Parece igual, no lo es

`owners_min` y la caza de duplicados en `games`

Nombres _____

Superset → **SQL → SQL Lab**, base de datos `Dades classe`, esquema `public`, tabla `games`, y vuestra consulta guardada `games_step1`. Dos objetos que se ven idénticos y no se apilan: algo oculto los hace distintos. En parejas, un solo teclado; cambiad quién teclea al acabar la Parte A. Ejecutad **una sentencia cada vez**: seleccionadla y pulsad **Run**.

Referencia

Herramienta	Qué hace
`ASCII(c)`	El número de un carácter: su huella dactilar.
`SPLIT_PART(x, sep, n)`	Corta `x` en cada `sep` y devuelve el trozo `n`.
`REGEXP_REPLACE(x, p, new, 'g')`	Sustituye **todas** las coincidencias del patrón `p`. `[^0-9]` significa «cualquier cosa que no sea una cifra».
`SELECT DISTINCT *`	Conserva una copia de cada fila totalmente idéntica.
`ROW_NUMBER() OVER (PARTITION BY k ...)`	Numera las filas 1, 2, 3… dentro de cada grupo. El grupo `k` es la **clave**: las columnas que dicen «esto es la misma cosa».

Números de carácter: 32 el espacio que escribe el teclado · 44 la coma · 46 el punto · 160 el **espacio de no separación**, que se ve igual que un espacio. En una página web se escribe ` ` y significa «no partas la línea aquí».

Parte A · De un rango en texto a un número (16 min)

`owners` dice `10,000,000 .. 20,000,000`. Queréis el primer número, como número de verdad. Cortar en `' .. '` falló.

A1 · La huella dactilar. Tomad el carácter de justo antes de los puntos y pedid su número.

```
SELECT ASCII(SUBSTR(owners, POSITION('..' IN owners) - 1, 1)) AS mystery,
       ASCII(' ') AS normal_space
FROM games WHERE game = 'Half-Life 2';
```

`mystery` _____ `normal_space` _____ ¿Por qué `' .. '` nunca encajaba? _____

A2 · Cortad por donde os podéis fiar. Los dos puntos son caracteres normales. Cortad ahí y medid.

```
SELECT SPLIT_PART(owners, '..', 1) AS part,
       LENGTH(SPLIT_PART(owners, '..', 1)) AS len, LENGTH('10,000,000') AS expected
FROM games WHERE game = 'Half-Life 2';
```

`len` _____ `expected` _____ ¿De dónde sale la diferencia? _____

A3 · Esta falla a propósito. Herramientas de la semana pasada: `TRIM` el espacio, `REPLACE` las comas, convertir.

```
SELECT REPLACE(TRIM(SPLIT_PART(owners, '..', 1)), ',', '')::bigint FROM games;
```

El error dice: _____

Mirad dentro de las comillas, después de las cifras. ¿Qué queda ahí, y por qué `TRIM` no lo ha quitado? _____

A4 · Quedaos solo con las cifras. Comas, espacios de cualquier tipo, lo que venga el año que viene: todo fuera.

```
SELECT REGEXP_REPLACE(SPLIT_PART(owners, '..', 1), '[^0-9]', '', 'g')::bigint AS owners_min,
       COUNT(*) AS n
FROM games GROUP BY owners_min ORDER BY owners_min;
```

Filas _____ El menor _____ El mayor _____ Juegos con el valor menor _____

¿El valor menor significa que nadie tiene esos juegos? ¿Qué mide realmente `owners_min`?

_____

A5 · Añadidlo a la consulta limpia. Abrid `games_step1` (**SQL → Saved Queries**, clic en el nombre). En el bloque `WITH`, cambiad el `owners` a secas por la expresión de A4 `AS owners_min` y renombrad el bloque a `step2`. Comprobadlo con:

```
SELECT game, owners_min FROM step2 ORDER BY owners_min DESC LIMIT 4;
```

Guardadla como `games_step2`: **Save**, cambiad el nombre, **Save as new** (**Update** sobrescribiría `games_step1`).

El juego de arriba del todo: _____ Los tres siguientes: _____

Parte B · La caza de duplicados (18 min)

74 nombres aparecen más de una vez en `games`. Antes de borrar nada, tres casos. **Duplicado verdadero**: todas las columnas idénticas, la misma fila dos veces (un archivo cargado dos veces); eliminad las copias. **Duplicado aparente**: mismo nombre, otro desarrollador y fecha, como los tres `Dark Matter`; conservadlos todos. **Zona gris**: mismo nombre **y** desarrollador, dos fechas, como `Penumbra Overture`; una decisión, por escrito.

B1 · ¿Tiene `games` duplicados verdaderos?

```
SELECT COUNT(*) FROM games;
SELECT COUNT(*) FROM (SELECT DISTINCT * FROM games) AS t;
```

Primer recuento _____ Segundo recuento _____ ¿Duplicados verdaderos, sí o no? _____

B2 · La doble carga. `UNION ALL` pega dos resultados uno debajo del otro: pulsar Upload dos veces, sin tocar la tabla.

```
WITH twice AS (SELECT * FROM games UNION ALL SELECT * FROM games)
SELECT COUNT(*) FROM twice;
```

Filas en `twice` _____. Conservad la línea `WITH` y contad `SELECT DISTINCT * FROM twice`, envuelto como en B1.

Filas ahora _____

B3 · Lo mismo con una clave. Quedarse solo con `WHERE rn = 1` eliminaría las copias.

```
WITH twice AS (SELECT * FROM games UNION ALL SELECT * FROM games),
numbered AS (
  SELECT *, ROW_NUMBER() OVER (PARTITION BY game, release_date, developer ORDER BY number) AS rn
  FROM twice
)
SELECT rn, COUNT(*) FROM numbered GROUP BY rn ORDER BY rn;
```

Filas con `rn = 1` _____ Filas con `rn = 2` _____

B4 · La clave descuidada. «Un juego es su nombre»: usad solo `PARTITION BY game`, en la tabla **original** (`FROM games`, sin `twice`).

Filas con `rn = 1` _____ Filas que un `WHERE rn = 1` borraría _____ Un juego que perderíais: _____

B1 decía que no hay duplicados verdaderos. Entonces, ¿qué ha borrado esta limpieza? _____

Habéis acabado cuando Todos los huecos rellenos · `games_step2` está en las consultas guardadas. **¿Pareja rápida?** Construid `owners_max` con el trozo 2 del corte. En vuestro conjunto de datos abierto, ¿alguna columna es una clave de verdad?

¿Atascados?

- **`invalid input syntax for type bigint`** en A4, con una coma dentro de las comillas: falta el `'g'` en `REGEXP_REPLACE`.
- **El resultado dice `1k rows`**: es el límite de visualización. Envolvedla en `SELECT COUNT(*) FROM (...) AS t`.
- **`games_step1` no está en Saved Queries**: pedid su SQL al profesor.

En casa En vuestro conjunto de datos abierto, ¿qué columnas juntas identifican una fila? Demostradlo con un `GROUP BY ... HAVING COUNT(*) > 1` que no devuelva nada.