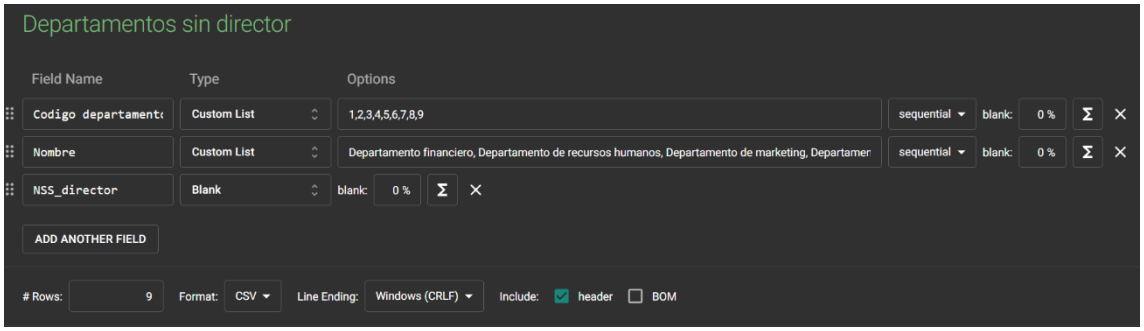
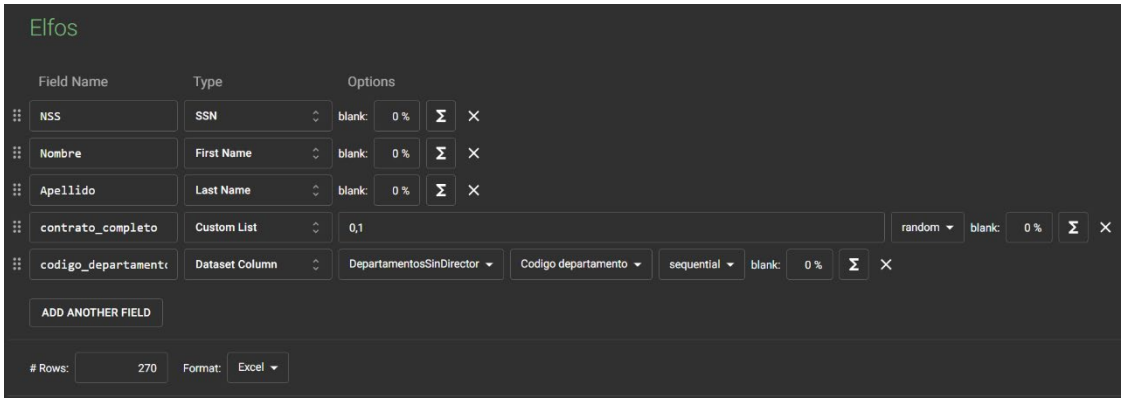
Creación de datos

Primero de todo cogemos de internet varios nombres de departamentos de los que suele componerse una empresa y creamos estos departamentos sin asignarles un director.



Posteriormente generamos los elfos de modo que haya el mismo número de ellos en cada departamento, a sabiendas que esta ha sido una decisión arbitraria, simplemente con el objetivo de asegurarnos de que todos los departamentos tienen al menos un elfo. Existe la posibilidad de que no haya elfos en un departamento con contrato a tiempo completo, pero ese hecho sería localizado posteriormente a la hora de generar los directores, en tal caso deberíamos volver a generar los datos de los elfos hasta que se cumpla que todos los departamentos tienen al menos un elfo con contrato a tiempo completo.

A partir de estos datos nos quedamos por separado solo con su tipo de contrato y su NSS, extrayendo estas dos columnas del archivo de Excel generado, lo que, con ayuda de la herramienta ordenar d, nos permitirá crear una lista con los NSS de los elfos que tienen un tipo de contrato, ordenando la columna contrato_completo. Incluiremos en Mockaroo estas dos listas como *datasets*, además del archivo inicial con todos los elfos. Para hacer lo anterior basta con guardar los archivos de Excel como archivos csv.

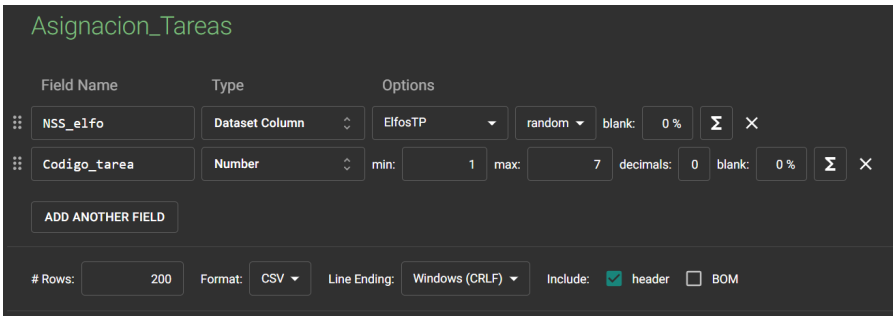


Pasamos a crear una base de datos (bruto) que incluya a todos los elfos y a los niños, que más adelante explicaremos cómo los generamos.

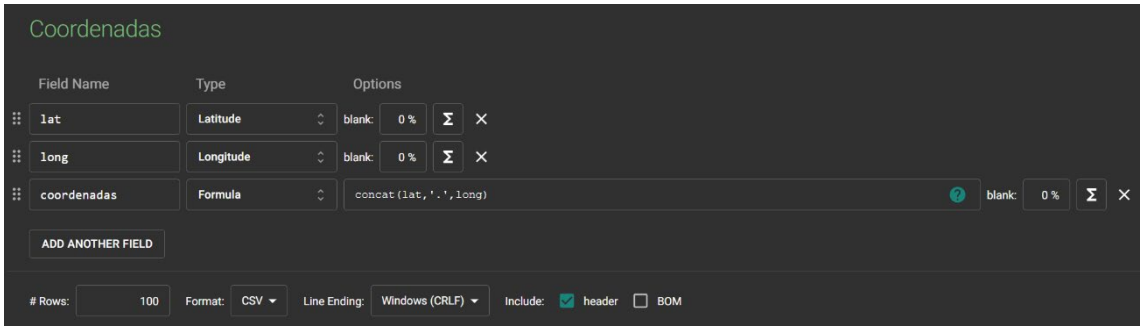
Para introducir los directores de los departamentos usamos el archivo Directores_NiñosSinRep.py, el cual toma los elfos generados y selecciona aleatoriamente un elfo de cada departamento que tenga contrato a tiempo completo.

Se nos ocurren una serie de tareas que podrían estar relacionadas con el trabajo de los elfos, así que las añadimos manualmente.

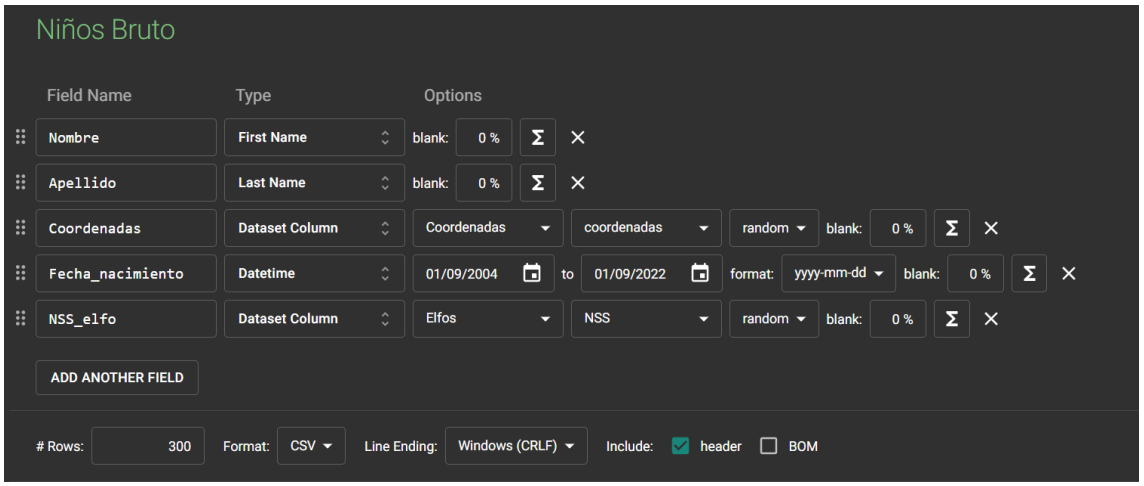
Para la creación de la lista de tareas asignadas simplemente creamos una combinación aleatoria de un NSS de elfos a tiempo parcial y el código de la tarea que le asignamos.



Previo a crear los datos de los niños, generamos una lista de coordenadas que guardamos como base de datos (con menor número de entradas que de niños), pues de esta forma nos aseguramos de que habrá niños que vivan juntos.



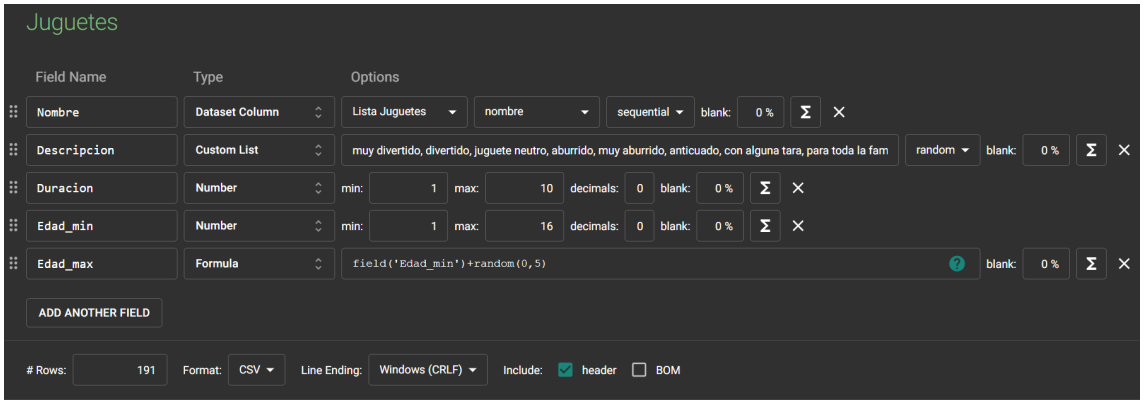
A la hora de crear a los niños, debido a que los elfos solo se pueden observar a un solo niño, debemos revisar bien estos datos, pues luego tendremos que crear otros datos con ellos. Este filtro de los datos lo realizamos con el archivo Directores_NiñosSinRep.py, con el que generamos una lista de niños sin repeticiones en NSS_elfo.



La lista de los nombres de juguetes la sacamos de la página web https://www.listchallenges.com/print-list/303585 y la introducimos como *dataset* a Mockaroo.

A esta lista le añadimos los datos que requerimos para cada juguete de forma aleatoria. La creación de una descripción aleatoria por parte de Mockaroo generaba problemas con el formato a la hora de introducir los datos en MySQL, por lo que nos limitamos a inventarnos unas descripciones. Tras generar la lista de

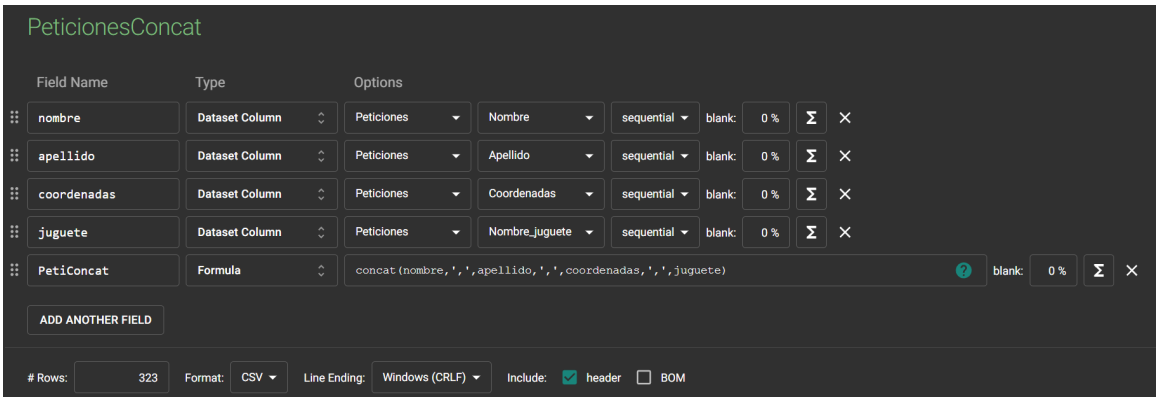
juguetes nos dimos cuenta de que el nombre de juguete 'Barbie' estaba repetido en nuestra lista de nombres inicial (números 37 y 51 que aparecen en la dirección web), por lo que simplemente eliminamos, sin criterio alguno, una de las entradas que compartían esa información.



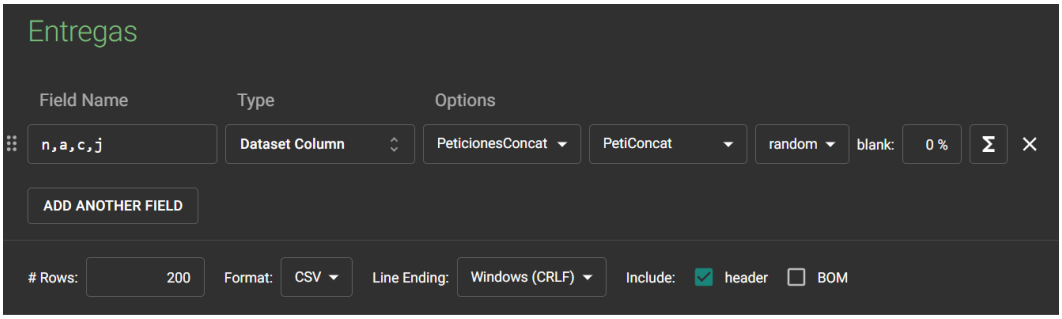
En este momento creamos otra base de datos (neto) que incluye a los niños sin repetirse el NSS_elfo y la lista de juguetes.

Para crear las peticiones cumpliendo el hecho de que a fecha de la entrega (que hemos supuesto que será el 25-12-2022) la edad del niño (suponiendo que si tiene más de X años y medio será considerado como que tiene X+1 años) está comprendida entre las edades mínima y máxima del juguete que quiere pedir usamos el archivo Peticiones_Familiares.py, el cual genera para cada niño una lista de entre 0 y 4 juguetes aptos para este, que conformarán sus peticiones.

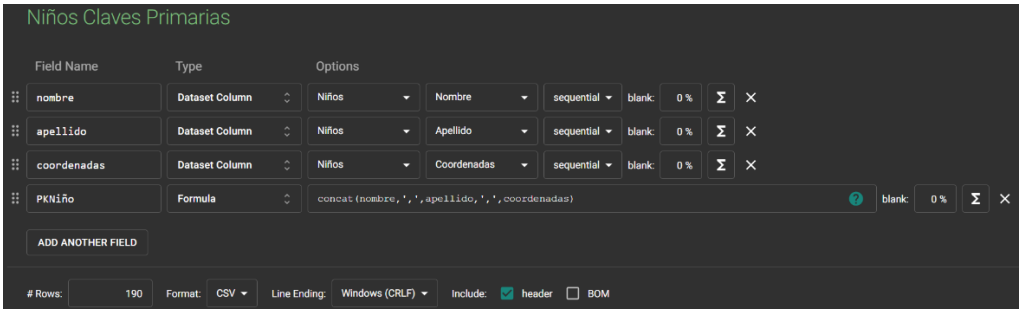
Para crear las entregas, añadimos a Mockaroo como *dataset* las peticiones obtenidas de la ejecución del programa. Necesitamos agrupar todos los campos en uno solo para que a la hora de generar las entregas considere cada fila como un todo. Para ello copiamos secuencialmente todos los datos de las columnas de la base de datos añadida a Mockaroo Peticiones y creamos una nueva columna que concatene los valores de cada columna separados por comas.



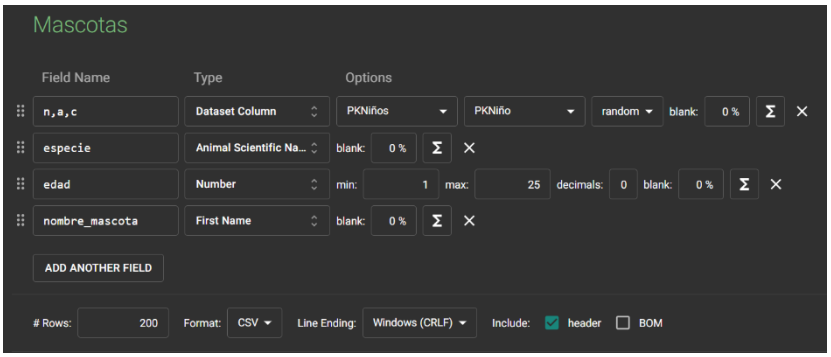
Guardamos esto como otro *dataset* en Mockaroo. Tras esto, escogemos aleatoriamente algunas de ellas (menos que el número de peticiones) como peticiones de los niños que sí han sido satisfechas, es decir, entregadas.



La creación de mascotas es simple utilizando una base de datos con las claves primarias de los niños, que generamos de la misma manera que hicimos con las peticiones, recordando en este caso que necesitamos usar como *dataset* la que corresponde con el archivo de los niños generado con el programa de Python.



Para generar los datos de las mascotas usamos de nuevo las claves primarias de los niños.



Para generar los familiares haremos uso del archivo Peticiones_Familiares.py, que empareja a todos los niños que viven juntos, suponiendo que eso implica que han de tener algún tipo de parentesco, sin repetir ningún par, lo cual será necesario para no producir incongruencias.

Esta lista de parejas de convivientes la subimos a Mockaroo como *dataset* y, tras realizar la concatenación de sus columnas, le añadimos aleatoriamente el tipo de parentesco. Como bien comentábamos previamente, si hubiera algún par repetido, con el orden invertido, podríamos tener que dos personas son hermanas y hermanastras a la vez, lo cual no tiene sentido.