- `TRX_ISOLATION_LEVEL`

  The isolation level of the current transaction.

- `TRX_UNIQUE_CHECKS`

  Whether unique checks are turned on or off for the current transaction. For example, they might be turned off during a bulk data load.

- `TRX_FOREIGN_KEY_CHECKS`

  Whether foreign key checks are turned on or off for the current transaction. For example, they might be turned off during a bulk data load.

- `TRX_LAST_FOREIGN_KEY_ERROR`

  The detailed error message for the last foreign key error, if any; otherwise `NULL`.

- `TRX_ADAPTIVE_HASH_LATCHED`

  Whether the adaptive hash index is locked by the current transaction. When the adaptive hash index search system is partitioned, a single transaction does not lock the entire adaptive hash index. Adaptive hash index partitioning is controlled by `innodb_adaptive_hash_index_parts`, which is set to 8 by default.

- `TRX_ADAPTIVE_HASH_TIMEOUT`

  Whether to relinquish the search latch immediately for the adaptive hash index, or reserve it across calls from MySQL. When there is no adaptive hash index contention, this value remains zero and statements reserve the latch until they finish. During times of contention, it counts down to zero, and statements release the latch immediately after each row lookup. When the adaptive hash index search system is partitioned (controlled by `innodb_adaptive_hash_index_parts`), the value remains 0.

- `TRX_IS_READ_ONLY`

  A value of 1 indicates the transaction is read only.

- `TRX_AUTOCOMMIT_NON_LOCKING`

  A value of 1 indicates the transaction is a `SELECT` statement that does not use the `FOR UPDATE` or `LOCK IN SHARED MODE` clauses, and is executing with `autocommit` enabled so that the transaction contains only this one statement. When this column and `TRX_IS_READ_ONLY` are both 1, `InnoDB` optimizes the transaction to reduce the overhead associated with transactions that change table data.

- `TRX_SCHEDULE_WEIGHT`

  The transaction schedule weight assigned by the Contention-Aware Transaction Scheduling (CATS) algorithm to transactions waiting for a lock. The value is relative to the values of other transactions. A higher value has a greater weight. A value is computed only for transactions in a `LOCK WAIT` state, as reported by the `TRX_STATE` column. A NULL value is reported for transactions that are not waiting for a lock. The `TRX_SCHEDULE_WEIGHT` value is different from the `TRX_WEIGHT` value, which is computed by a different algorithm for a different purpose.

Example

```
mysql> SELECT * FROM INFORMATION_SCHEMA.INNODB_TRX\G
*************************** 1. row ***************************
                    trx_id: 1510
```