

```

a INT,
b INT,
KEY(a)
) ENGINE=NDB;
    
```

Engine condition pushdown can be used with queries such as the one shown here, which includes a comparison between a nonindexed column and a constant:

```
SELECT a, b FROM t1 WHERE b = 10;
```

The use of engine condition pushdown can be seen in the output of [EXPLAIN](#):

```

mysql> EXPLAIN SELECT a,b FROM t1 WHERE b = 10\G
***** 1. row *****
      id: 1
    select_type: SIMPLE
      table: t1
       type: ALL
possible_keys: NULL
        key: NULL
       key_len: NULL
         ref: NULL
        rows: 10
     Extra: Using where with pushed condition
    
```

However, engine condition pushdown *cannot* be used with the following query:

```
SELECT a,b FROM t1 WHERE a = 10;
```

Engine condition pushdown is not applicable here because an index exists on column [a](#). (An index access method would be more efficient and so would be chosen in preference to condition pushdown.)

Engine condition pushdown may also be employed when an indexed column is compared with a constant using a [>](#) or [<](#) operator:

```

mysql> EXPLAIN SELECT a, b FROM t1 WHERE a < 2\G
***** 1. row *****
      id: 1
    select_type: SIMPLE
      table: t1
       type: range
possible_keys: a
        key: a
       key_len: 5
         ref: NULL
        rows: 2
     Extra: Using where with pushed condition
    
```

Other supported comparisons for engine condition pushdown include the following:

- `column [NOT] LIKE pattern`
`pattern` must be a string literal containing the pattern to be matched; for syntax, see [Section 12.8.1](#), “String Comparison Functions and Operators”.
- `column IS [NOT] NULL`
- `column IN (value_list)`
 Each item in the `value_list` must be a constant, literal value.
- `column BETWEEN constant1 AND constant2`
`constant1` and `constant2` must each be a constant, literal value.