const arrayEpisodes = (RMDataResults) => {

let arrayEpisodes = []; //Array vacío donde guardaremos los Episodios

for (let i=0; i < RMDataResults.length; i++) {//recorremos el arreglo de personajes

if ( arrayEpisodes.length === 0){ //si el arreglo de personajes ya está vacío

arrayEpisodes.push(RMDataResults[i].episode[0]); //guarda el primer episode de ese personajes en el arreglo de episodios

} //si ponemos un else, no recorrerá todos los episodios del primer personaje

if (arrayEpisodes.length !== 0) { //si el arreglo de episodios no está vacío

for (let x=0; x < RMDataResults[i].episode.length; x++) { //recorremos el arreglo de episodios del personaje

let count = 0; //contador que nos ayudará a reconocer cuando guardar el episodio en el arreglo de episodios sin que se repitan.

for (let z=0; z < arrayEpisodes.length; z++) { //recorremos el arreglo de episodios

if (!(RMDataResults[i].episode [x].localeCompare(arrayEpisodes[z]))) {//si el episodio del personaje es igual a un episodio del arreglo de episodios

continue; //continuar

} else {

count++; //si no, aumenta el contador

if (count == arrayEpisodes.length) { //si el contador es igual al largo del arreglo de episodio, significa que el episodio del personaje no se ha guardado en el arreglo de episodios, por lo tanto, lo tenemos que guardar

arrayEpisodes.push(RMDataResults[i].episode[x]); //guarda el episodio de ese personaje en el arreglo de episodios

}

}

}

}

}

}

return arrayEpisodes;

}