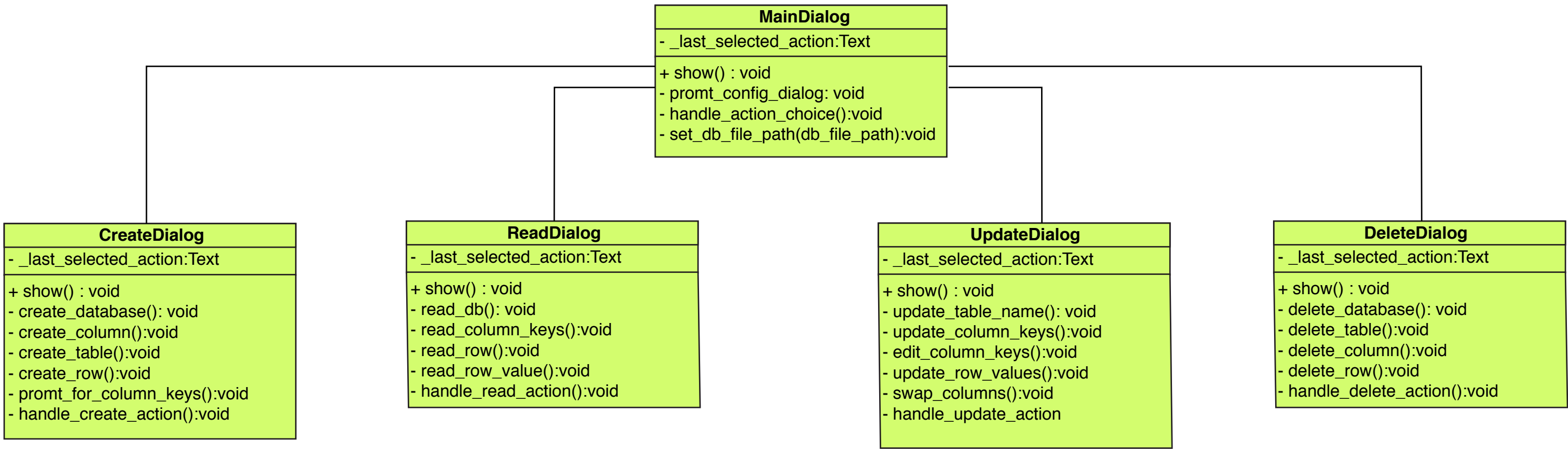


| Test.db | | | | | |
|----------------|------------|-----------|----------------|------------|-----|
| Employee table | | | | | |
| row_id | first_name | last_name | phone_number | work_title | age |
| 1 | David | Paulson | 222-333-222-11 | Consultant | 18 |
| 2 | Dick | Peterson | 2220000111 | Singer | 20 |
| 3 | John | Peterson | 222-222-222-11 | clerk | 15 |
| 4 | John | James | 222 | Janitor | 44 |



| SQLiteHelper |
|---|
| + choose_database_file() : Text + choose_file(title) : Text + choose_file_name(title, file_name) : Text + export_data(export_data) : void + create_new_database():Text + choose_table_name(file_path):Text + choose_rows(file_path, table_name):List<Text> + choose_column_keys(file_path, table_name):List + choose_column_key(file_path, table_name):Text + choose_row(file_path, table_name):int + choose_row_id(file_path, table_name):Text + choose_row_value(selected_row):Text + choose_value(db_file_path):Text + choose_row_values(selected_row):List<Text> + choose_row_value_indices(row, column_keys):List<int> + choose_row_value_index(row, column_keys):int + column_key(file_path, table_name, the_row, the_row_value):Text + row_id(the_row):Text |

| SQLiteUtil |
|---|
| + condition_procedure(conditions, alteration):Text + cap_row_values(the_text, cap_len) : Text + cap_values(the_list, cap_len) : Text + cap_items(the_list, cap_len) : void |

| SQLiteAsserter |
|--|
| + has_table(file_path, table_name):Boolean + has_column(file_path, table_name, column_name) : Boolean + has_row(file_path, table_name, conditions) : Boolean |

| SQLiteParser |
|--|
| + version_number() : Number + read_table(file_path, table_name, queries) : Text + read_row(file_path, table_name, conditions, queries) : Text + row_count(file_path, table_name) : int + table_count(file_path) : int + column_count(file_path, table_name) : int + schema(file_path, table_name) : Text + table_names(file_path) : Text + inner_join(file_path, table_a, table_b, table_a_queries, table_a_condition, table_b_query, table_b_Conditions) : Text + column_names(file_path, table_name) : Text |

| SQLiteModifier |
|---|
| + create_db(file_path) : void + create_table(file_path, table_name, column_ids) : void + insert_row(file_path, table_name, keys, row) : void + update_rows(file_path, table_name, conditions, input) : void + update_row(file_path, table_name, condition, data_set) : void + remove_table(file_path, table_name) : void + rename_table(file_path, table_name, new_table_name) : void + transfer_table(file_path, from_table, to_table, from_column_keys, to_column_keys):void + rename_columns(file_path, table_name, old_column_names, new_column_names):void + swap_columns(file_path, table_name, coloumn_key_a, column_key_b) : void + swap_column_keys(file_path, table_name, coloumn_key_a, column_key_b) : void + swap_column_data(file_path, table_name, coloumn_key_a, column_key_b) : void + remove_columns(file_path, table_name, column_names) : void + remove_row(file_path, table_name, condition) : void + add_column(file_path, table_name, column_name) : void + add_columns(file_path, table_name, column_names) : void |

| FileParser |
|---|
| + read_URL(file_URL) : Text + posix_path(the_path) : Text + file_URL(file_path) : Text + file_path(fileURL) : Text + file_name_by_url(file_URL):Text + implicit_file_URL(HFS_path) : Text + parent_folder(file_path) : Text + file_kind(file_path) : Text + file_kind_by_URL(file_URL) : Text + file_properties(file_path) : Text + file_info(the_file) : Text + file_extension(the_file) : Text + folder_names(the_folder) : List + file_names(the_folder) : List + file_name(the_file_path) :Text + file_names_sans_ext(the_folder) : List |

| FileModifier |
|--|
| + write_data(this_data, target_file, append_data) : Boolean + delete_file(file_path) : void |

| TextParser |
|--|
| + every_word(the_text) : List + every_paragraph(the_text) : List + split(the_text, delimiter) : List + text_length(the_text) : List + comma_delimited_text(text_items) : Text + delimited_text(text_items, delimiter) : Text + to_paragraphs(the_list) : Text + ordinal(the_number) : Text + encode(the_text) : Text + decode(the_text) : Text + quoted_form(the_text) : Text + sub_string(the_start, the_end, the_text) : Text |

| TextModifier |
|--|
| + replace_text(the_text, match, replacement):Text + wrap_every_text_item(text_items, wrap) : void + wrap_text(the_text, wrap) : Text |

| TextAsserter |
|---|
| + has_word(the_string, the_word) : Boolean + is_equal(a, b) : Boolean + is_text(the_text) : Boolean |

| NumberParser |
|--|
| + random_number(from_number, to_number):Number |

| ListAsserter |
|--|
| + equals_to(a, b) : Boolean + contains_List(a, b) : Boolean |

| ListParser |
|--|
| + log_items(the_list) : void + index_of(array, theItem) : int + indices(the_list, the_items) : List + clone(array) : List + items_at(the_list, indices) : List |

| ListModifier |
|--|
| + remove_range(the_list, from_index, to_index) : void + remove_at(the_list, the_index) : void + add_at(the_list, the_item, the_index) : List + replace(the_list, to_match, replacment) : List + replace_many(the_list, the_matches, the_replacments) : void + swap(the_list, a, b) : void + remove(the_list, the_item) : List + remove_many(the_Array, the_items) : List + combine(list_a, list_b, seperator) : List + bubble_sort(the_list) : void |