkzara . com / auther / C . html . page ,

URI

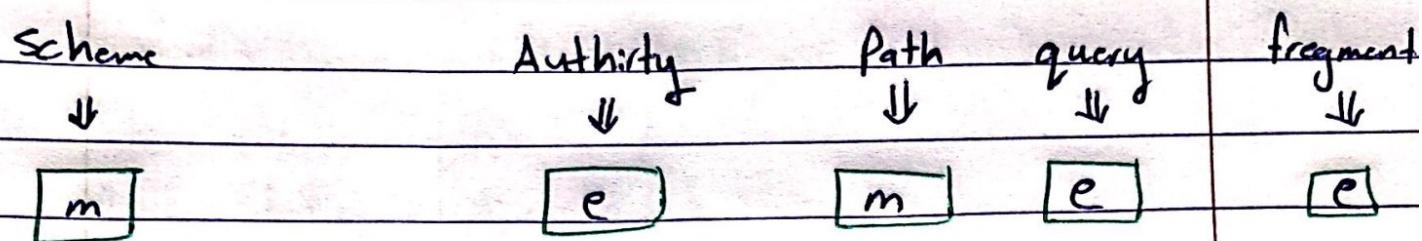
URI كل لوقع *

URL ← HTML

http كل لوقع *

URN ← http كل ل الواقع بلا *

http : // www.facebook.com \ c?Emp=4 & col=6



الشكل التفاصيلي
للموقع

مُولف للوقت

URI \Rightarrow Scheme mandatory m
Authority elective e

URI & URL: مكتبة بالجافا، لم يتم إدخال دينم

URn : Don't have library

URL :

Protocol domainname Path query fragment

Java . net . URI

Java . net . URL

برنامجه لتنفيذ الـ URI يعمل على تفسيم مكوناته وتحديد مكوناته

الـ 5

```
import java.net.URI;  
public class URITest {  
    public static void main() {  
        throw exception();  
    }
```

URI uri = new URI("http://www.facebook.com/c?EmpData=14
*&column=64");

s.o.pln(uri.getScheme());

s.o.p(uri.getAuthority());

s.o.p(uri.getPath());

s.o.p(uri.getQuery());

s.o.p(uri.getFragment());

}

}

⇒ (uri.getPort());

الحصول على الـ Port

⇒ (uri.getUserInfo());

الحصول على اسم المستخدم والـ Password

الحصول على الموقع

① network eng

المعلومات تغير ←

② web Dev

③ ethical hacker

① mining → E-commerce

البيانات تغير ←

② management → load server

memory

* Convert URI to URL

```
import java.net.URL;  
import java.net.URI;
```

```
public class URItoURL {
```

```
    public static void main()
```

```
        throws Exception {
```

```
            URI uri = new URI("http://www.you.com");
```

```
            URL url = uri.toURL();
```

```
            System.out.println("original" + uri);
```

```
            System.out.println("converted" + url);
```

```
}
```

```
}
```

* Convert file to URI

```
import java.net.URI;
```

```
import java.net.IO.File;
```

```
public class FiletoURI {
```

```
    public static void main()
```

```
        throws Exception {
```

```
            File f = new File("test.txt");
```

```
            URI uri = f.toURI();
```

```
            System.out.println("original file" + f);
```

```
            System.out.println("converted" + uri);
```

```
}
```

```
}
```

HTTP response message ? What's it

HTTP/1.1 200 ok

↓
http version status
↓
الموقع صالح للعمل

الموقع صالح للفل

connection open

Connection open

server

Last Modified

Content length

Content type

Data: html file

header lines

HTTP response message

1) Connection : open or close

2) status : Inc / key field

~~HTTP response message~~

برنامجه يعزم فيه ملحوظات الواقع
ويحصل منه على كل ملحوظات الواقع

```
import java.net.URL;
import java.net.URLConnection;
import java.net.HttpURLConnection;
import java.sql.Date;
import java.io.IOException;
```

```
public class URLInfo {
```

```
    public static void main()
```

```
        throws IOException {
```

```
        URL url = new URL("http://opensource.com");
```

```
        URLConnection c = url.openConnection();
```

```
        c.connect();
```

```
        System.out.println(c.getContentType()); // محتوى الملف
```

```
        System.out.println(c.getContentEncoding()); // صيغة المحتوى
```

```
        System.out.println(new Date(c.getDate())); // ميلاد التاريخ
```

```
        System.out.println(new Date(c.getLastModified())); // ميلاد آخر تغيير للمحتوى
```

```
        System.out.println(new Date(c.getExpiration())); // ميلاد تاريخ انتهاء صلاحية المحتوى
```

* Date → have its own library.

```
        if (c instanceof HttpURLConnection) {
```

```
            }
```

```
        else {
```

```
            HttpURLConnection h = (HttpURLConnection) c;
```

```
}
```

is instance of
URLConnection

S.o.p("Request Method" + h.getRequestMethod());
 [* Get , Post , Delete , Put] Method of HTTP.

S.o.p("Response Message" + h.getResponseMessage());
 [* OK, moved Permanently , Bad Request]

OK: succeeded - 200

Moved Permanently : moved to another location 301

Bad Request : not understand request by server 400

Not Found : not found on server 404

HTTP Not supported : 500

* 500, 404, 400, 301, 200

S.o.p("Response Code" + h.getResponseCode());

* Key Header { Header } الامر السابق يطبع

import java.net.URL;

import java.net.URLConnection;

import java.io.IOException

public class URLHeader {

public static void main() throws IOException {

URL u = new URL("http://www.wafa.com");

URLConnection uc = u.openConnection();

for (int i=0; i < 5; i++):

String header = uc.getHeaderField();

if (header == null)

break;

200 OK

S.o.p(uc.getHeaderField(i) + header);

3

طريقتان لتجزيع المحتويات URL *

* HTML
* CSS
* Links } Content محتوى

1) Buffered readers inherit from Reader

ReadLine() بخط 16 bit
تخزين ثم قراءة

2) Buffered Input Stream inherit from InputStream

Read() حرف حرفاً 8 bit تدفق التخزين ثم قراءة

الامر لقراءة البيانات من محتويات الموقعة [2]

```
import java.net.URL;
import java.net.URLConnection;
import java.io.IOException;
import java.io.InputStream;
import java.io.BufferedReader;

public class URL_BufferedInputStream {
    public static void main() throws IOException {
        URL u = new URL("https://ptuk.edu.ps");
        URLConnection uc = u.openConnection();
        InputStream is = uc.getInputStream();
        BufferedInputStream buf = new BufferedInputStream(is);
        int i = 0;
        Char c في ASCII
        while ((i = buf.read()) != -1) {
            c = (Char)i;
            System.out.print(c);
        }
    }
}
```

```

import java.io.BufferedReader;
import java.io.InputStreamReader; + 5

public class URLBufferedReader {
    public static void main() throws {
        URL u = new URL ("http://google.com");
        URLConnection uc = u.openConnection();
        InputStream is = uc.getInputStream();
        InputStreamReader isr = new InputStreamReader(is);
        BufferedReader buf = new BufferedReader(isr);
        String s = buf.readLine();
        while (s != null) {
            System.out.println(s);
            s = buf.readLine();
        }
    }
}

```

???

HTTPs: Secure connection between Client & Server

SSL: Secure Socket Layer

Digital certificate to give authenticate website identity & enables an encrypted connection between web server & web browser

→ **TSL:** Transport Security Layer

HTTPs = HTTP + TSL

[Publickey, Hashcodes, Algorithm cipher] \rightarrow SSL Content

information Gathering

البيانات جمع

```
import Java.net.URL  
import Java.. URLConnection  
import Java.io.IOException  
import Java.io.InputStream  
import Java.io.BufferedReader  
import Java.io.InputStreamReader  
import Java.security.cert.Certificate  
import JavaX.net.SSL.HttpsURLConnection
```

Public Class HTTP ClientServer {

```
Public static void main() throw {
```

```
URL u = new URL ("http://ptuk.edu.ps");
```

```
URLConnection uc = u.openConnection();
```

```
HTTPsURLConnection con = (HTTPsURLConnection) uc;
```

```
InputStream is = con.getInputStream();
```

```
InputStreamReader isr = new InputStreamReader(is);
```

```
BufferedReader buf = new BufferedReader(isr);
```

```
String s = buf.readLine();
```

```
while (s != null) {
```

```
s.o.println(s);
```

```
s = buf.readLine();
```

```
}
```

s.o.println("cipherCode" + con.getcipherSuite())

ادخل طريقة التشفير المطلوب *

ما هي تخرج هذه الـ SSL

⇒ ① طريقة لاكتشاف داخل الـ class

Java.Security.Cert. Certificate[] cert;
library \rightarrow class \rightarrow array
ما هي تخرج هذه الـ SSL

Certs con.getServerCertificate()

الـ SSL يطلب التفاصيل من السيرفر المجهولة بتكره هذه العبرة

ترفق معلومات الموقع

بعد ذلك اطلب العوادة

⇒ ② طريقة المكابحة خارج البرنامج

Certificate[] cert = con.getServerCertificates();

server client ?

```
for (int i=0; i<=2; i++) {  
    S.o.p ln ("CertType" + cert[i].get Type());  
    S.o.p ln ("CertHashCode" + Cert[i].hashCode());  
    S.o.p ln ("CertPublicKey" + Cert[i].GetFormat());  
    S.o.p ln ("CertPublicKeyAlgorithm" + Cert[i].get Algorithm());  
}
```

? ?

لو عيني (Client) وهو (جهاز من المطاعنة)
ويختلف جهاز

URL Encoding & Decoding

UTF-8: Unicode Transformation Format (8 bit)

١) ASCII زل طریقہ استعمال فی التیفڑھ

وكانت بالبداية فتحت تحف اللغة لازجليزية وبعدها الرموز

$(a-z) \& (A-Z) \& (\$! = > <)$
 $(\text{char } 128) \rightarrow \text{null}$

2 بعدها تستخدم طريقة ال ANST

محبوبة كلها عصمت اللغات لكن اخترت لغات عديدة
لأنني لست بالذكاء لكل اللغات

UTF-8 بعدها تختبروا صيغة الـ (3)

من غير نسخ خوف المبرمج [تقريباً كل لغات العالم] والكلود على المعاشر HTML ومتافق مع 5 Hexadecimal في الواقع هو واحد في الما meta في وصف الواقع.

| | | | | | | | | | | |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Space | ! | * | \$ | : | & | * | + | - | / | : |
| 20% | 21% | 23% | 24% | 25% | 26% | 2A% | 2B% | 2D% | 2F% | 3A% |

* فـاـكـتـسـفـر <= decode كـيـفـنـعـل

```
import java.net.URLDecoder  
      .io.IOException  
      .net.URLEncoder
```

```
class Public URLDecoder {
```

```
Public static void main() throws
```

* هـذـاـالـمـدـدـهـ[ـبـلـعـزـلـ]ـحـبـنـاـ20%

* هـذـهـالـمـدـدـهـ[ـبـلـعـزـلـ]ـحـبـنـاـ20%

```
String decoded = method String s = "http://www.google.com/my%20name%20  
mahmoud%21 moshref%23";
```

String decoded = URLDecoder.decode(s, "UTF-8")
فـاـكـتـسـفـر

S.o.P(s);

S.o.P(decoded);

?

* فـاـكـتـسـفـر <= Encode كـيـفـنـعـل

2A% ، %21 يـاءـنـفـسـعـادـيـ بـرـلـ string ①

ـهـذـهـالـمـدـدـهـ[ـبـلـعـزـلـ]ـحـبـنـاـ20%

"com"? name! mohamed! Moshref";

String Encoded = URLEncoder.encode(s, "UTF-8");
المـدـدـهـ[ـبـلـعـزـلـ]ـحـبـنـاـ20% ②

ـهـذـهـالـمـدـدـهـ[ـبـلـعـزـلـ]ـحـبـنـاـ20% ③

* Protocol Testing

* Application layer : HTTP, HTTPS, FTP, telnet
LDAP, JDBC, NFS, file

Supported ↗
not supported ↘
برنامجه فحص Protocols

Two way to check Protocols

- 1) Regular expression
- 2) Use String

URL مكونات

- 1) Protocol
- 2) host
- 3) file

```
import java.net.URL
```

```
public class ProtocolTest
```

```
public static void main() throws
```

```
String host = "www.google.com";
```

```
String file = "index.html";
```

```
String[] regex = {Http, Https, FTP, telnet, mailto  
Ldap, jdbc, nfs, file};
```

```
for (int i=0; i < regex.length; i++) {
```

```
try {
```

```
URL u = new URL(regex[i], host, file)
```

```
s.o.println(regex[i] + " is supported");
```

```
catch (Exception e) {
```

```
s.o.println(regex[i] + " not supported")
```

```
}
```

```
}
```

```
}
```

```
}
```

* Http is supported

use String النوع الثاني *

Public URL Protocol Tester

Public static void main() throws

Tester("http://www.google.com");
Tester("https://www.google.com");
Tester("file:///www-google.com");
Tester("ftp://get.google.com");
Tester("mailto:moshraf2021@gmail.com");
Tester("telnet:google.com");
Tester("ldap:google.com");
Tester("nfs:google.com");

public static void Tester(String url){

try{

URL url = new URL(url);

s.o.p(Cu.getProtocol() + " is supported");

} catch(~~sازن بـ~~ [http:]

String s = url.substring(0, url.indexOf(":"));

(":");

URL url بـ

s.o.p(s + "not supported");

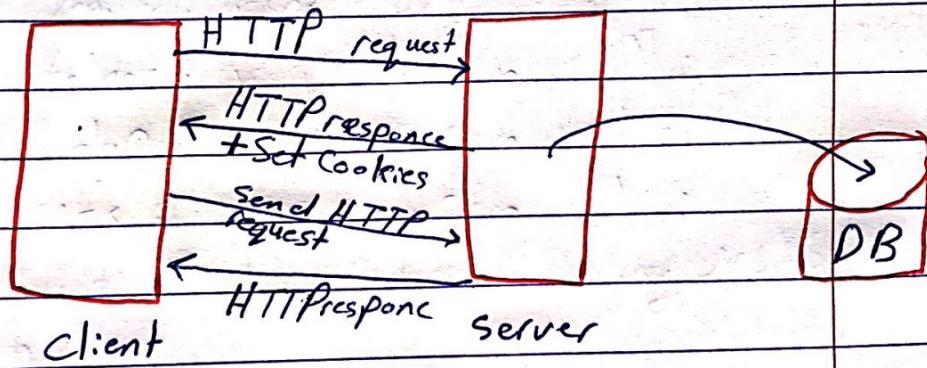
???

Try & catch (i) \leftarrow (1) اكتب قليلًا عن مفهوم try catch

for loop (2) \leftarrow

```
URL url = new URL(regex[i], host, file);
if (u.getFile() != null) {
    soap(regex[i] + "is supported");
}
else {
    soap(regex[i] + "is not supported");
}
```

Cookies \leftarrow الـ Cookies



Cookies : Save some Data about users

why ??

1. Authorization \leftarrow User of server
2. shopping cart \leftarrow state
3. recommendation
4. User Session State

1) Send cookies to HTTP server:

3 library
public class Send URLCookies {
public static void main() throws {

URL u = new URL("https://researchgate.net");

URLConnection con = u.openConnection();

con.setRequestProperty("Cookie", "name=mahmoud" +
"password=123");

con.connect();

?
?

Header key Header value Send with Cookies
massage.

2) Get Cookies from http Server.

(Set-Cookie : cookie-name = value ; expire = "VIII/2023"
Path = "/" ; domain = "example.com" ; Secure)

إذا أتيت بـ Set-Cookie جابك الـ Cookies

أجبك Cookies still تفاصيل Cookies! المهم

Header key : Set Cookies

Header value : name , Path , domain , expire , Secure .

(Set cookies : Cookies = name ;

(header name)

(header-value)

(key)

(value)

expire = 1/3/2022; Path = /c ;

domain = example . com ; Secure)

↓ (إنه القيمة التي بعد هيدر تكون مشفرة)

URLConnection Uc = U . openConnection () ;

لأن هنا الوصول إلى الكائن سكريپت بتجربة نفس العدد

for (int i = 0 ; i < ; i ++)

String headerName = Uc . getHeaderFieldKey (i) ;

↓ (هنا لايتم بحسب الـ key على (المتغير المعرفة))

String headerValue = Uc . getHeaderField (i) ;

↓ (حينما تنتهي مساحة بحسب)

إذا صدرت الشروط فاضنها من كليه يطبع

if (headerName == null) headerValue == null)

{

break ;

}

سؤال السؤال التالي:-

if ("Set-Cookies".equalsIgnoreCase(headerName)),
يعني هنا هاد هو بعده وموهنه بكل حرفها وبينجبيها وينطبع
واعمل حرف كل الحروف كبيرة او صغيرة

s.o.println(headerName);
لـ مباشه هلام اللي جبته

String[] fields = headerValue.split(" ");
بعض لـ معوجه لـ الـ z وخفـ في Index1 نـ اعـنـ السـ طـ خـ زـ يـ ؟
* حـ سـ اـ جـ يـ حـ سـ وـ فـ شـ بـ دـ وـ رـ عـ طـ وـ حـ كـ لـ تـ سـ تـ حـ مـ دـ (ـ كـ اـ فـ)

for (int j=0; j < Fields.length; j++) {
if ("Secure".equalsIgnoreCase(FieldsOf[j]))
بعـ زـ اـ نـ شـ وـ فـ القـ يـ الـ جـ اـ بـ زـ : اـ اـ لـ اـ كـ اـ دـ سـ اـ لـ اـ سـ اـ كـ اـ دـ دـ ماـ بـ عـ
مـ عـ حـ مـ اـ حـ اـ لـ حـ عـ اـ حـ اـ بـ اـ بـ زـ .

s.o.println("secure = true");
}

elseif (Fields[i].indexOf("=") > 0)
لـ هـ يـ رـ تـ عـ لـ عـ لـ وـ رـ اـ = !: الـ قـ يـ تـ هـ يـ رـ جـ يـ
الـ سـ رـ اـ حـ اـ .

s.o.println(headerValue);
}

{ } } } } } }

* اـ نـ اـ بـ جـ يـ بـ الدـ لـ وـ هـ اـ منـ حـ كـ اـ بـ عـ دـ مـ حـ لـ السـ فـ رـ هـ اـ

Throw Exception نـ Try, catch

لو حـ اـ قـ اـ قـ يـ بـ خـ لـ data base ^{لـ} الـ يـ اـ لـ وـ هـ اـ رـ كـ اـ لـ سـ يـ بـ خـ لـ حـ ضـ يـ حـ اـ فـ الـ حـ اـ لـ
JDBC \Rightarrow Java Data Base Connectivity \Rightarrow APT

[لدينا نعمل نفس حار البرنامج وبنفسك محيّي
Cookies نحيي الـ

* Cookies Protection :

إذا الفكر بـ encapsulation نجعل للـ main

Public Class Encapsulate Cookies {

 Private static String headerName;

 Private static String headerValue;

 Public static URLConnection encapsulateCookies (URLConnection Uc)

}

 for (int i = 0; i < 5; i++) {

 // يكتبني البرنامج (السابق كله) لمن for i<5

 return Uc;

}

Public class main Encapsulate {

 Public static void main (String []) {

 URL u = new URL ("---");

 URLConnection Uc = u.openConnection();

 // هنا (الحالات الخمس) نداء Uc البرايمت إلى يستقبله حار (الكلام)

 EncapsulateCookies encapsulateCookies (Uc);

 S.O.P ("unreachable");

3. (Remote Method Invocation) RMI

إذا حصلنا على خارق للحيد المجهز الذي خوفنا