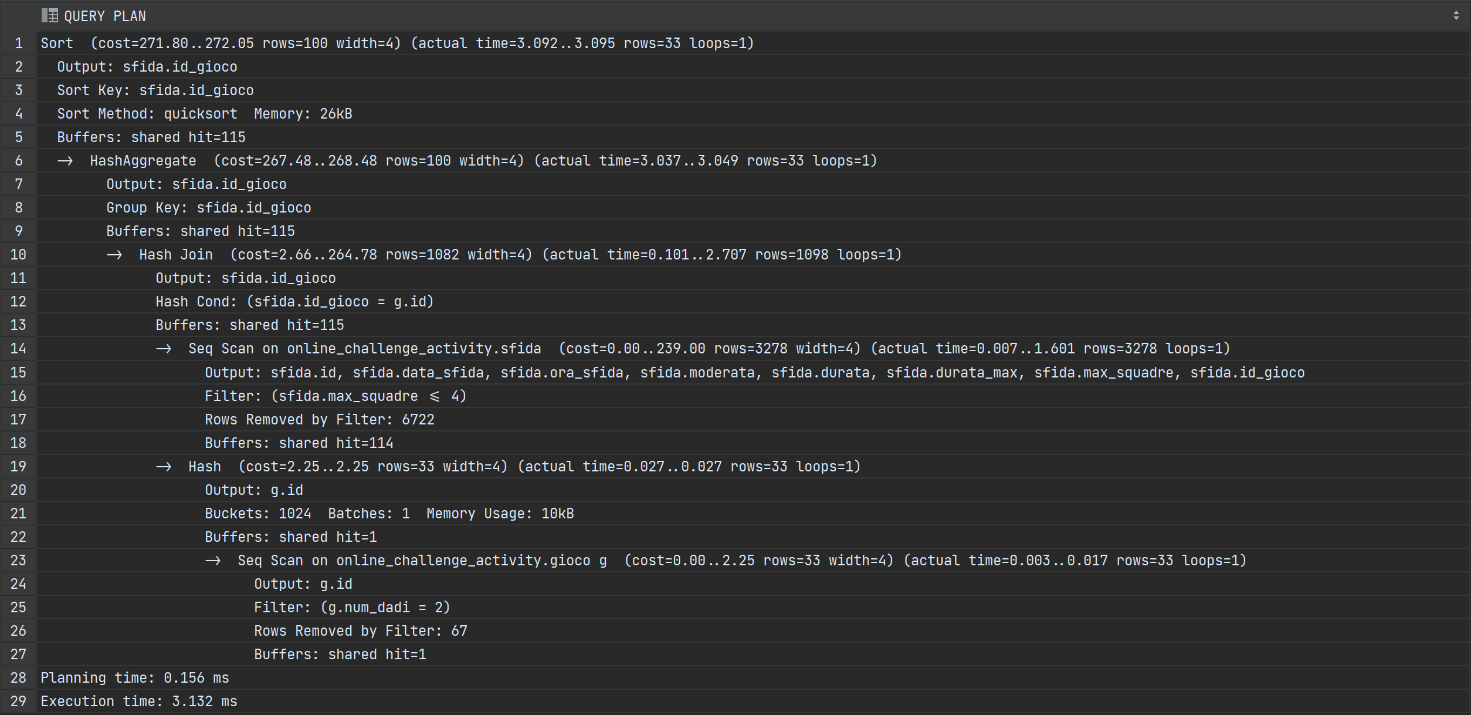
Basi di Dati A.A. 2020-2021 – Lorenzo Billi (3930391)

Progetto “Online Challenge Activity”

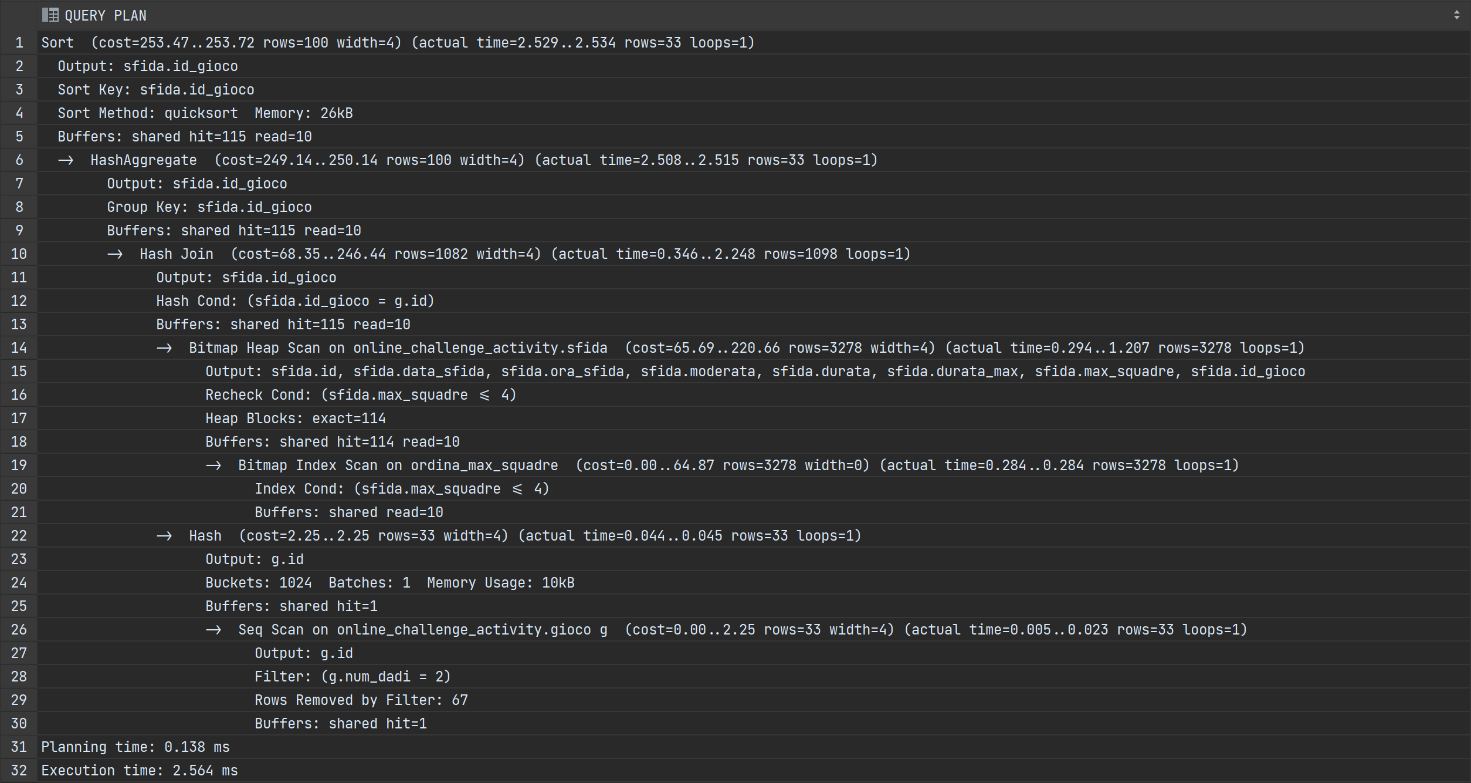
Piani di esecuzione

Piano di esecuzione per carico di lavoro 1 – senza indici



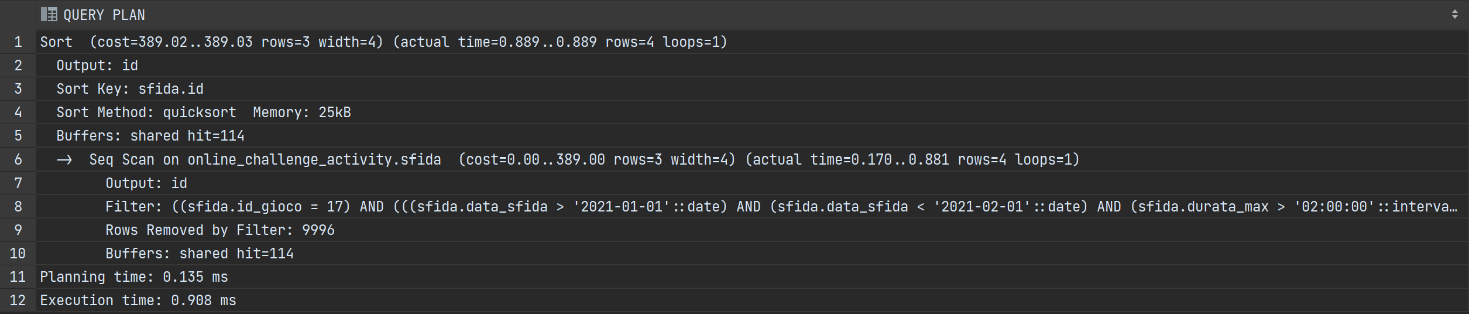
Il carico di lavoro 1 prevede un JOIN, che viene eseguito dal sistema come HASH JOIN. La selezione avviene tramite scansione sequenziale di default. Il tempo di esecuzione totale è di circa 3 millisecondi.

Piano di esecuzione per carico di lavoro 1 – con indici



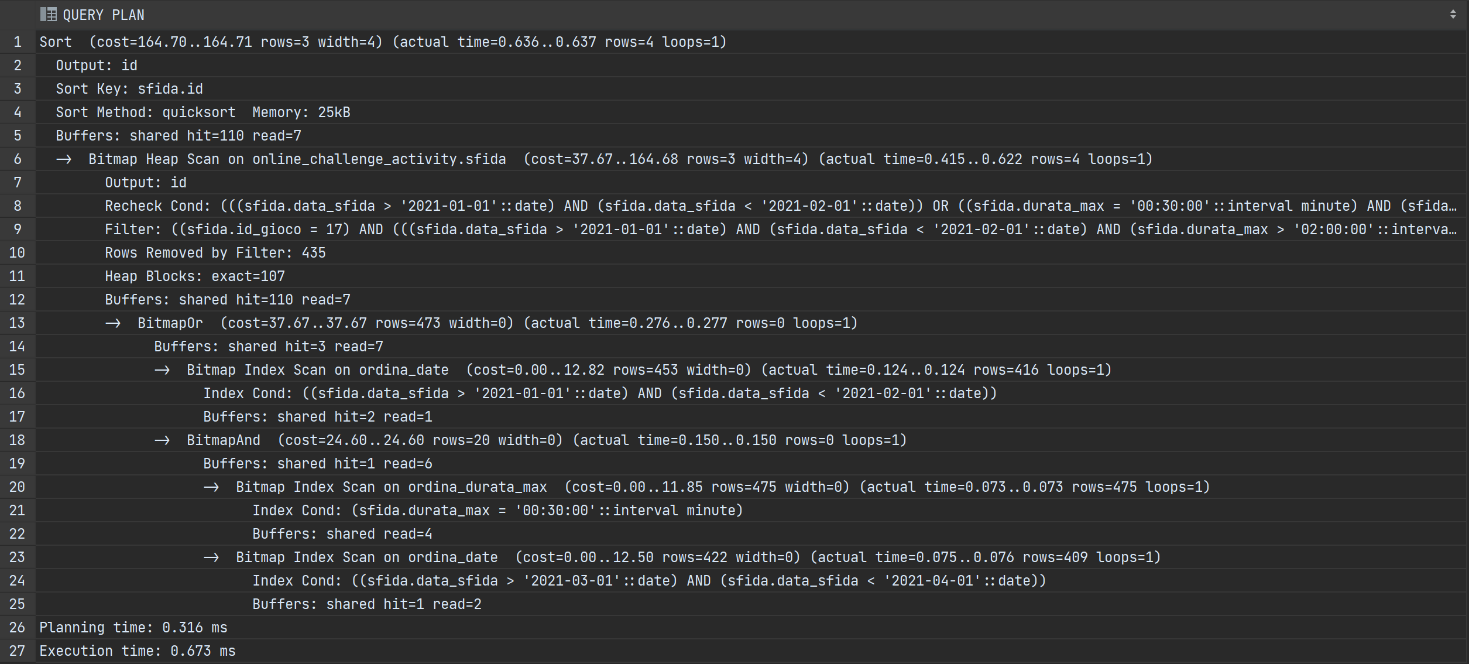
Con l’introduzione degli indici, il JOIN viene sempre eseguito come HASH JOIN, con un costo medio leggermente superiore a prima, in quanto il file è stato clusterizzato. La scansione sequenziale tuttavia è ora più efficiente rispetto a prima, proprio grazie alla clusterizzazione. Questo accorcia il tempo di esecuzione di questo carico di lavoro a poco più di 2,5 millisecondi.

Piano di esecuzione per carico di lavoro 2 – senza indici



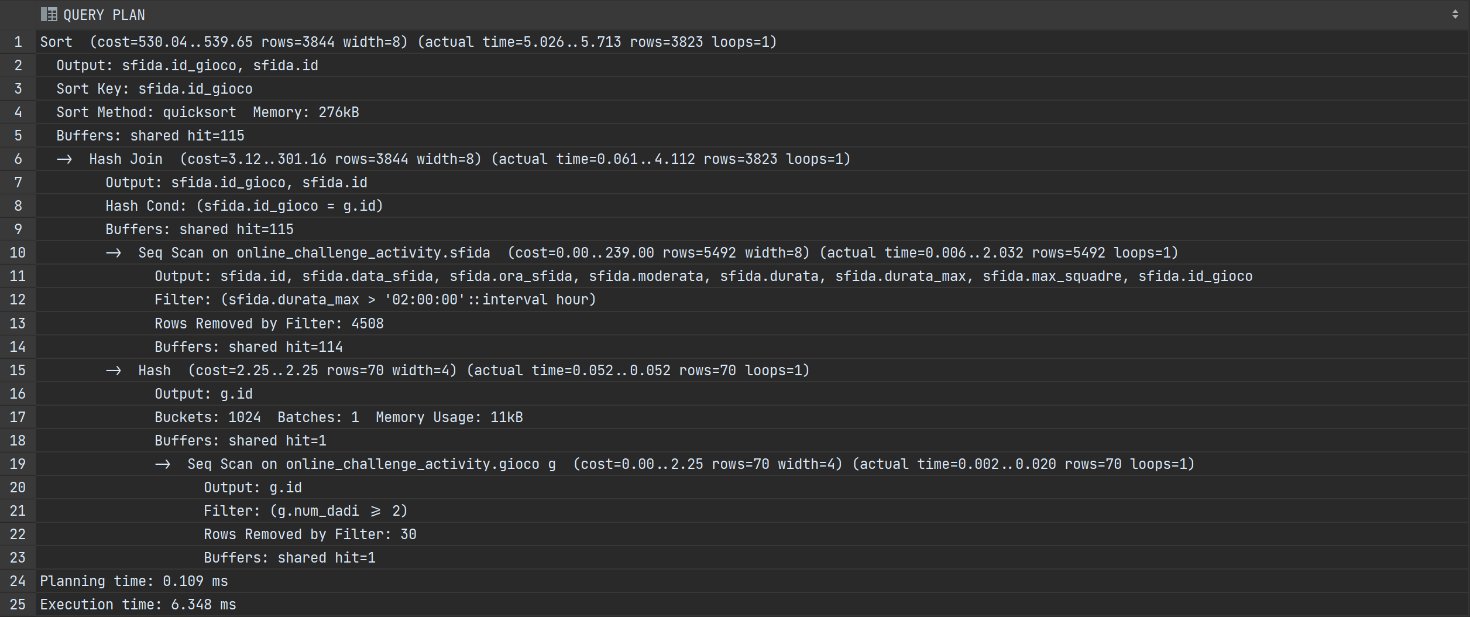
Anche in questo caso il sistema esegue una scansione sequenziale, in quanto nel secondo carico di lavoro si sta operando molto sui range di valori. Tempo di esecuzione di poco inferiore al millisecondo.

Piano di esecuzione per carico di lavoro 2 – con indici



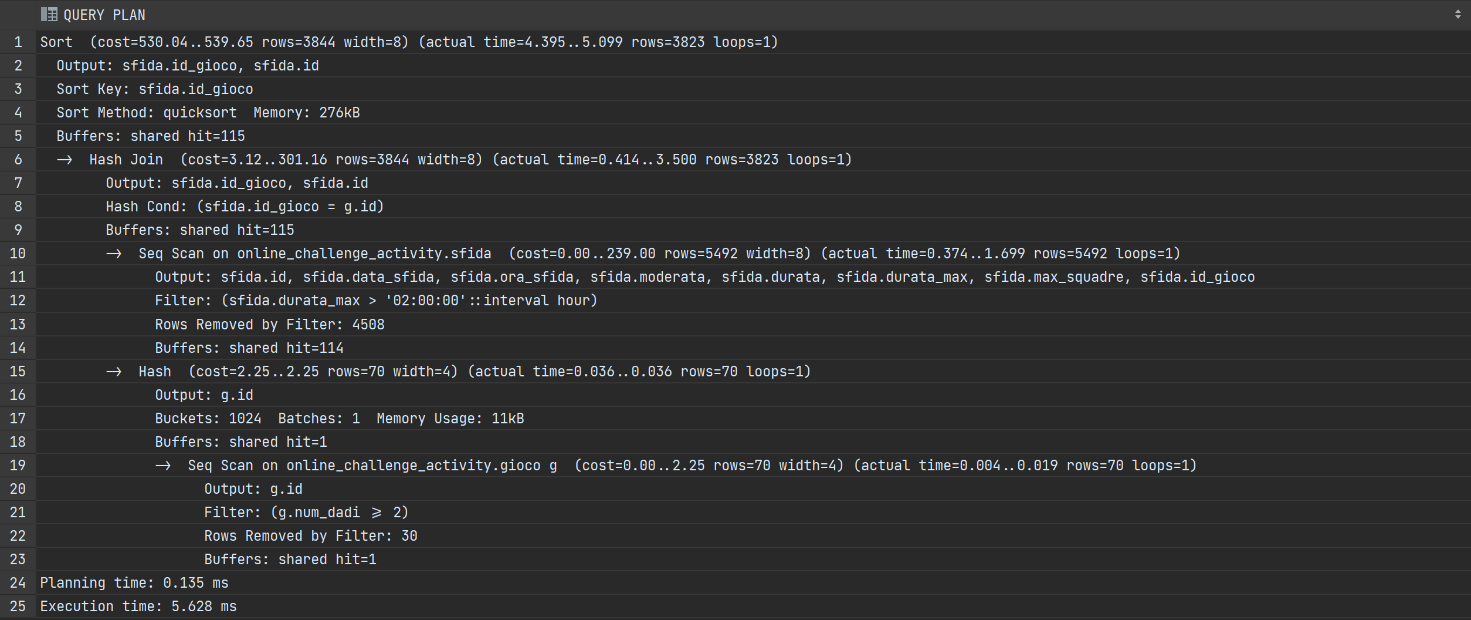
L’introduzione di un indice ordinato sulle date delle sfide riduce drasticamente il costo delle due scansioni sequenziali, e porta il tempo di esecuzione al di sotto del millisecondo (0,6).

Piano di esecuzione per carico di lavoro 3 – senza indici



Come nel caso del primo carico di lavoro, anche nel terzo carico di lavoro si ha a che fare con un JOIN, che viene implementato sempre tramite HASH JOIN. Tempo di esecuzione: 6,3 millisecondi.

Piano di esecuzione per carico di lavoro 3 – con indici



L’introduzione degli indici non ha in questo caso apportato significativi benefici in termini di costi di accesso al disco. E’ però interessante notare come il tempo di esecuzione sia comunque diminuito (5,6 millisecondi), tuttavia tale miglioramento è più da attribuire al buffer (che probabilmente ha ancora in memoria le tuple della precedente interrogazione senza indici) che agli indici in sé.