2021-2022
GEBZE TECHNICAL UNIVERSITY
CSE102
HW11

Name Surname : Emre Oytun
Student Number : 200104004099
Project : Database Application in C

1. Functionality of the Program :

1.1-) User can create a new database or use an old database if there is any.

1.2-) User can add a new table into the database.

1.3-) User can remove a table from the database.

1.4-) User can show tables in the database.

1.5-) User can describe a table which is in the database.

1.6-) User can insert a key into the database and make a field of the table indexed so that reaching a record from the table with that field is faster.

1.7-) User can insert a record into a table.

1.8-) User can select all records from table, and displays them.

1.9-) User can select a specific record by giving a field and value of the field.

2. Design of the Memory :

2.1-) Design of Database :

Database Struct :

```
typedef struct database {
        tables *tlist;
        int n;        /* number of tables */
        char *name; /* name of database */
} database;
```

→ Database keeps table list as linked list in ‘tlist’ pointer.
→ It keeps number of tables.
→ It keeps its name.

2.2-) Design of Tables :

Tables Struct :

```
typedef struct tables {
        struct tables *next;
        table *t;
} tables;
```

→ Tables keeps a table and the address of next tables node.

Table Struct :

```
typedef struct table {
	char *tableName;
	int field_num;

	char **field;
	char **type;
	int *isNull;
	int *isKey;

	/* Keeps the records linked list. */
	record_t *records;
	keys *keys;
} table;
```

Addition to homework description

→Table keeps table name and field number.
→ Table keeps records' lists in the variable 'records' whose type is 'record_t*' defined below the report.
→ Table keeps key lists in the variable 'keys' whose type is 'keys*' defined below the report.

2.3-) Design of Records :

Record_t Struct :

```
typedef struct record_t {
        struct record_t *next;
        data_node *recordData;
} record_t;
```

→ Records are kept in a table as linked lists of record_t.
→ record_t keeps the next record_t node/next record.
→ record_t keeps a record's data in the array 'recordData' whose type is 'data_node*' defined below.

Data_node Struct :

```
typedef struct data_node {
        int data_type;
        int charNum_ifStr;
        var_data_t data;
} data_node;
```

→ data_node keeps the type of the data in 'data_type' variable such that :
** 1-int, 2-double, 3-float, 4-char, 5-string
** It keeps char number if the data type is string in 'charNum_ifStr' variable.

→ data_node keeps the actual data in the variable 'data' whose type is 'var_data_t' defined below.

Var_data_t Union :

```
typedef union var_data_t {
        int i;
        double d;
        float f;
        char c;
        char *str;
} var_data_t;
```

→ In this union, the actual data is kept according to its type.

2.4-) Design of Keys :

Keys Struct :

```
typedef struct keys {
        struct keys *next;

        char *field;
        int data_type;
        int charNum_ifStr;

        key_node *keyList;
} keys;
```

→ Keys are kept in a table as linked list.
→ keys keeps the next keys node/next key in ‘next’ variable.
→ keys keeps the field and the data type of the field which is indexed by this key.

→ keys keeps the indexes in ‘keyList’ variable whose type is ‘key_node*’ defined below.

Key_node Struct :

```
typedef struct key_node {
        struct key_node *next;
        var_data_t keyValue;
        record_t *recordAdress;
}key_node;
```

→ key_node keeps the next key_node in the variable ‘next’.
→ key_node keeps the key value data of a record of a table in the variable ‘keyValue’.
→ key_node keeps the address of the record which the key value belongs to, in the variable ‘recordAdress’.

3-) Outputs of the Test Program :

```
emre@ubuntu:~/Desktop/hw11$ make testProgram
************** CREATE DATABASE FUNCTION TEST **************

Part case 1 Database Name : db1, The Database Name should be : db1
Part case 2 Database Name : db2, The Database Name should be : db2
Part case 3 Database Name : db3, The Database Name should be : db3
```

```
************** SHOW TABLE FUNCTION TEST **************


Case 1 Show table function result :

------ TABLES IN db1 DATABASE -------
table1

Case 1 Tables should be :
table1

Case 2 Show table function result :

------ TABLES IN db1 DATABASE -------
table1
table2

Case 2 Tables should be :
table1
table2


Case 3 Show table function result :

------ TABLES IN db2 DATABASE -------
Empty set.

Case 3 Tables Should be : Empty Set.
```

```
************** INSERT TABLE FUNCTION TEST **************

Case 1 Insert Table --- Show table result :

New table 'workers' has been inserted into the database 'db3'.
------ TABLES IN db3 DATABASE -------
workers

Case 1 Expected Tables should be :
workers

Case 2 Insert Table --- Show table result :

New table 'houses' has been inserted into the database 'db3'.
------ TABLES IN db3 DATABASE -------
workers
houses

Case 2 Expected Tables should be :
workers
houses

Case 3 Insert Table --- Insert Table Function Result =
There is a problem with the command. Be sure you enter valid command and try again...

Case 3 Insert Table --- Insert Table Function Result Should Be =
There is a problem with the command. Be sure you enter valid command and try again...

Case 3 Insert Table --- Show table result :

------ TABLES IN db3 DATABASE -------
workers
houses

Case 3 Expected Tables should be :
workers
houses
```

```
************* DESCRIBE TABLE FUNCTION TEST *************

Case 1 Describe Table 'workers' result :

TABLE NAME FOUND = workers
----------------------------------------
FIELD                                                   TYPE                                                    NULL                                                    KEY

id                                                      int                                                     YES
name                                                    char(40)                                                YES
level                                                   char                                                    YES
salary                                                  double                                                  YES

Case 1 Expected Result :
id              int                     YES
name            char(40)                YES
level           char                    YES
salary          double                  YES

Case 2 Describe Table 'houses' result :

TABLE NAME FOUND = houses
----------------------------------------
FIELD                                                   TYPE                                                    NULL                                                    KEY

address                                                 char(40)                                                YES
room_number                                             int                                                     YES
year                                                    int                                                     YES
price                                                   double                                                  YES

Case 1 Expected Result :
address                 char(40)                YES
room_number             int                     YES
year                    int                     YES
price                   double                  YES

Case 3 Describe Table 'students' result :

There is no table in the database with that name.

Case 3 Expected Result :
There is no table with that name.
```

```
************* INSERT RECORD/SELECT FROM TABLE FUNCTIONS TEST *************

Case 1 Insert New Record Below Into the Table 'workers' and Show All Records with Select From Function :
New Record : id = 1, name = Emre Oytun, level = J, salary = 10000

Enter id|int : 1
Enter name|char(40) : Emre Oytun
Enter level|char : J
Enter salary|double : 10000

Select from table 'workers' result after insertion of new record :

id|int                                                     name|char(40)                                            level|char                                               salary|double
1                                                       Emre Oytun                                              J                                                       10000.000000

Case 2 Insert New Record Below Into the Table 'workers' and Show All Records with Select From Function :
New Record : id = 2, name = Mehmet Yapici, level = S, salary = 15000

Enter id|int : 2
Enter name|char(40) : Mehmet Yapici
Enter level|char : S
Enter salary|double : 15000

Select from table 'workers' result after insertion of new record :

id|int                                                     name|char(40)                                            level|char                                               salary|double
1                                                       Emre Oytun                                              J                                                       10000.000000
2                                                       Mehmet Yapici                                           S                                                       15000.000000
```

```
************* SELECT SPECIFIC RECORD FROM TABLE FUNCTION TEST *************

Case 1 Select the record from table 'workers' whose id = 1
Expected Result = 1     Emre Oytun      J       10000

Result from function =
id|int                                                    name|char(40)                                                   level|char                                                     salary|double
1                                                       Emre Oytun                                                      J                                                       10000.000000

Case 2 Select the record from table 'workers' whose name = Mehmet Yapici
Expected Result = 2     Mehmet Yapici   S       15000

Result from function =
id|int                                                    name|char(40)                                                   level|char                                                     salary|double
2                                                       Mehmet Yapici                                                   S                                                       15000.000000

Case 3 Select the record from table 'students' whose grade = 86.120000
Expected Result = There is no table in the database with that name.

Result from function =
There is no table in the database with that name.
```

```
************* INSERT KEY FUNCTION TEST *************

Case 1 Insert Key into 'workers' table in 'id' field :
Key has been inserted into the 'id' field of 'workers' table.

Describe table after insertion of key :
TABLE NAME FOUND = workers
----------------------------------------
FIELD                                                   TYPE                                                            NULL                                                            KEY
id                                                      int                                                             NO                                                              YES
name                                                    char(40)                                                        NO
level                                                   char                                                            NO
salary                                                  double                                                          NO

Select a specific record from table (Key should be used) :
Select the record from table 'workers' whose id = 1
Result =

KEY HAS BEEN FOUND. LOOKING FOR THE RECORD IN INDEXES...
id|int                                                    name|char(40)                                                   level|char                                                     salary|double
1                                                       Emre Oytun                                                      J                                                       10000.000000

Case 2 Insert Key into 'workers' table in 'old' field :
Result =
There is no field called 'old' in the table 'workers'

Expected Result = There is no field called 'old' in the table 'workers'
```

```
************* WRITE/READ FUNCTIONS TEST *************

Writing database 'db3' into the 'database.txt' file =
Database 'db3' has been written into the 'database.txt' file
Key has been inserted into the 'id' field of 'workers' table.

Show table function result should be =
-workers
-houses

Show table result =
------ TABLES IN db3 DATABASE -------
workers
houses

Select from table 'workers' result should be =
1       Emre Oytun      J       10000
2       Mehmet Yapici   S       15000

Select from table result =
id|int                                    name|char(40)                           level|char                              salary|double
1                                       Emre Oytun                              J                                       10000.000000
2                                       Mehmet Yapici                           S                                       15000.000000
```

```
*************** REMOVE TABLE FUNCTION TEST ***************

Before Removing Table :
------ TABLES IN db3 DATABASE -------
workers
houses

----- Remove table 'workers' from database 'db3' ----

After Removing Table :
------ TABLES IN db3 DATABASE -------
houses

----- Remove table 'houses' from database 'db3' -----

After Removing Table :
------ TABLES IN db3 DATABASE -------
Empty set.
```

4-) Needs to be Developed :

4.1-) Deleting a record
4.2-) Inserting a record by entering not only data of all fields as in the program, but entering data of some fields.
4.3-) Selecting from table by field so to display records belongs to that field.
4.4-) Exception Handling in insert record function