Una fecha de verdad

Texto disfrazado: `release_date` pasa por el horno Nombres _____

Superset → **SQL → SQL Lab**, base de datos `Dades classe`, esquema `public`, tabla `games`. El horno convierte el mineral (texto) en lingotes (valores con tipo). En parejas, uno teclea y el otro escribe; cambiad en la Tarea 4. Ejecutad **una sentencia cada vez**: seleccionadla y pulsad **Run**. Pedid al profesor el SQL largo de las Tareas 3 y 4 en formato texto y pegadlo: no lo tecleéis de nuevo.

Referencia

Herramienta	Qué hace
`CAST(x AS type)` `x::type`	El horno: convierte `x` en ese tipo, o se detiene con un error.
`TRIM(x)`	Corta los espacios de los dos extremos.
`REPLACE(x, a, b)`	Cambia cada `a` por `b`.
`TO_DATE(x, pattern)`	Lee una fecha escrita como dice el patrón: `Mon` mes de tres letras, `DD` día, `MM` número del mes, `YYYY` año.

Pieza del patrón	Significa
`x ~ 'pattern'`	¿Encaja el texto `x`?
`^ … $`	Desde el principio hasta el final.
`[A-Z][a-z]{2}`	Una mayúscula + dos minúsculas: `Nov`.
`[0-9]{1,2}`	Una o dos cifras (el día).
`[0-9]{4}`	Exactamente cuatro cifras (el año).

Tarea 1 · Calentad el horno

Primero escribid vuestra predicción. Después ejecutad la línea.

Línea	Vuestra predicción	Qué devuelve
`SELECT ' Valve ' = 'Valve', TRIM(' Valve ') = 'Valve';`		
`SELECT REPLACE('1,234.50', ',', '')::numeric;`		
`SELECT REPLACE(REPLACE('1.234,50', '.', ''), ',', '.')::numeric;`		

`1,234.50` (inglés) y `1.234,50` (español) son el mismo número. ¿Por qué el español necesita dos `REPLACE`? _____

Tarea 2 · Probad la manera perezosa

Las dos líneas fallan **a propósito**: el error es la lección.

```
SELECT release_date::date FROM games;
SELECT TO_DATE(release_date, 'Mon DD, YYYY') FROM games;
```

Primer error: _____

Segundo error: _____

¿Qué tipo de valor hace fallar las dos líneas? _____

Tarea 3 · Contad las formas

`CASE` es una bifurcación en el camino: cada fila toma el primer camino cuya condición es verdadera.

```
SELECT CASE WHEN release_date ~ '^[A-Z][a-z]{2} [0-9]{1,2}, [0-9]{4}$' THEN 'full'
            WHEN release_date ~ '^[A-Z][a-z]{2} [0-9]{4}$'             THEN 'month only'
            ELSE 'other' END AS shape, COUNT(*) AS n
FROM games GROUP BY shape ORDER BY n DESC;
```

`full` _____ `month only` _____ `other` _____ Mirad las filas de `other` (un `WHERE` con los mismos patrones, o el `NOT LIKE` de la semana pasada). Su valor: _____ ¿Qué tiene de raro? _____

Tarea 4 · Construid el convertidor

La misma bifurcación, pero ahora cada camino funde su propia forma. No hay `ELSE`: lo que no encaja en ningún camino se convierte en `NULL`. `WITH step1 AS (...)` da un **nombre** a una consulta, y el `SELECT` de abajo puede usar su resultado como una tabla. `numeric(8,2)` es un decimal exacto con dos decimales: el tipo adecuado para el dinero.

```
WITH step1 AS (
  SELECT number, TRIM(game) AS game,
    CASE WHEN release_date ~ '^[A-Z][a-z]{2} [0-9]{1,2}, [0-9]{4}$'
           THEN TO_DATE(release_date, 'Mon DD, YYYY')
         WHEN release_date ~ '^[A-Z][a-z]{2} [0-9]{4}$'
           THEN TO_DATE(release_date, 'Mon YYYY')
    END AS release_date,
    price::numeric(8,2) AS price,
    owners, developer, publisher, average_playtime, median_playtime, metascore
  FROM games
  WHERE game IS NOT NULL
)
SELECT COUNT(*), MIN(release_date), MAX(release_date), COUNT(*) - COUNT(release_date) AS no_date
FROM step1;
```

Filas _____ Más antigua _____ Más reciente _____ `no_date` _____

¿Qué valor se ha quedado sin fecha? ¿Accidente o decisión? _____

Ejecutad `SELECT TO_DATE('Jun 2009', 'Mon YYYY');` Devuelve _____. ¿Quién ha elegido ese día? _____

Tarea 5 · Usadlo y guardadlo

Conservad el bloque `WITH` y cambiad solo el `SELECT` final: juegos publicados por año, imposible la semana pasada.

```
SELECT EXTRACT(YEAR FROM release_date)::int AS year, COUNT(*) AS n
FROM step1 GROUP BY year ORDER BY year;
```

Año récord _____ con _____ juegos. **Save** (junto a **Run**) como `games_step1`. Sale en **SQL → Saved Queries**.

Tarea 6 · El error silencioso

Hoy es 4 de noviembre, escrito a la manera española. Predecid qué devuelve cada línea. Después ejecutadlas.

```
SELECT '04/11/2026'::date;
SELECT TO_DATE('04/11/2026', 'DD/MM/YYYY');
```

Primera línea _____ Segunda línea _____ ¿Algún mensaje de error? _____

¿Qué es peor, el error de la Tarea 2 o este resultado? ¿Por qué? _____

Habéis acabado cuando Todos los huecos rellenos · `games_step1` está en las consultas guardadas (el lunes se parte de ahí). **¿Pareja rápida?** Convertid en número o fecha una columna de texto de vuestro conjunto de datos abierto.

¿Atascados?

- **`Error parsing near ')'`** después del bloque `WITH`: habéis dejado un `;` antes del `SELECT` final. **`Error parsing near 'SELECT'`**: falta el `)` de cierre.
- **La Tarea 2 no da ningún error**: el límite de filas es inferior a 1 000. Subidlo. **Save sale en gris**: guardad el SQL en un archivo de texto.
- **`no_date` es de decenas o de miles**: un patrón está mal escrito (falta un espacio o un `$`). Pegadlo de nuevo desde el texto del profesor.

En casa Una columna `preu` contiene texto como `1.250,75`. Escribid la expresión que lo convierte en un número y explicad cada paso.