

resolucionExamenDAI.pdf



Rubenjo7



Apuntes Variados



4º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada



MÁSTER EN

Inteligencia Artificial & Data Management

MADRID

Formamos
talento para un futuro
Sostenible

saber más



Importante

Puedo eliminar la publi de este documento con 1 coin

¿Cómo consigo coins? → Plan Turbo: barato
→ Planes pro: más coins

perdo
espacio



Necesito
concentración

ali ali ooh
esto con 1 coin me
lo quito yo...

WUOLAH

Desarrollo de Aplicaciones para Internet

Resolución del Examen de Teoría - Convocatoria Febrero de 2014

Aclaración Inicial: Este documento ofrece una posible resolución al examen de teoría que alcanzara la máxima nota de 10 puntos. Sin embargo, dado el carácter abierto de las preguntas existen muchas posibilidades para responderlas que igualmente obtendrán la máxima calificación.

1. Describa brevemente como funciona el mecanismo de sesiones en una web dinámica.

Las sesiones son un mecanismo por el cual es posible almacenar información en el *servidor* relacionada con la navegación de cada cliente (por ejemplo las páginas visitadas, datos sobre un usuario autenticado, etc). Para que el servidor sepa que cliente está haciendo cada petición concreta el servidor negocia con el cliente el establecimiento de una *cookie* que será transmitida en cada petición al servidor.

2. Comente de manera breve algunas alternativas para guardar información en el CLIENTE de una aplicación web.

La manera más habitual es utilizar *cookies*, que son pequeños ficheros con una fecha de caducidad asociada y que almacena el programa cliente (navegador usualmente).

Otra posibilidad es utilizar el mecanismo de persistencia local (*Web Storage*) disponible a partir de HTML5.

Un método posible (poco práctico, pero que podría implementarse) sería almacenar la información a través de los parámetros de los métodos GET o POST que intercambian cliente y servidor.

3. Comente de manera breve cuáles son las diferencias de tratamiento de cadenas Unicode entre Python 3.1 y Python 2.7.

La diferencia fundamental es que por defecto Python 2.7 *no* utiliza cadenas Unicode (si quieres utilizarlas tienes que especificarlo mediante el operador `u` ante la cadena en cuestión) mientras que en la versión de Python 3.1 por defecto todas las cadenas son Unicode.

4. Estamos desarrollando una aplicación con Webpy. Encontramos el siguiente código:

```
urls = (  
    '/sun(.*)', 'Sun',  
    '/moon(.*)', 'Moon',  
    '/()', 'Moon',  
)  
  
app = web.application(urls, globals())
```

¿Qué significado tiene la variable `urls`? ¿Qué URLs se atenderán y cómo?

La variable `urls` se utiliza para hacer un emparejamiento entre las *URLs* que atenderá la aplicación web y las *clases* definidas en nuestra aplicación que gestionarán las peticiones web (con sus respectivos métodos GET y POST). La variable `urls` contiene parejas de cadenas. La primera de cada pareja corresponde con una expresión regular que identifica las posibles *URLs* que se atenderán y la segunda es el nombre de la *clase* que la atenderá.

En el caso particular del ejemplo que se presenta y, suponiendo que nuestra aplicación está alojada en el dominio `example.com`, que atiende peticiones por el puerto 80 usando el protocolo `http`, tendremos que la aplicación:

- Responder a cualquier URL que *comience* por `http://example.com:80/sun` usando la clase `Sun`.

- Responder a cualquier URL que *comience* por `http://example.com:80/moon` usando la clase `Moon`.
- Responder a la petición de la página principal (`http://example.com:80/`) usando también la clase `Moon`.

5. En las prácticas de la asignatura se ha desarrollado un sitio web usando plantillas. Describa brevemente la estructura de directorios y archivos que permiten que dicho sitio web funcione.

En las prácticas se ha desarrollado una aplicación web usando la biblioteca `Mako` (para la gestión de plantillas) y la biblioteca `Webpy` para generar la aplicación en `s`.

Para la aplicación hace falta al menos un archivo `.py` en el raíz de la aplicación donde se definan las propiedades básicas de la aplicación (`web.application(urls, globals())...`) así como las *clases* que gestionen las peticiones.

Asimismo es normal que tengamos una carpeta de contenidos *estáticos* (usualmente llamada `static`) como serían por ejemplo imágenes, hojas de estilo (CSS) o archivos con código que se ejecutan en el cliente (Javascript).

Por último, para gestionar las plantillas usualmente tendremos una carpeta donde se almacenarán las mismas. Dicha carpeta se suele llamar `templates`. Dentro tendremos distintos archivos HTML con pequeños *snippets* que rellenarán la información dinámica de la página.

```
|-- Raíz de la Aplicación
|
|-- aplicacion.py
|-- otrasPosiblesClases.py
|
| .
| .
| .
|-- static
|
| |
| | |-- imagen.png
| | |-- imagen2.jpg
| | |-- hojaEstilo.css
| | |
| | | .
| | | .
| | | .
| | |-- funciones.js
| | |-- jquery.js
|
|-- templates
|
| |-- basica.html
| |-- cabecera.html
| |
| | .
| | .
| | .
|-- pie.html
```

6. Existen distintas alternativas para tratar un archivo XML (RSS, por ejemplo) en nuestra aplicación web. Comente alguna de ellas y qué ventajas e inconvenientes tienen.

- **lxml “Sax Parser”:** Los mecanismos **SAX** recorren de manera secuencial el archivo XML etiqueta a etiqueta generando eventos cada vez que cada una de ellas aparece (por ejemplo eventos al abrir o cerrar etiquetas, al encontrar el contenido de una etiqueta o al finalizar el archivo). Tiene como ventaja que no necesita cargar el archivo completo en memoria (muy útil con XMLs muy grandes), pero tiene como desventaja la gestión de los distintos niveles de anidamiento (que tienen que ser controlados por el programador).
- **lxml “etree Parser”:** En este caso se maneja el XML mediante el uso de una biblioteca de manejo del DOM. El archivo se carga completo en memoria en una estructura de árbol (puede ser

Imagínate aprobando el examen

Necesitas tiempo y concentración

Planes	 PLAN TURBO	 PLAN PRO	 PLAN PRO+
 Descargas sin publi al mes	10 	40 	80 
 Elimina el video entre descargas			
 Descarga carpetas			
 Descarga archivos grandes			
 Visualiza apuntes online sin publi			
 Elimina toda la publi web			
 Precios Anual <input type="checkbox"/>	0,99 € / mes	3,99 € / mes	7,99 € / mes

Ahora que puedes conseguirlo,
¿Qué nota vas a sacar?



WUOLAH

Apuntes Variados



Banco de apuntes de la

WUOLAH



Comparte estos flyers en tu clase y consigue más dinero y recompensas

- 1** Imprime esta hoja
- 2** Recorta por la mitad
- 3** Coloca en un lugar visible para que tus compis puedan escanar y acceder a apuntes
- 4** Llévate dinero por cada descarga de los documentos descargados a través de tu QR



problemático si el fichero es grande) que luego puede ser consultada y recorrida como se desee (mayor facilidad de tratamiento).

- **Bibliotecas especializadas:** Si el fichero XML a tratar es un subconjunto estandarizado (como por ejemplo el protocolo RSS) podemos utilizar bibliotecas específicas para el tipo concreto de contenido. Por ejemplo, para RSS tenemos la biblioteca *Universal Feed Parser*. Este tipo de bibliotecas simplifican mucho la gestión y análisis del documento.