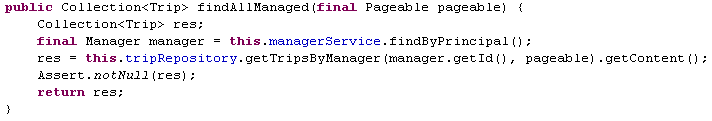
**CÓMO PAGINAR**

Paso 1: Para cada consulta que devuelva una colección de algo que luego se vaya a mostrar en una vista, hay que hacer dos queries en el repositorio. Una de ellas será igual a la query original, pero en lugar de devolver una Collection devolverá una Page, y además hay que añadir un objeto Pageable como parámetro adicional. La segunda query debe sacar el número total de elementos que devolvería la otra query si no estuviera paginada, este número sirve para que luego el Display Tag sepa cuántas páginas va a haber en total.





Paso 2: En el servicio correspondiente, hay que hacer un método para cada una de las queries, como siempre. El que devuelve el número total de elementos consistirá simplemente en llamar al método del repositorio y devolver el Integer, teniendo en cuenta los parámetros que hagan falta (por ejemplo, en el caso de estas capturas, el ID del mánager se saca por el findByPrincipal y se pasa a la query directamente dentro del método, sin que haga falta pasar un ID como parámetro al método del servicio). Para la query que devuelve una Page, el método en el servicio devolverá ya una Collection, que se saca con el método getContent. Hay que poner como parámetro el objeto Pageable, para pasárselo luego desde el controlador.



Para los tests de listado habrá que crear el objeto Pageable. En la parte donde se explica lo que hay que hacer en los controladores aparece cómo hay que crearlo.

Paso 3: En la vista, hay que añadir algunas cosas a la etiqueta display:table. Hay que meter los atributos “sort”, “partialList” y “size” con los mismos valores de la captura (evidentemente, la variable del “size” dependerá del nombre que se use en el controlador, pero mejor que todos usemos el mismo). Por otro lado, para cada columna que pueda ser ordenada, hay que añadir a la etiqueta display:column correspondiente el atributo “sortName” y un valor, lo normal para no liarse es usar el mismo nombre de la propiedad que se va a ordenar.





Paso 4: En el controlador, lo primero que hay que hacer es añadir el parámetro HttpServletRequest al método correspondiente. Las variables nuevas que hay que inicializar son el objeto Pageable, un objeto Direction (a null) y un Integer que será el número de página (a 0). Después, hay que sacar tres Strings desde el objeto HttpServletRequest, tal y como aparece en la captura, para sacar parámetros que le hacen falta al Display Tag. En principio, lo único que hay que tener en cuenta en esas tres líneas es lo de “row”, que debe ser igual que el ID que se ha puesto en la etiqueta display:table en la vista y por lo tanto habrá que cambiarlo si se pone una ID distinta. Luego, la parte de los if-else debería entenderse echándole un vistazo. Se deben poder copiar todas esas líneas sin que haya ningún problema. Lo único que se puede cambiar ahí es la creación del objeto Pageable. Para crearlo, se usa el constructor de la clase PageRequest, y, según el número de parámetros que se le pase (4 ó 2), se considera que se está ordenando la lista o no. Si son 4, los parámetros son: número de página (empezando en 0), número de elementos por página (importante que sea igual que el “pagesize” que se ha puesto en la vista), dirección de ordenación, y atributo de ordenación. Si son 2, solo son: número de página, y número de elementos por página. Por lo tanto, lo que hay que tener en cuenta aquí es el número de elementos por página, que, como se ha indicado antes, tiene que ser igual que el “pagesize” de la etiqueta “display:table”. Una vez que se ha construido el objeto Pageable, se pasa al método del servicio y se obtiene la Collection correspondiente. También hay que calcular el total de elementos de la lista completa, usando el método que se creó antes, el que devolvía un Integer. Finalmente, solo queda pasar la colección y el número total de elementos a la vista.



Paso 5: Disfrutar de la eficiencia extrema.