
WEB SISTEMAK

2020/2021 IKASTURTEA

HTTP - HyperText Transfer Protocol

Datu bidalketa bezerotik zerbitzarira: inprimakiak

2021-02-11 M



Web Sistemak by [Oskar Casquero](#) & [María Luz Álvarez](#) is licensed under a [Creative Commons Reconocimiento 4.0 Internacional License](#).

HTTP-REN FUNTZIONAMENDUA: INPRIMAKIAK

- **Zelan bidaltzen da informazioa bezerotik zerbitzarira?**
 - HTTP bezero batek web zerbitzari bati datu mota ezberdinak bidali diezazkioke:
 - datuak formatu binarioan (adibidez: irudiak)
 - datuak JSON edo XML formatuetan
 - **datuak inprimaki formatuan**
- Orain datuen bidalketa inprimaketa formatuan egiten ikasiko dugu, hau da, izen-balio bikote moduan bidaltzen diren datu kate laburrak, alegia.
 - Eredu orokorra: name1=value1&name2=value2&name3=value3
 - Ereduaren adibidea: nan=12345678Z&izena=oskar&abizena=casquero

HTTP-REN FUNTZIONAMENDUA: INPRIMAKIAK -- POST

- Demagun web zerbitzari batean NAN letra kalkulatzen duen baliabide bat dagoela. Baliabide horri *nan* izeneko parametroan NAN zenbakia duen HTTP eskaera bat bidaltzen bazaio, baliabideak NAN letra duen HTTP erantzun bat itzultzen du.
- Datua inprimaki formatuan bidaltzeko, bezeroak HTTP eskaera horrela definitzen du:
 - Metodoa: POST
 - Mezuaren gorputza bereizteko goiburuak:
 - Content-Type: application/x-www-form-urlencoded
 - Content-Length
 - Izen-balio bikoteak era jakin batean kodifikatu (UTF-8 + ehuneko kodifikazioa) eta formateatu (izena=balioa eta "&" bikoteen arteko bereizle bezala) egin behar dira

Eskaera baten adibidea

```
POST /resource HTTP/1.1
Host: ws2017.com
Accept: text/plain
Content-Type: application/x-www-form-urlencoded
Content-Length: 12
User-Agent: Python bezeroa

dni=12345678
```

Erantzun baten adibidea

```
HTTP/1.1 200 OK
Date: Wed, 25 Nov 2015 08:07:43 GMT
Content-Length: 1
Content-Type: text/plain

Z
```

HTTP-REN FUNTZIONAMENDUA: INPRIMAKIAK -- GET

- Izen-Balio bikoteak mezuaren gorputzean bidali beharrean, URI-an bertan bidali daitezke “?” karakterea juntagailu bezela erabiliz → *query string*
 - Metodoa: GET
 - Ez dira *Content-Type* eta *Content-Length* goiburuak erabili behar
 - Query String-ean inprimaki formatudun datuak bakarrik bidali daitezke.
 - HTTP eskeraren egiturak query string-a mugatzen du: izen-balio bikoteen katea “?” karakterearen eta “ HTTP/1.1” katearen tartean kokatzen da.
 - Datuen kodifikazioa eta formatua lehengo moduan egiten da.
 - Desabantaila: bidali daitekeen datu koputua mugaturik dago: 1024 zortzikote.

Eskaera baten adibidea

```
GET /recurso?dni=12345678 HTTP/1.1
Host: ws2017.com
Accept: text/plain
User-Agent: Python bezeroa
```

Erantzun baten adibidea

```
HTTP/1.1 200 OK
Date: Wed, 25 Nov 2015 08:07:43 GMT
Content-Length: 1
Content-Type: text/plain
```

Z

HTTP-REN FUNTZIONAMENDUA: LABURPENA

GET /app/servlet/contactUs?name=XXX&email=YYY&message=ZZZ HTTP/1.1

Host: ws2017.com

Accept: text/html

Accept-Language: en-US,en

Connection: keep-alive

POST /app/servlet/contactUs HTTP/1.1

Host: ws2017.com

Accept: text/html

Accept-Language: en-US,en

Connection: keep-alive

Content-Type: application/x-www-form-urlencoded

Content-Length: 30

name=XXX&email=YYY&message=ZZZ

HTTP-REN FUNTZIONAMENDUA:

DATUEN KODIFIKAZIOA INPRIMAKI FORMATUAN

- Inprimaki formatuan, edukiarentzako ondorengo kodifikazio prozedura zehazten da izen-balio bikotearen izen eta balio kateentzat:
 - Karaktere berezi bat (URI-an edo izen-balio bikote katean bereizle diren karaktereak) edo [US-ASCII](#) alfabetoan agertzen ez den karaktere bat kodifikatu nahi bada (adibidez, azentua duten karaktereak), izen-balio bikote katea osatu aurretik bi pausu burutu behar dira:
 1. Karakterea [UTF-8](#)-n kodifikatu.
 - Adibidea 1: & → 26
 - Adibidea 2: á → C3 A1
 2. Aurreko pausuaren emaitza “ehuneko” kodifikazioan (percent encoding)
 - Adibidea 1: 26 → %26
 - Adibidea 2: C3 A1 → %C3 %A1

OHARRA: goiko adibide bakoitzean, karaktere bakarra kodifikatzen da; baina kodifikazio prozeduraren eraginez, zenbat karaktere bidaliko dira?

- Adibidea 1: %26 → 3 karaktere: %, 2, 6
- Adibidea 2: %C3 %A1 → 6 karaktere: %, C, 3, %, A, 1

HTTP-REN FUNTZIONAMENDUA:

DATUEN KODIFIKAZIOA INPRIMAKI FORMATUAN

- Adibidea:
 - bidali nahi diren datuak: izena=Iñaki y abizena=Pérez
 - mezuaren gorputzean kodifikatuko diren datuak:
izena=I%C3%B1aki&abizena=P%C3%A9rez
- izen-balio bikotearen izen eta balio kateetan zuriune bat agertzen bada, hau “+” batekin kodifikatzen da:
 - bidali nahi diren datuak: izena_abizena=Iñaki Pérez
 - mezuaren gorputzean kodifikatuko diren datuak:
izena_abizena=I%C3%B1aki+P%C3%A9rez
- “+” izen-balio bikotearen izen eta balio kateetan datu bezela erabili nahi bada, orduan UTF-8 + % kodifikazio prozedura jarraitu behar da:
 - bidali nahi diren datuak: num1=2, num2=2 y operator=+
 - mezuaren gorputzean kodifikatuko diren datuak:
num1=2&num2=2&operator=%2B

ADIBIDEA:

DATU BIDALKETA INPRIMAKI FORMATUAN -- POST

- NAN letraren zenbakia lortu: <http://ws-sendingformdata.appspot.com/>

Bezeroaren eskaera

POST /processForm HTTP/1.1
Host: ws-sendingformdata.appspot.com
Content-Type: application/x-www-form-urlencoded
Content-Length: 12

nan=12345678

Zerbitzariaren erantzuna

HTTP/1.1 200 OK
Date: Wed, 25 Nov 2015 08:07:43 GMT
Server: Google Frontend
Content-type: text/html; charset=utf-8
Content-Length: 1

Z

```
import requests  
import sys
```

```
metodoa = 'POST'  
uria = "http://ws-sendingformdata.appspot.com/processForm"  
goiburuak = {'Host': 'ws-sendingformdata.appspot.com',  
             'Content-Type': 'application/x-www-form-urlencoded'}  
edukia = {'nan': sys.argv[1]}  
edukia_encoded = urllib.parse.urlencode(edukia)  
goiburuak['Content-Length'] = str(len(edukia_encoded))  
erantzuna = requests.request(metodoa, uria, data=edukia_encoded,  
                             headers=goiburuak, allow_redirects=False)
```

```
kodea = erantzuna.status_code  
deskribapena = erantzuna.reason  
print(str(kodea) + " " + deskribapena)  
for goiburua in erantzuna.headers:  
    print(goiburua + ": " + erantzuna.headers[goiburua])  
edukia = erantzuna.content  
print(edukia)
```

Datuen
kodifikazioa

ADIBIDEA:

DATU BIDALKETA INPRIMAKI FORMATUAN -- GET

- NAN letraren zenbakia lortu: <http://ws-sendingformdata.appspot.com/>

Bezeroaren eskaera

GET /processForm?nan=12345678 HTTP/1.1
Host: ws-sendingformdata.appspot.com

Zerbitzariaren erantzuna

HTTP/1.1 200 OK
Date: Wed, 25 Nov 2015 08:07:43 GMT
Server: Google Frontend
Content-type: text/html; charset=utf-8
Content-Length: 1

Z

```
import requests  
import urllib  
import sys
```

```
metodoa = 'GET'  
base_urria = "http://ws-sendingformdata.appspot.com/processForm"  
goiburuak = {'Host': 'ws-sendingformdata.appspot.com'}  
edukia = {'nan': sys.argv[1]}  
edukia_encoded = urllib.parse.urlencode(edukia)  
urria = base_urria + '?' + edukia_encoded  
erantzuna = requests.request(metodoa, urria,  
                             headers=goiburuak, allow_redirects=False)
```

```
kodea = erantzuna.status_code  
deskribapena = erantzuna.reason  
print(str(kodea) + " " + deskribapena)  
for goiburua in erantzuna.headers:  
    print(goiburua + ": " + erantzuna.headers[goiburua])  
edukia = erantzuna.content  
print(edukia)
```

Datuak
URLan

HTTP-REN FUNTZIONAMENDUA: INPRIMAKIAK (HTML)



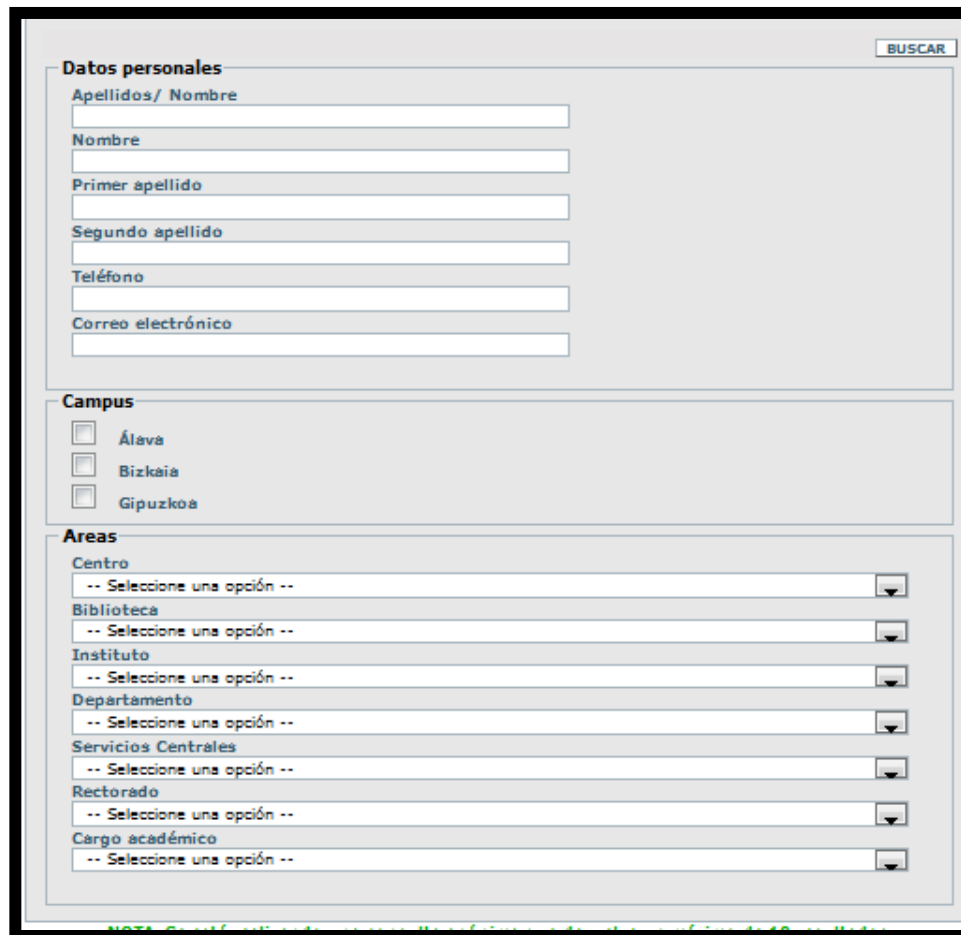
```
<html>
  <head>
    <title>Formularioa</title>
  </head>
  <body>
    <p>
      <form method="GET" action="/app/servlet/contactUs">
        <b>Izena: </b>
        <input type="text" name="name">
        <br/>
        <b>Posta elektronikoa: </b>
        <input type="text" name="email">
        <br/>
        <b>Mezua: </b>
        <input type="text" name="message">
        <br/><br/>
        <input type="submit">
      </form>
    </p>
  </body>
</html>
```

(*) image extracted from <http://www.freedback.com/>

ARIKETA: KONTSULTA EZAZU EHUko DIREKTORIOA

BURP ETA PYTHON ERABILIZ

- <https://www.ehu.eus/bilatu/buscar/bilatu.php?lang=es>



The image shows a web form for searching the EHU directory. It is divided into three main sections: 'Datos personales', 'Campus', and 'Areas'. The 'Datos personales' section contains input fields for 'Apellidos/ Nombre', 'Nombre', 'Primer apellido', 'Segundo apellido', 'Teléfono', and 'Correo electrónico'. The 'Campus' section has three checkboxes for 'Álava', 'Bizkaia', and 'Gipuzkoa'. The 'Areas' section contains several dropdown menus for 'Centro', 'Biblioteca', 'Instituto', 'Departamento', 'Servicios Centrales', 'Rectorado', and 'Cargo académico'. A 'BUSCAR' button is located in the top right corner of the form.

Datos personales

Apellidos/ Nombre

Nombre

Primer apellido

Segundo apellido

Teléfono

Correo electrónico

Campus

☐ Álava

☐ Bizkaia

☐ Gipuzkoa

Areas

Centro

-- Seleccione una opción --

Biblioteca

-- Seleccione una opción --

Instituto

-- Seleccione una opción --

Departamento

-- Seleccione una opción --

Servicios Centrales

-- Seleccione una opción --

Rectorado

-- Seleccione una opción --

Cargo académico

-- Seleccione una opción --

BUSCAR