



## **AMIT-C Programming Project**

Create STD.h which have the following data type

- typedef unsigned char uint8
- typedef char Int8
- typedef unsigned short uint16
- typedef short int16
- typedef unsigned int uint32
- typedef int int32

Create SDB.h to have the function declaration that's you use in the project and the following new datatype

```
typedef struct SimpleDb
```

```
{
```

```
    uint32 Student_ID;
```

```
    uint32 Student_year;
```

```
    uint32 Course1_ID;
```

```
    uint32 Course1_grade;
```

```
    uint32 Course2_ID;
```

```
    uint32 Course2_grade;
```

```
    uint32 Course3_ID;
```

```
    uint32 Course3_grade;
```



```
}student;
```

Note. Create your database with maximum 10 students and minimum 3 students

Create SDB.c which have the following functions

- `bool SDB_IsFull ()`
  - \*Description: check if the database is full or not.
  - \*Return: if the database is full return true else return false
- `uint8 SDB_GetUsedSize()`
  - \* Description: get how many students in your database have added
  - \* Return: Return the number of students in the database.
- `Bool SDB_AddEntry()`
  - \* Description: When the user chooses to add entry this function asks him for the required data in the database to add one student which is his (id, year, Course1\_ID, Course1\_grade, Course2\_ID, Course2\_grade, Course3\_ID, and



Course3\_grade) and then add to your database.

\*Return: return 1 if the data is added successfully return 0 if the data not added to your database which depend on the input of the user whether is the data entered is correct or not and the data is full or not

- Void SDB\_DeletEntry (uint32 id)
  - \* Description: delete the student data with the given id.
- Bool SDB\_ReadEntry (uint32 id)
  - \*Description: print the data of the user by searching on his data by the given id
  - \*Return: return 1 if the data is existed return 0 if this id not your database.
- Void SDB\_GetList (uint8 \* count, uint32 \* list)
  - Description: it gets the number of ids in the count and the list of all ids in list array that's you get his base address.
- Bool SDB\_IsIdExist (uint32 id)
  - \*Description: Checks if the given id the is



Exist

\* Return: return 1 if the id is existed and return 0 if the id not found.

Create SDBAPP.c which have the following function

- Void SDB\_APP ()

Description: it has a super loop it will make your project until the user chose to exit you will ask the user to choose between these options

1. To add entry, enter 1
2. To get used size in database, enter 2
3. To read student data, enter 3
4. To get the list of all student IDs, enter 4
5. To check is ID is existed, enter 5
6. To delete student data, enter 6
7. To check is database is full, enter 7
8. To exit enter 0

The it will call the SDB\_action

- Void SDB\_action (uint8 choice)

Description: it well takes the user choice and call the responding function for this choice.



You make the right response to give a valid  
Output for the user

In your main.c

Just call SDB\_APP () that's will start your program

Note that

- You have to follow the modular programing concept that's you have taken during the c sessions.
- You need to follow the MISRA rules when writing the functions
- You will need to comment every function how to well work depend in you implantation and why you implement it in this way.