

Introducción

El ejercicio consta de 2 partes: generar una serie de objetos (o mapas), *Tren*, con las propiedades **id**, **salida**, **sinDevolucion**, **conDevolucion** y **normal** y, una vez agrupados estos objetos/mapas, convertirlos en una cadena con la **coma** como separador entre campos; el símbolo **#** como separador entre trenes y el símbolo **%** como separador entre días.

Primera parte

Implementar el método **rellenarDias(fechaInicio, fechaFinal)** que generará, para cada día comprendido entre sus 2 argumentos (ambos inclusive), los objetos *Tren* de ese día con su correspondiente horario. El horario de cada día se encuentra en un array, sobre el que se debe trabajar:

['09:40', '10:40', '12:20', '13:10', '15:10', '16:40', '17:40', '18:40', '19:30', '21:40'];

Existen, además del horario anterior, dos más: si es sábado, hay un tren a las '08:40'; si es domingo, uno a las '20:10'. La generación de estos últimos se hará como casos especiales al horario de cada día. El método devolverá un array en el que cada elemento contendrá los trenes de un día (10 por día salvo sábados y domingos, con 11).

El argumento **fechaInicio** deberá ser mayor al día actual; el argumento **fechaFinal** deberá ser mayor o igual al argumento **fechaInicio**. Si no es así, se devolverá un array vacío.

[1 punto] Para generar los objetos/mapas *Tren*, **rellenarDias** llamará a **rellenarTren(dia, horario)** que creará y devolverá un objeto o mapa con los campos especificados en la introducción. El contenido de esos campos será: **id** → una cadena compuesta del día (primer argumento), con dos dígitos, más la cadena del horario (2º argumento) sin los dos puntos; **salida** → una cadena con el horario (2º argumento) tal y como aparece en el array de horarios; **sinDevolucion**, **conDevolucion** y **normal** son campos numéricos, generados de forma aleatoria, con el número de billetes disponibles en cada categoría teniendo en cuenta que cada tren tiene 90 asientos y que la suma de los billetes **sinDevolucion** y **conDevolucion** es, como máximo, el 50% del total. Si falta algún argumento, el objeto devuelto tendrá todos sus campos undefined.

Para apoyar a los 2 métodos anteriores, se deben crear dos funciones: una que devuelva un número entero aleatorio comprendido entre dos valores enteros, ambos incluidos, pasados como argumentos; y otra, sin argumentos, que devuelva una fecha un mes (ni 30 ni 31 ni 28 días, un mes) en el futuro a partir del día siguiente de la fecha actual.

Segunda parte

Implementar el método **formarCadenaDias(arrayDias)** que, recibiendo el array que genera **rellenarDias**, devuelva una cadena resultante de convertir todos los datos del array. Los separadores entre campos, trenes y días se han indicado en la introducción. Si el array de entrada está vacío, se devolverá una cadena vacía.

Extra

Si en la solución no se utilizan variables/métodos/funciones globales

Si en la solución, para acceder a los arrays con los que se trabaja, se utilizan exclusivamente métodos de array (nada de estructuras for, while, etc)