DavisBase Project: Part 1 Submission

Team Yellow

Running the program

- To run the program execute the `run` bash script using `./run` .
- Note that this will create a directory named `data/` within the `bin/` folder (if it doesn't already exist). The `bin/data/` directory contains all the `.tbl` files.

Design Assumptions

- `Date` expects the following format when inserting: `YYYY-MM-DD` .

- `Time` expects milliseconds after midnight when inserting, but will display as `hh:mm:ss` in 24-hour time or military time.

- `DateTime` expects the following format when inserting: `YYYY-MM-DD_hh:mm:ss` .

- `Year` expects the following format when inserting: `YYYY` . Note that we do not validate whether the given value falls within the

range `[1872, 2127]`, so values outside this range will experience over/underflow.

- When inserting, `INSERT INTO <table> (<column_list>) VALUES (<value_list>)`, the given `<column_list>` will match on the columns of `<table>` using the NAMES of the columns, ie: it does not matter the order of the columns passed in `<column_list>` - just the names must match the actual column names.

- When inserting, `INSERT INTO <table> (<column_list>) VALUES (<value_list>)`, the given `<value_list>` is comma delimited, and strings need not be in quotes - quotes will be dropped. This means you cannot insert the string `NULL` into a nullable text column.

Features Supported

- INSERT performs data type validation and will ABORT any invalid insertions.

- Nullable Columns: NULL may only be inserted into a nullable column, columns by default are nullable, and default value for a nullable column is NULL.

- Unique Columns: Columns by default are NOT unique. Attempting to INSERT a duplicate value into a unique column

will fail.

- MetaData Updates: Record Count in `davisbase_tables.tbl` is updated on INSERT.

- Display RowId: `SHOW ROWID;` will enable RowId to be displayed on `SELECT`;