

Para realizar los ejercicios, no basaremos en el siguiente fichero

<https://gist.github.com/DavidBernalGonzalez/c3514686f695f5a13b8ce4c0d0aa67c3>

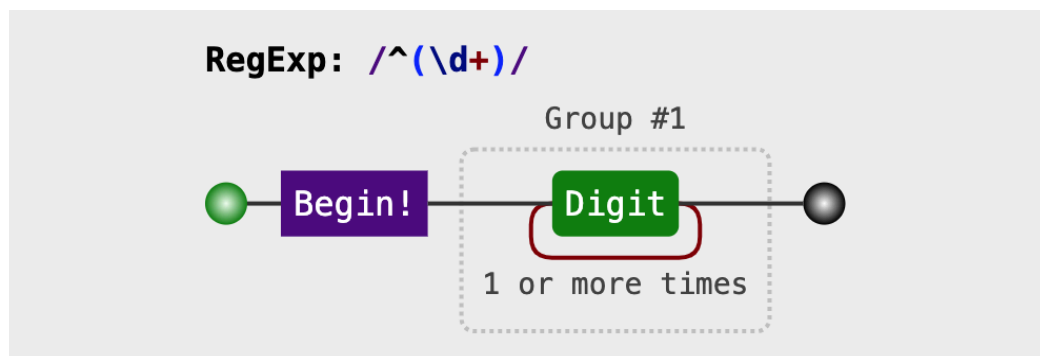
```
file.txt
1 1 ABARCA INGRID iabarcae@yahoo.es 70s
2 2 ABARCA MARITZA maeillanes@hotmail.com 70s
3 3 ABARCA OSVALDO osabarca@hotmail.com 70s
4 4 ABRIGO CAMILA cabrigo@garmendia.cl
5 5 ABRIGO IGNACIO Sb.nashxo.sk8@hotmail.com Beneficiario Ruta
6 6 ABUMOHOR FRANCISCA fran.afull@live.cl
7 7 AGUILERA CARLOS carlosaguileram@hotmail.com
```

Ejercicio 1: Basándote en el fichero original, añade un “)” inmediatamente después del primer número del resultado debe ser el siguiente:

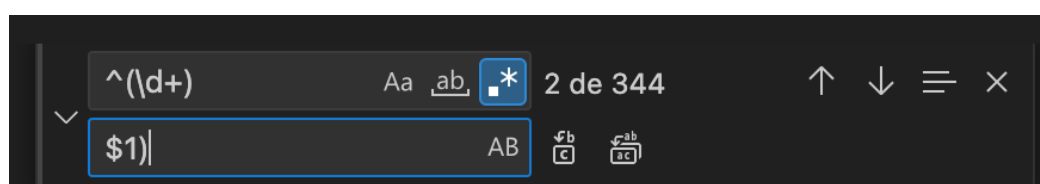
```
file.txt
1 1) ABARCA INGRID iabarcae@yahoo.es 70s
2 2) ABARCA MARITZA maeillanes@hotmail.com 70s
3 3) ABARCA OSVALDO osabarca@hotmail.com 70s
4 4) ABRIGO CAMILA cabrigo@garmendia.cl
5 5) ABRIGO IGNACIO Sb.nashxo.sk8@hotmail.com Beneficiario Ruta
```

□ Explica aquí como lo has logrado, poniendo el regex, una captura de pantalla, etc.

Con `(\d+)` le decimos que encuentre un dígito o más, y si le añadimos el `^` al principio indicamos que tiene que empezar con un número.



Después le decimos que del match, coja el valor del primer bloque con `$1` y luego le ponemos el `)` que queremos añadir, y así por cada uno, cogerá el número que ya tiene y le añadirá un `)` después

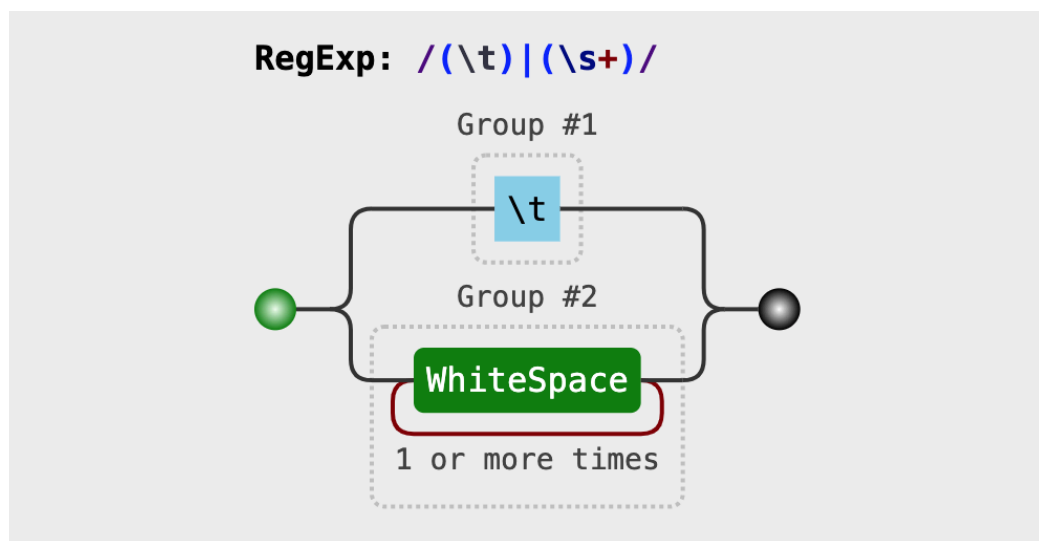


Ejercicio 2: Basándote en el fichero original, haz que todos los elementos tengan un solo espacio entre palabra y palabra. Por tanto, es necesario quitar tabulaciones y doble espaciados. Además, también debes de seleccionar los espacios del final independientemente de si son tabulaciones o espacios.

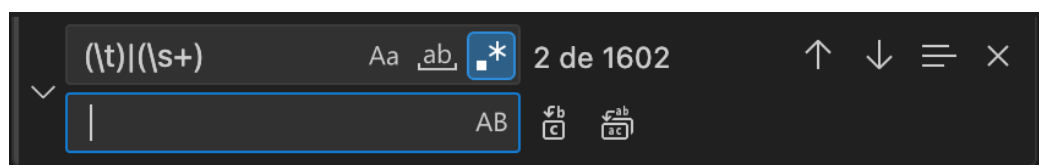
```
1 1> ABARCA INGRID -> iabarcae@yahoo.es -> 70s
2 2> ABARCA MARITZA -> maeillanes@hotmail.com -> 70s
3 3> ABARCA OSVALDO -> osabarca@hotmail.com -> 70s
4 4> ABRIGO CAMILA -> cabrigo@garmendia.cl -> 
5 5> ABRIGO IGNACIO -> Sb.nashxo.sk8@hotmail.com -> Beneficiario Ruta
6 6> ABUMOHOR FRANCISCA -> fran.afull@live.cl -> 
7 7> AGUILERA CARLOS -> carlosaguileram@hotmail.com -> 
8 8> AGUILERA CATALINA -> ikis_rojo@hotmail.com -> 
9 9> AGUILERA DANIELA -> daniela_aguilera_m500@hotmail.com -> 
10 10> AGUILERA FRANCISCA -> vizkala@hotmail.com -> 
11 11> AGUILERA FRANCISCO -> alexus3@hotmail.com -> 
12 12> AGUILERA LUIS -> capitanaguilera@hotmail.com ->
```

□ Explica aquí como lo has logrado, poniendo el reggex, una captura de pantalla, etc.

Con `\t` indicamos que busque las tabulaciones, el `|` lo utilizamos para decirle que también tiene que buscar otro match, que serán los espacios utilizando `\s+`.

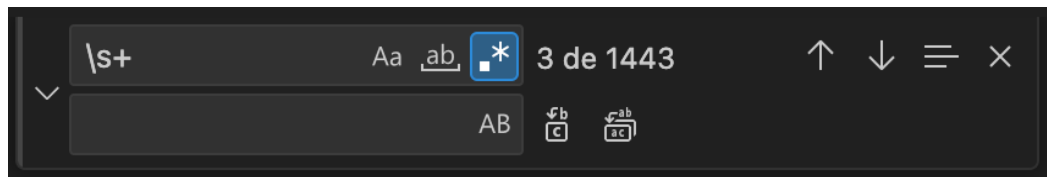


Después le decimos que lo sustituyamos por un espacio en blanco.



Ejercicio apartado B (Ampliación): Una vez lo tengas resuelto intenta hacerlo en los menos pasos posibles.

Utilizando solamente `\s+` nos hará lo mismo.



`\s+$`

Ejercicio 3: Basándote en el fichero original, sustituye todos los subdominios de los correos que acaben en `cl` o `ch` por `gmail` (incluidos los que ya tengan dicha casuística):

	Subdominio:	Y acaban en
@	Lo que sea (adretail, canal, uvavins, gmail...)	cl o ch

Por ejemplo:

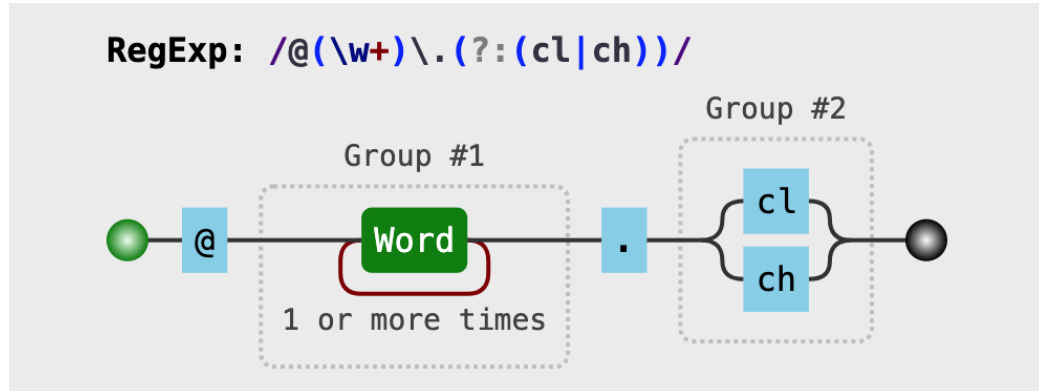
```
35 ASENJO MARCELO masenjog@gmail.com 70s
36 ASPEE SERGIO Sergio.Aspe@adretail.cl 70s
37 ASPEE CONSUELO caspe@canal13.cl 70s
38 BANTO LUIS bantomai@gmail.com 70s
39 BANTO MARCELO mfbanto@gmail.com 70s
40 BANTO RODRIGO Rodrigo.banto@Uvavins.ch 70s
41 BARRERA JORGE jbarrera05@hotmail.com 70s
```

El resultado será el siguiente:

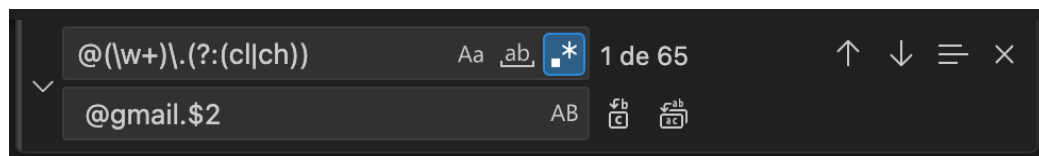
```
35 ASENJO MARCELO masenjog@gmail.com 70s
36 ASPEE SERGIO Sergio.Aspe@gmail.cl 70s
37 ASPEE CONSUELO caspe@gmail.cl 70s
38 BANTO LUIS bantomai@gmail.com 70s
39 BANTO MARCELO mfbanto@gmail.com 70s
40 BANTO RODRIGO Rodrigo.banto@gmail.ch 70s
41 BARRERA JORGE jbarrera05@hotmail.com 70s
```

□ Explica aquí como lo has logrado, poniendo el reggex, una captura de pantalla, etc.

Como todos los subdominios se encuentran detras de un @, le indicamos que tiene que hacer match con @ el subdomicio con \w+ luego . y por ultimo que termine en cl o ch con (cl | ch).



Por último le decimos que el match lo sustituya por “@gmail.” y le añadiremos el valor del grupo 2 con \$2, que es donde tenemos guardado si es cl o ch.



En clase:

`\w+(\.cl\|.ch)`

Ejercicio 3 apartado B (Ampliación): Una vez lo tengas resuelto intenta hacerlo en los menos pasos posibles.

Ejercicio 4: Elimina todo lo que no sea un email de cada una de las líneas:

```
file.txt
1 1 ABARCA INGRID iabarcae@yahoo.es 70s
2 2 ABARCA MARITZA maeillanes@hotmail.com 70s
3 3 ABARCA OSVALDO osabarca@hotmail.com 70s
4 4 ABRIGO CAMILA cabrigo@garmendia.cl
5 5 ABRIGO IGNACIO Sb.nashxo.sk8@hotmail.com Beneficiario Ruta
6 6 ABUMOHOR FRANCISCA fran.afull@live.cl
7 7 AGUILERA CARLOS carlosaguileram@hotmail.com
8 8 AGUILERA CATALINA ikis_rojo@hotmail.com
9 9 AGUILERA DANIELA daniela_aguilera_m500@hotmail.com
```

El resultado final debe de ser el siguiente:

```
file.txt
1 iabarcae@yahoo.es
2 maeillanes@hotmail.com
3 osabarca@hotmail.com
4 cabrigo@garmendia.cl
5 Sb.nashxo.sk8@hotmail.com
6 fran.afull@live.cl
7 carlosaguileram@hotmail.com
8 ikis_rojo@hotmail.com
9 daniela_aguilera_m500@hotmail.com
10 vizkala@hotmail.com
11 alexus3@hotmail.com
12 carlosaguileram@hotmail.com
```

□ Explica aquí como lo has logrado, poniendo el reggex, una captura de pantalla, etc.

Primero de todo le indicamos que coja el correo electrónico que se compone de 5 partes. `(\w+)` para el nombre del correo, el `@` seguido del subdominio, el punto `.` y la terminación `(\w+)`.

Seguidamente en la sustitución le decimos que inserte un salto de línea con `\n`, después vuelva a escribir el valor que hemos capturado anteriormente con `$&` y por último otro salto de línea con `\n` para las que no terminan únicamente con un correo electrónico.

```
(\w+)@(\w+)\.(\w+)
\n$&\n
```

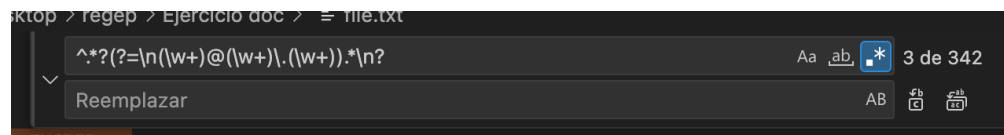
Una vez dividido el documento en diferentes columnas, nos quedarán los correos aislados en la línea de los otros textos.

Ahora en el `match` le diremos que las líneas que empiecen y acaben con un correo electrónico las tiene que ignorar.

Para esto, le indicaremos el comienzo de línea con `^` seguido de `.` para que coincida con el prompt y `*` para que sea una o más veces, y por último con el `?` le indicamos que no sea codicioso.

Seguidamente lo meteremos dentro de un `(?=())` para que lo que coincida no lo marque, y lo que nos coincide son los correos electrónicos que comienzan en una nueva línea.

Todo le diremos que lo sustituya por *nada*, y le daremos *dos veces* para que así se aplique el resultado de la manera que queremos.

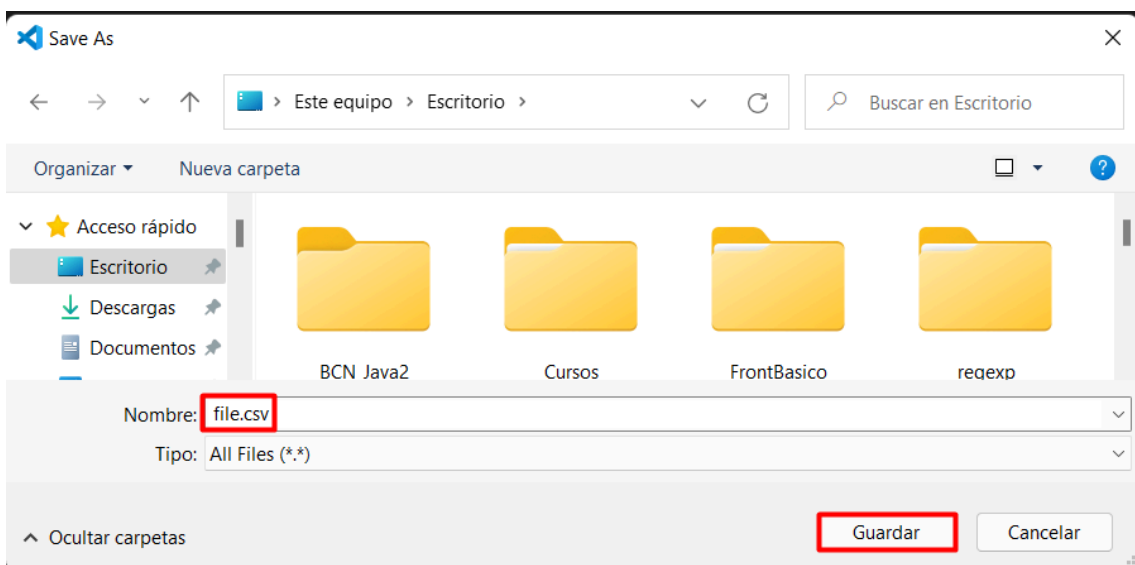
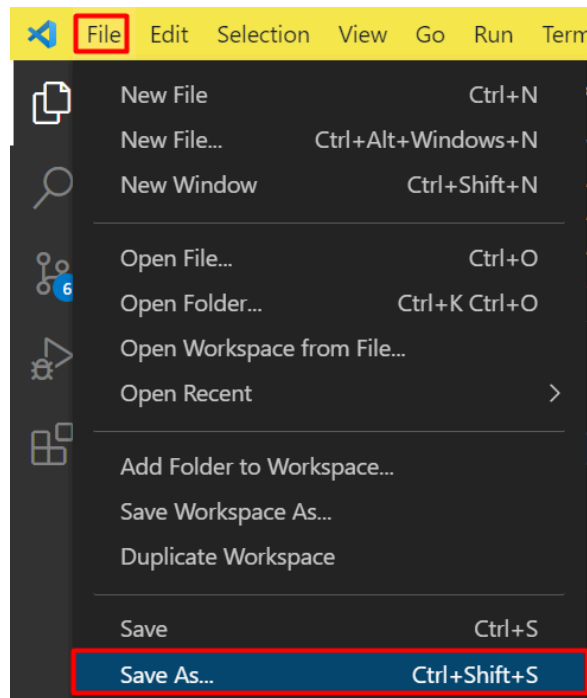


Esta es la única forma que he conseguido sacar los correos únicamente. Pues intente hacer negaciones, pero por lo que he visto no se pueden con regex sin utilizar plugins desde VSC. El resto de formas también terminaba necesitando hacerlo en más de una línea diferente para que lo maquetara de la forma correcta.

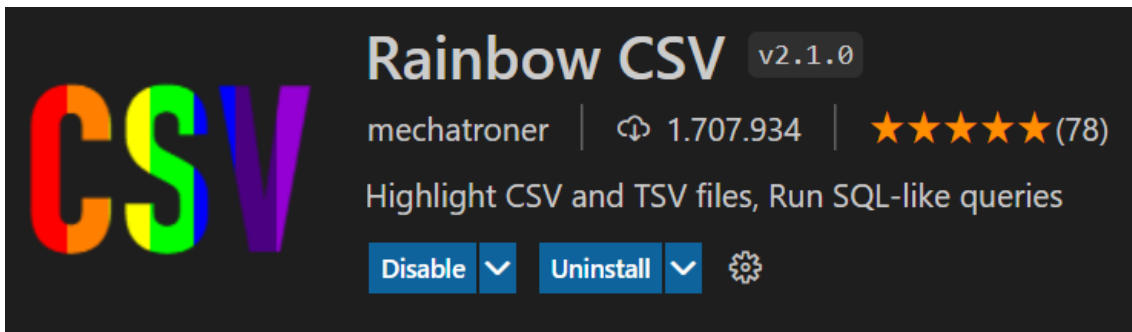
Clase: `.*(\b[\w\._-])+@\w+\.\w+.*`

Ejercicio 5: la compañía nos comenta que necesitamos pasar el fichero a un formato CSV. Ya que quieren poder abrir el fichero desde Excel por lo que tendremos que transformar el fichero a un formato CSV. Para ello, nos han pedido que no tenga espacios y que los elementos estén separados por puntos y comas de una forma similar a la siguiente:

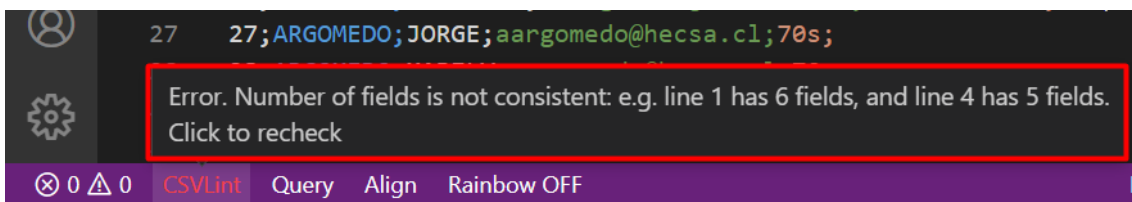
Finalmente, y para testear que el fichero esté correctamente creado, renombrarlo a una extensión .CSV. Para ello, realizamos lo siguiente:



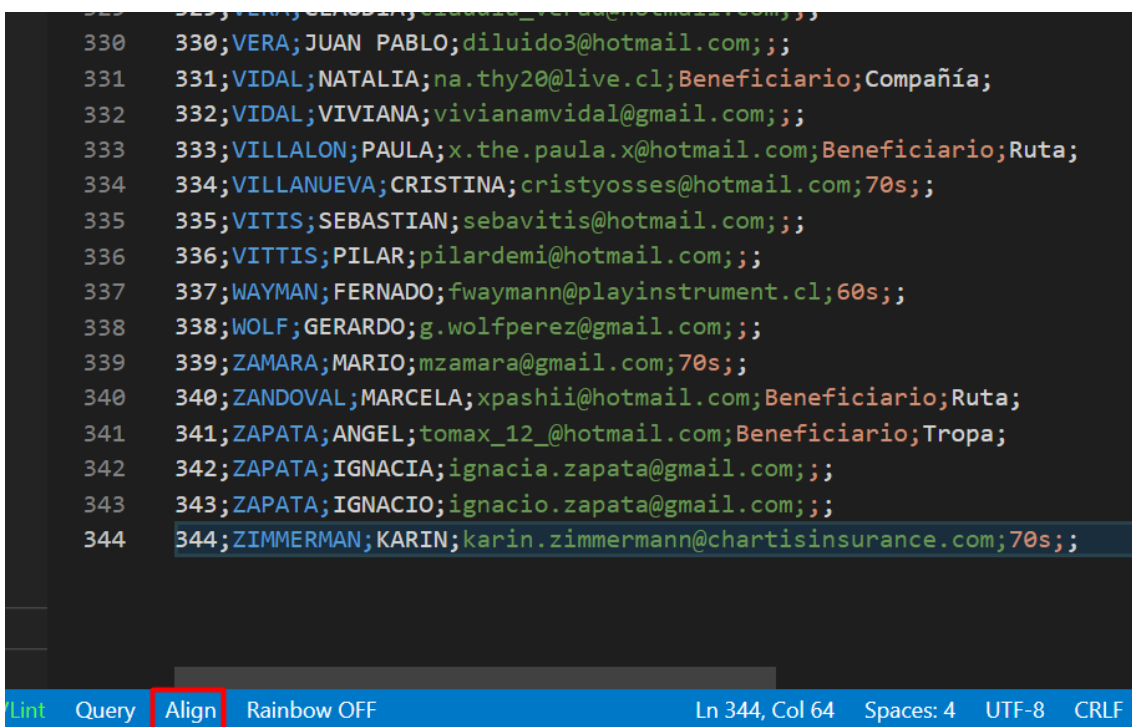
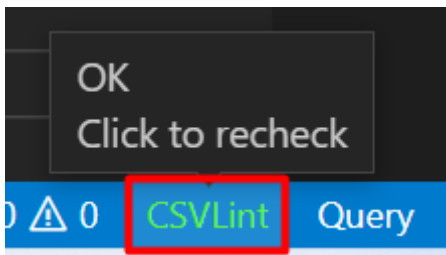
Y tras guardarlo, vamos a instalar un plugin para VSC para poder visualizarlo:



Finalmente, si intentamos abrir el documento desde CSVLint vemos que nos dice que las columnas no son las mismas. Por lo que vamos a añadir



Lo que nos está pasando es que todos los registros deben de tener los mismos puntos y comas. Por los que los debes de igualar los ; hasta que nos aparezca lo siguiente

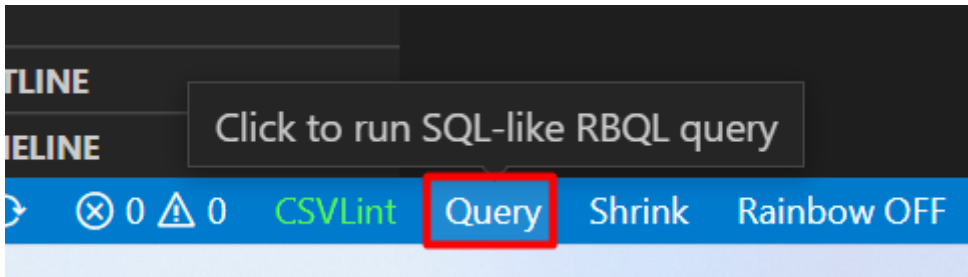


```

338 338 ;WOLF.....;GERARDO.....;g.wolfperez@gmail.com.....;.....;
339 339 ;ZAMARA.....;MARIO.....;mzamara@gmail.com.....;70s.....;
340 340 ;ZANDOVAL.....;MARCELA.....;xpashii@hotmail.com.....;Beneficiario.....;Ruta.....;
341 341 ;ZAPATA.....;ANGEL.....;tomax_12@hotmail.com.....;Beneficiario.....;Tropa.....;
342 342 ;ZAPATA.....;IGNACIA.....;ignacia.zapata@gmail.com.....;.....;.....;
343 343 ;ZAPATA.....;IGNACIO.....;ignacio.zapata@gmail.com.....;.....;.....;
344 344 ;ZIMMERMAN.....;KARIN.....;karin.zimmermann@chartisinsurance.com.....;70s.....;

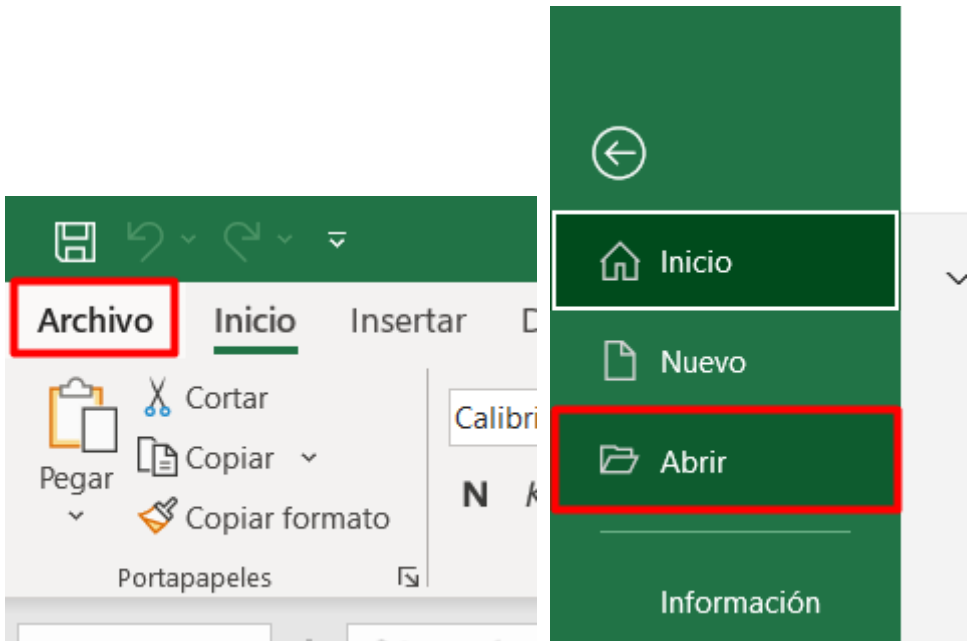
```

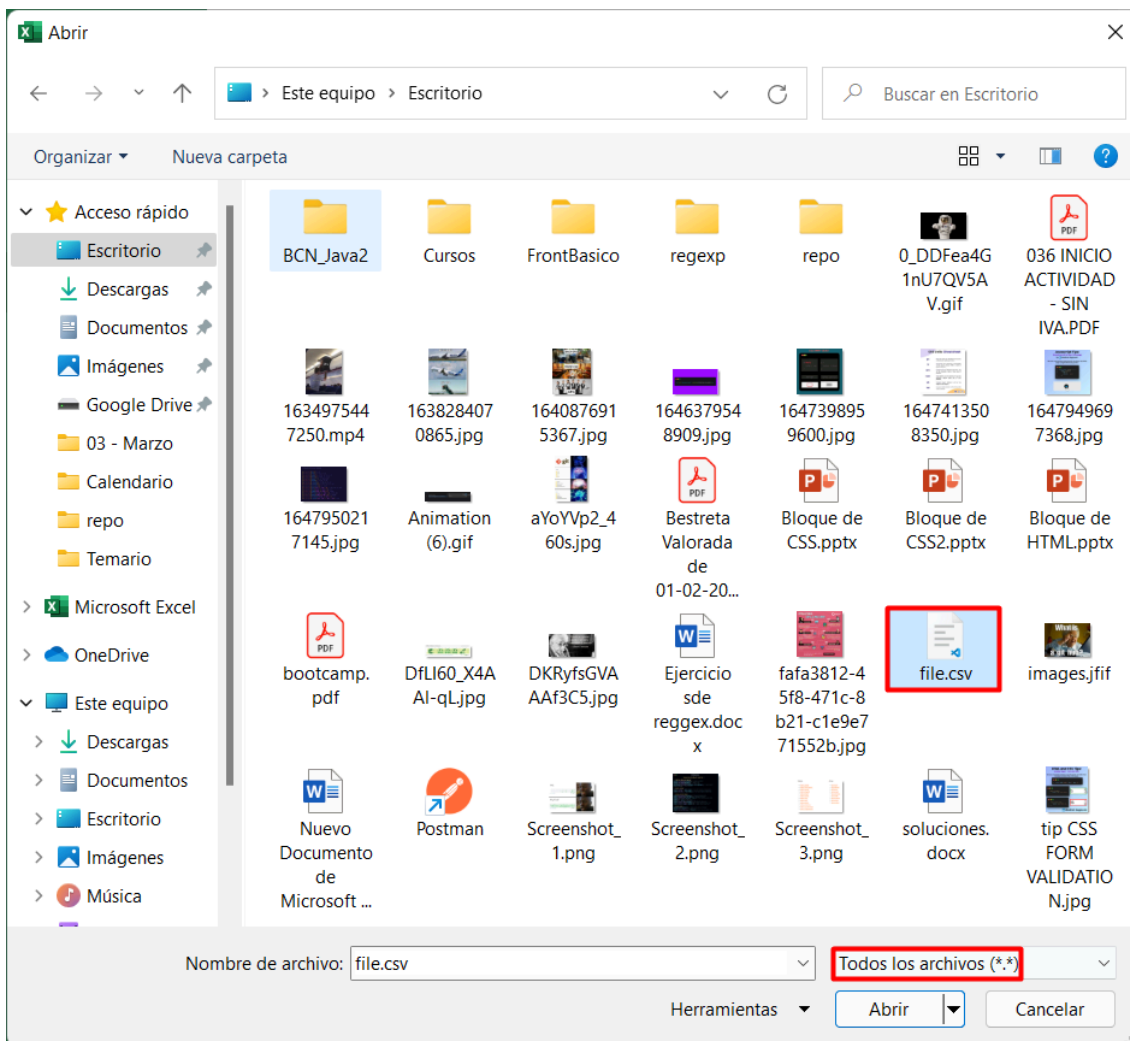
E incluso lo podemos presentar como si fuera una consulta de SQL de la siguiente manera:



NR	a1	a2	a3	a4	a5	a6
1	1	ABARCA	INGRID	iabarcae@yahoo.es	70s	
2	2	ABARCA	MARITZA	maeillanes@hotmail.com	70s	
3	3	ABARCA	OSVALDO	osabarca@hotmail.com	70s	
4	4	ABRIGO	CAMILA	cabrigo@garmendia.cl		
5	5	ABRIGO	IGNACIO	Sb.nashxo.sk8@hotmail.com	Beneficiario	Ruta
6	6	ABUMOHOR	FRANCISCA	fran.afull@live.cl		
7	7	AGUILERA	CARLOS	carlosaguileram@hotmail.com		
8	8	AGUILERA	CATALINA	ikis_rojo@hotmail.com		
9	9	AGUILERA	DANIELA	daniela_aguilera_m500@hotmail.com		
10	10	AGUILERA	FRANCISCA	vizkala@hotmail.com		

Finalmente, si abrimos el Excel y vamos a:





Podemos ver que ya lo tenemos operativo para trabajar con Excel

	A	B	C	D	E	F
1	1	ABARCA	INGRID	iabarcae@yahoo.es	70s	
2	2	ABARCA	MARITZA	maeillanes@hotmail.com	70s	
3	3	ABARCA	OSVALDO	osabarca@hotmail.com	70s	
4	4	ABRIGO	CAMILA	cabrigo@garmendia.cl		
5	5	ABRIGO	IGNACIO	Sb.nashxo.sk8@hotmail.com	Beneficiario	Ruta
6	6	ABUMOHOR	FRANCISCA	fran.afull@live.cl		
7	7	AGUILERA	CARLOS	carlosaguileram@hotmail.com		
8	8	AGUILERA	CATALINA	ikis_rojo@hotmail.com		
9	9	AGUILERA	DANIELA	daniela_aguilera_m500@hotmail.com		
10	10	AGUILERA	FRANCISCA	vizkala@hotmail.com		
11	11	AGUILERA	FRANCISCO	alexus3@hotmail.com		
12	12	AGUILERA	LUIS	capitanaguilera@hotmail.com		
13	13	ALAMOS	ANDREA	apalamosg@hotmail.com		
14	14	ALWYN	NICOLAS	niikhox_@hotmail.com	Beneficiario	Tropa
15	15	ANDRADE	LUCILA	luuuuuuci@hotmail.com	00s	
16	16	ANGULO	CRISTIAN	kristian_siempre_azul@hotmail.com	Beneficiario	Tropa