F Seguir Sé el primero de tus amigos

en seguir a Miríada X.

Mi Página | Cursos | Universidades e instituciones | Conócenos | Soporte

Desarrollo de servicios en la nube con HTML5, Javascript y node.js

Inicio Syllabus Foro Blog

Módulos

- Modulo 0. Introducción al curso, al programa y al Sistema Operativo UNIX
- Modulo 1. Introducción a JavaScript de servidor y a node.js. Sentencias, Variables, Booleanos, Números, Strings y Funciones

Modulo 2. Introducción a JavaScript de

- servidor y a node.js. Bucles, Clases predefinidas, Objetos, Propiedades y Métodos; Prototipos y Clases; Arrays; JSON; Funciones como Objetos y Cierres (Closures) Modulo 3. Modulos node.js;
- Expresiones Regulares; Eventos, Entorno de Ejecución y Concurrencia en node.js; Ficheros y Flujos Modulo 4. Introducción a HTTP y a los
- Servidores Web; Introducción a express y al Middleware Static; Introducción a REST; Aplicaciones express.js y Composición de Middlewares; Formularios GET y POST; **Parámetros Ocultos**
 - Tema 0: Transparencias del módulo
- Tema 1. introducción a HTTP
- Tema 1. Cuestionario obligatorio
- Tema 2. Servidor Web Tema 2. Cuestionario obligatorio
- Tema 3. Introducción a express.js y al middelware static
- Tema 3. Cuestionario obligatorio Tema 4. Introducción a REST
- Tema 4. Cuestionario obligatorio
- Tema 5. Aplicaciones REST con express.js
- Tema 5. Cuestionario obligatorio
- Tema 6. Acceso a campos de la ruta
- Tema 6. Cuestionario obligatorio Tema 7. Composición y ejecución de
- middlewares Tema 8. Formulario GET
- Tema 9. URL encode
- Tema 10. Formulario POST
- override Ejercicio P2P Obligatorio

Modulo 5. Gestión de versiones de

proyectos con git y GITHUB; Proyecto,

Tema 11. Parámetro oculto y method

(Commit); Arboles y Ramas de un proyecto; Repositorios Remoto y colaboración a través de GITHUB

Espacio de Trabajo y Versiones

Modulo 6. Proyecto Quiz I: Patrón Modelo-Vista-Controlador (MVC); generación del proyecto con expressgenerator; Primera Página y Primera Pregunta; Despliegue en la nube (Heroku) Modulo 7. Proyecto Quiz II: La Base de

Datos (DB), Tablas, sequelize.js y

SQLite; Despliegue en Heroku

Listas de Quizes y Autoload Modulo 8. Proyecto Quiz III: Gestión de Listas de Quizes, Creación, Edición y Borrado

Modulo 9. Proyecto Quiz IV: Creación y

Moderación de Comentarios a Quizes;

utilizando Postgres; Presentación de

Relaciones entre Tablas de la Base de Datos; Sesiones, Autenticación y Autorización; HTTP Seguro (HTTPS)

Modulo 4. Introducción a HTTP y a los Servidores Web; Introducción a express y al Middleware Static; Introducción a REST; Aplicaciones express.js y Composición de Middlewares; Formularios GET y POST; Parámetros Ocultos Prueba realizada

servidor.

Tu resultado en el test ha sido: 80%

Tus respuestas

Has superado el test.

Método que puede ser cacheado porque no modifica datos o recursos en el

Indicar qué respuesta define correctamente el término: Interfaz uniforme

- Dirección que identifica sin ambigüedad un recurso en Internet. Unidad de información gestionada en una aplicación REST con el interfaz
- uniforme.
- Comando seguro de HTTP para traer recursos al cliente para su visualización.
- Método HTTP cuyo resultado es independiente del número de veces que se invoca.
- Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE.
- Comando idempotente de HTTP para borrar recursos en un servidor.
- Comando idempotente de HTTP para editar recursos en un servidor.

Comando no idempotente de HTTP para crear recursos en un servidor.

- Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de HTTP.
- ncorrect 1
- Indicar qué respuesta define correctamente el término: GET

servidor.

 Dirección que identifica sin ambigüedad un recurso en Internet. Unidad de información gestionada en una aplicación REST con el interfaz

Método que puede ser cacheado porque no modifica datos o recursos en el

- Comando seguro de HTTP para traer recursos al cliente para su visualización.
- Método HTTP cuyo resultado es independiente del número de veces que se invoca.
- Comando no idempotente de HTTP para crear recursos en un servidor. Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE.
- Comando idempotente de HTTP para borrar recursos en un servidor. Comando idempotente de HTTP para editar recursos en un servidor.
- Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de HTTP.
- Indicar qué respuesta define correctamente el término: Método seguro

Método que puede ser cacheado porque no modifica datos o recursos en el

HTTP.

Correct

uniforme.

servidor. Dirección que identifica sin ambigüedad un recurso en Internet.

- Unidad de información gestionada en una aplicación REST con el interfaz uniforme.
- Método HTTP cuyo resultado es independiente del número de veces que se invoca. Comando no idempotente de HTTP para crear recursos en un servidor.

Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE.

Comando seguro de HTTP para traer recursos al cliente para su visualización.

- Comando idempotente de HTTP para borrar recursos en un servidor. Comando idempotente de HTTP para editar recursos en un servidor.
- Correct

Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de

Método que puede ser cacheado porque no modifica datos o recursos en el servidor.

Dirección que identifica sin ambigüedad un recurso en Internet. Unidad de información gestionada en una aplicación REST con el interfaz

Indicar qué respuesta define correctamente el término: REST

uniforme. Comando seguro de HTTP para traer recursos al cliente para su visualización.

Método HTTP cuyo resultado es independiente del número de veces que se

invoca. Comando no idempotente de HTTP para crear recursos en un servidor.

Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE.

- Comando idempotente de HTTP para borrar recursos en un servidor.
- Comando idempotente de HTTP para editar recursos en un servidor.

Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de

Incorrect Indicar qué respuesta define correctamente el término: Recurso REST

Método que puede ser cacheado porque no modifica datos o recursos en el

Dirección que identifica sin ambigüedad un recurso en Internet.

HTTP.

servidor.

invoca.

HTTP.

uniforme.

Correct

uniforme.

7 Correct

servidor.

invoca.

uniforme.

HTTP.

Unidad de información gestionada en una aplicación REST con el interfaz uniforme.

- Comando seguro de HTTP para traer recursos al cliente para su visualización.
- Método HTTP cuyo resultado es independiente del número de veces que se invoca. Comando no idempotente de HTTP para crear recursos en un servidor.

Comando idempotente de HTTP para editar recursos en un servidor.

Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE. Comando idempotente de HTTP para borrar recursos en un servidor.

Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de

7 Correct

Indicar qué respuesta define correctamente el término: Método idempotente

Dirección que identifica sin ambigüedad un recurso en Internet. Unidad de información gestionada en una aplicación REST con el interfaz uniforme.

Comando seguro de HTTP para traer recursos al cliente para su visualización.

Método HTTP cuyo resultado es independiente del número de veces que se

Método que puede ser cacheado porque no modifica datos o recursos en el

- Comando no idempotente de HTTP para crear recursos en un servidor. Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE.
- Comando idempotente de HTTP para borrar recursos en un servidor. Comando idempotente de HTTP para editar recursos en un servidor. Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de
- 7 Correct Indicar qué respuesta define correctamente el término: PUT

Método que puede ser cacheado porque no modifica datos o recursos en el

- servidor. Dirección que identifica sin ambigüedad un recurso en Internet. Unidad de información gestionada en una aplicación REST con el interfaz
- Comando no idempotente de HTTP para crear recursos en un servidor. Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE

Comando idempotente de HTTP para borrar recursos en un servidor.

Método HTTP cuyo resultado es independiente del número de veces que se

Comando seguro de HTTP para traer recursos al cliente para su visualización.

- Comando idempotente de HTTP para editar recursos en un servidor. Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de HTTP.
- Indicar qué respuesta define correctamente el término: DELETE

Método que puede ser cacheado porque no modifica datos o recursos en el

Unidad de información gestionada en una aplicación REST con el interfaz

Comando seguro de HTTP para traer recursos al cliente para su visualización. Método HTTP cuyo resultado es independiente del número de veces que se

Dirección que identifica sin ambigüedad un recurso en Internet.

Comando idempotente de HTTP para editar recursos en un servidor. Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de HTTP.

Comando no idempotente de HTTP para crear recursos en un servidor.

Comando idempotente de HTTP para borrar recursos en un servidor.

Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE.

Indicar qué respuesta define correctamente el término: POST

Método que puede ser cacheado porque no modifica datos o recursos en el

Unidad de información gestionada en una aplicación REST con el interfaz uniforme. Comando seguro de HTTP para traer recursos al cliente para su visualización.

Dirección que identifica sin ambigüedad un recurso en Internet.

Comando no idempotente de HTTP para crear recursos en un servidor. Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE.

Comando idempotente de HTTP para borrar recursos en un servidor.

Comando idempotente de HTTP para editar recursos en un servidor.

Indicar qué respuesta define correctamente el término: URL

Método HTTP cuyo resultado es independiente del número de veces que se

HTTP. 7 Correct

Método que puede ser cacheado porque no modifica datos o recursos en el

Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de

- servidor. Dirección que identifica sin ambigüedad un recurso en Internet. Unidad de información gestionada en una aplicación REST con el interfaz
- Método HTTP cuyo resultado es independiente del número de veces que se invoca. Comando no idempotente de HTTP para crear recursos en un servidor.

Comando seguro de HTTP para traer recursos al cliente para su visualización.

- Subconjunto de los métodos de HTTP que incluye GET, PUT, POST y DELETE. Comando idempotente de HTTP para borrar recursos en un servidor. Comando idempotente de HTTP para editar recursos en un servidor.
- Interfaz de aplicación Web que solo utiliza los métodos del interfaz uniforme de HTTP.
- Has realizado 1 de 10 intentos ¿Desea volver a acceder al test?

anterior

Política de cookies

Jelefónica

Correct

Mejorar nota

Santander

Política de privacidad

uni>ersia

Siguiente 2

Telefónica Educación Digital

2012-2015 Miríada X Aviso legal