

**Universidad de La Laguna. Escuela Técnica Superior de Ingeniería Informática**  
**Tercero del Grado de Informática**  
**DESARROLLO DE SISTEMAS INFORMÁTICOS: 2ª PARTE**  
04/04/2017 6 páginas

Nombre: \_\_\_\_\_  
Alu: \_\_\_\_\_ GitHub Id: \_\_\_\_\_ GitHub Team: \_\_\_\_\_

### **Preguntas de Repaso de Heroku**

1. Una vez instalado el Heroku cli nos debemos autenticar en heroku con el cliente. ¿Cual es el comando para autenticarnos?
2. ¿Cual es el comando para crear nuestra aplicación en Heroku (suponemos que ya esta bajo el control de `git`? ¿Qué remoto tendremos definido después de crear nuestra aplicación en Heroku?
3. ¿Cual es el comando para desplegar nuestra aplicación en Heroku?
4. Si la versión que queremos publicar en heroku no está en la rama `master` sino que está en la rama `tutu` ¿Como debemos modificar el comando anterior?
5. ¿Con que comando puedo abrir una ventana en el navegador que visite la aplicación desplegada? ¿Que formato tiene la URL para nuestra aplicación?
6. ¿Con que comando puedo ver los logs de la aplicación desplegada?
7. ¿Como se debe llamar el fichero en el que explico que comando debe usarse para arrancar nuestra aplicación en Heroku?
8. Heroku se percata que nuestra aplicación es una aplicación desarrollada con `Node.js` por la presencia de un cierto fichero. ¿De que fichero estamos hablando?
9. ¿Cual es la mejor forma de ejecutar en local una aplicación `express.js` que va a ser desplegada en Heroku?
10. Explique los pasos para desplegar una aplicación en Heroku
11. Explique como resolver los problemas que pueden surgir cuando la aplicación desplegada en Heroku no funciona correctamente

## Preguntas de Repaso de Expresiones Regulares

1. Escriba una expresión JavaScript que permita reemplazar todas las apariciones de palabras consecutivas repetidas (como `hello hello`) por una sólo aparición de la misma
2. ¿Cual es la salida?

```
> "bb".match(/b|bb/)
```

```
> "bb".match(/bb|b/)
```

Justifique su respuesta.

3. El siguiente fragmento de código tiene por objetivo escapar las entidades HTML para que no sean interpretadas como código HTML. Rellene las partes que faltan.

```
var entityMap = {
  "&": "&__;",
  "<": "&__;",
  ">": "&__;",
  "'": '&quot;',
  '"': '&#39;',
  "/": '&#x2F;';
};

function escapeHtml(string) {
  return String(string).replace(/_____/g, function (s) {
    return _____;
  });
}
```

## Preguntas de Repaso de Cookies

1. Defina y explique el concepto de Cookie
2. Defina y explique el concepto de Cookie para la Gestión de una Sesión
3. Defina y explique el concepto de Cookie para la Autenticación en un Website
4. ¿Que es un secure cookie? (cookie seguro)

A secure cookie has the secure attribute enabled and is only used via HTTPS, ensuring that the cookie is always encrypted when transmitting from client to server. This makes the cookie less likely to be exposed to cookie theft via eavesdropping.

4. ¿Como se llama el middleware express que me facilita el manejo de los cookies? ¿Cual es el código para poner en funcionamiento dicho middleware?

```
% cookie-parser
% var cookieParser = require('cookie-parser');
% app.use(cookieParser());
```

5. Dentro de un middleware express ¿Cómo se llama el método del objeto `res` que me permite establecer un cookie? ¿Que argumentos recibe?

```
% res.cookie('cookie_name', 'cookie_value', {expire : new Date() + 9999}).send(
%     "Cookie is set: goto to browser's console and write document.cookie.");
```

6. ¿Que método del objeto `request` me permite ver los cookies establecidos?

```
console.log("Cookies : ", req.cookies);
```

7. ¿Como borro un cookie en el servidor express?

```
res.clearCookie('cookie_name');
```

8. ¿Que es la Query String? Ponga un ejemplo de Query String

On the World Wide Web, a query string is the part of a uniform resource locator (URL) containing data that does not fit conveniently into a hierarchical path structure. The query string commonly includes fields added to a base URL by a Web browser or other client application, for example as part of an HTML form. A typical URL containing a query string is as follows:

```
http://example.com/over/there?name=ferret
```

9. If a form is embedded in an HTML page as follows:

```
<form action="/hello" method="get">
  <input type="text" name="first" />
  <input type="text" name="second" />
  <input type="submit" />
</form>
```

and the user inserts the strings `this is a field` and `was it clear already` in the two text fields and presses the `submit` button, the handler of the route `/hello` (the route specified by the `action` attribute of the form element in the above example) will receive a query string. Write down the query string it receives

```
first=this+is+a+field&second=was+it+clear+already
```

## Rutas en Express

1. Escriba un servidor que sirva ficheros estáticos desde el directorio `/public`
2. El servidor deberá responder a requests `GET` en las rutas `/user/nombredeusuario` (donde `nombredeusuario` varía) con una página que diga `Hola nombredeusuario` usando el método `render` del objeto `response`
  - La página se elaborara con una vista que debe estar en el directorio `views/` usando el motor de vistas `ejs`
  - La página elaborada en la respuesta tendrá un tag `img` referenciando a una imagen que está en `public/`
3. Escriba un middleware que intercepte en las rutas `/user/nombredeusuario` y que vuelque en la consola información sobre el `request`: (por ejemplo los atributos `method`, `path`, etc.)
4. Explique como se puede aislar el código anterior en un fichero `routes/user.js` que sea cargado desde el programa principal
5. Explique que hay que hacer para desplegar la aplicación en la máquina virtual del iaas

## Preguntas de Repaso de Mocha, Chai

1. ¿Como creamos el directorio con el esqueleto inicial para las pruebas con mocha?
2. Rellene lo que falta en estas pruebas del código del conversor de temperatura:

```
var assert = chai._____;

suite('temperature', function() {
  test('[1,{a: 2}] == [1,{a: 2}]', function() {
    assert._____( [1, {a:2}], [1, {a:2}]);
  });
  test('5X = error', function() {
    original.value = "5X";
    calculate();
    assert._____(converted.innerHTML, /ERROR/);
  });
});

var assert = chai.assert;
```

3. Este es un fichero `test/index.html` apto para ejecutar las pruebas con Mocha y Chai en la práctica de la Temperatura. Rellene las partes que faltan:

```
[~/srcPLgrado/temperature(karma)]$ cat tests/index.html
<!DOCTYPE html>
<html>
  <head>
    <title>Mocha</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="mocha.css" />
  </head>
  <body>
    <div id="____"></div>    <!-- para la salida de las pruebas -->
    <input id="original" placeholder="32F" size="50">
    <span class="output" id="converted"></span>

    <script src="_____"></script>
    <script src="_____"></script>
    <script src="../temperature.js"></script>
    <script>mocha.____('___')</script>
    <script src="tests.js"></script>

    <script>
      mocha.____();
    </script>
  </body>
</html>
```

4. Rellene las partes que faltan del fichero con las pruebas TDD en Mocha y Chai para la práctica de la temperatura:

```
[~/srcPLgrado/temperature(karma)]$ cat tests/tests.js
var assert = _____.assert;

_____('temperature', function() {
  _____('32F = 0C', function() {
    original.value = "32F";
    calculate();
    assert._____(converted.innerHTML, "0.0 Celsius");
  });
});
```

5. ¿Como puedo ejecutar las pruebas escritas usando Mocha y Chai usando el comando `npm test`?. (no se asume el uso de Karma en esta pregunta) Explique como hacerlo.