

Data Persistence in SQLite in ios

SQLite can be used in iOS for handling data. It uses sqlite queries, which makes it easier for those who know SQL.

Steps Involved

Step 1 – Create a simple **View based application**.

Step 2 – Select your project file, then select targets and then add **libsqlite3.dylib** library in choose frameworks.

Step 3 – Create a new file by selecting File→ New → File... → select **Objective C class** and click next.

Step 4 – Name the class as **DBManager** with "**sub class of**" as NSObject.

Step 5 – Select create.

Step 6 – Update **DBManager.h** as follows –

```
#import <Foundation/Foundation.h>
#import <sqlite3.h>

@interface DBManager : NSObject {
    NSString *databasePath;
}

+(DBManager*)getSharedInstance;
-(BOOL)createDB;
-(BOOL) saveData:(NSString*)registerNumber name:(NSString*)name
    department:(NSString*)department year:(NSString*)year;
-(NSArray*) findByRegisterNumber:(NSString*)registerNumber;

@end
```

Step 7 – Update **DBManager.m** as follows –

```
#import "DBManager.h"
static DBManager *sharedInstance = nil;
static sqlite3 *database = nil;
static sqlite3_stmt *statement = nil;

@implementation DBManager

+(DBManager*)getSharedInstance {
    if (!sharedInstance) {
        sharedInstance = [[super allocWithZone:NULL]init];
        [sharedInstance createDB];
    }
    return sharedInstance;
}

-(BOOL)createDB {
```

```

NSString *docsDir;
NSArray *dirPaths;

// Get the documents directory
dirPaths = NSSearchPathForDirectoriesInDomains
(NSDocumentDirectory, NSUserDomainMask, YES);
docsDir = dirPaths[0];

// Build the path to the database file
databasePath = [[NSString alloc] initWithString:
[docsDir stringByAppendingPathComponent:@"student.db"]];
BOOL isSuccess = YES;
NSFileManager *filemgr = [NSFileManager defaultManager];

if ([filemgr fileExistsAtPath: databasePath ] == NO) {
    const char *dbpath = [databasePath UTF8String];
    if (sqlite3_open(dbpath, &database) == SQLITE_OK) {
        char *errMsg;
        const char *sql_stmt =
            "create table if not exists studentsDetail (regno integer
            primary key, name text, department text, year text)";

        if (sqlite3_exec(database, sql_stmt, NULL, NULL, &errMsg) != SQLITE_OK) {
            isSuccess = NO;
            NSLog(@"Failed to create table");
        }
        sqlite3_close(database);
        return isSuccess;
    } else {
        isSuccess = NO;
        NSLog(@"Failed to open/create database");
    }
}
return isSuccess;
}

- (BOOL) saveData:(NSString*)registerNumber name:(NSString*)name
department:(NSString*)department year:(NSString*)year; {
    const char *dbpath = [databasePath UTF8String];

    if (sqlite3_open(dbpath, &database) == SQLITE_OK) {
        NSString *insertSQL = [NSString stringWithFormat:@"insert into
        studentsDetail (regno,name, department, year) values
        (\">%d\","%@\", \"%@\", \"%@\""),[registerNumber integerValue],
        name, department, year];
        const char *insert_stmt = [insertSQL UTF8String];
        sqlite3_prepare_v2(database, insert_stmt,-1, &statement, NULL);

        if (sqlite3_step(statement) == SQLITE_DONE) {
            return YES;
        } else {

```

```

        return NO;
    }
    sqlite3_reset(statement);
}
return NO;
}

- (NSArray*) findByRegisterNumber:(NSString*)registerNumber {
    const char *dbpath = [databasePath UTF8String];

    if (sqlite3_open(dbpath, &database) == SQLITE_OK) {
        NSString *querySQL = [NSString stringWithFormat:
            @"select name, department, year from studentsDetail where
            regno=\"%@\\"", registerNumber];
        const char *query_stmt = [querySQL UTF8String];
        NSMutableArray *resultArray = [[NSMutableArray alloc] init];

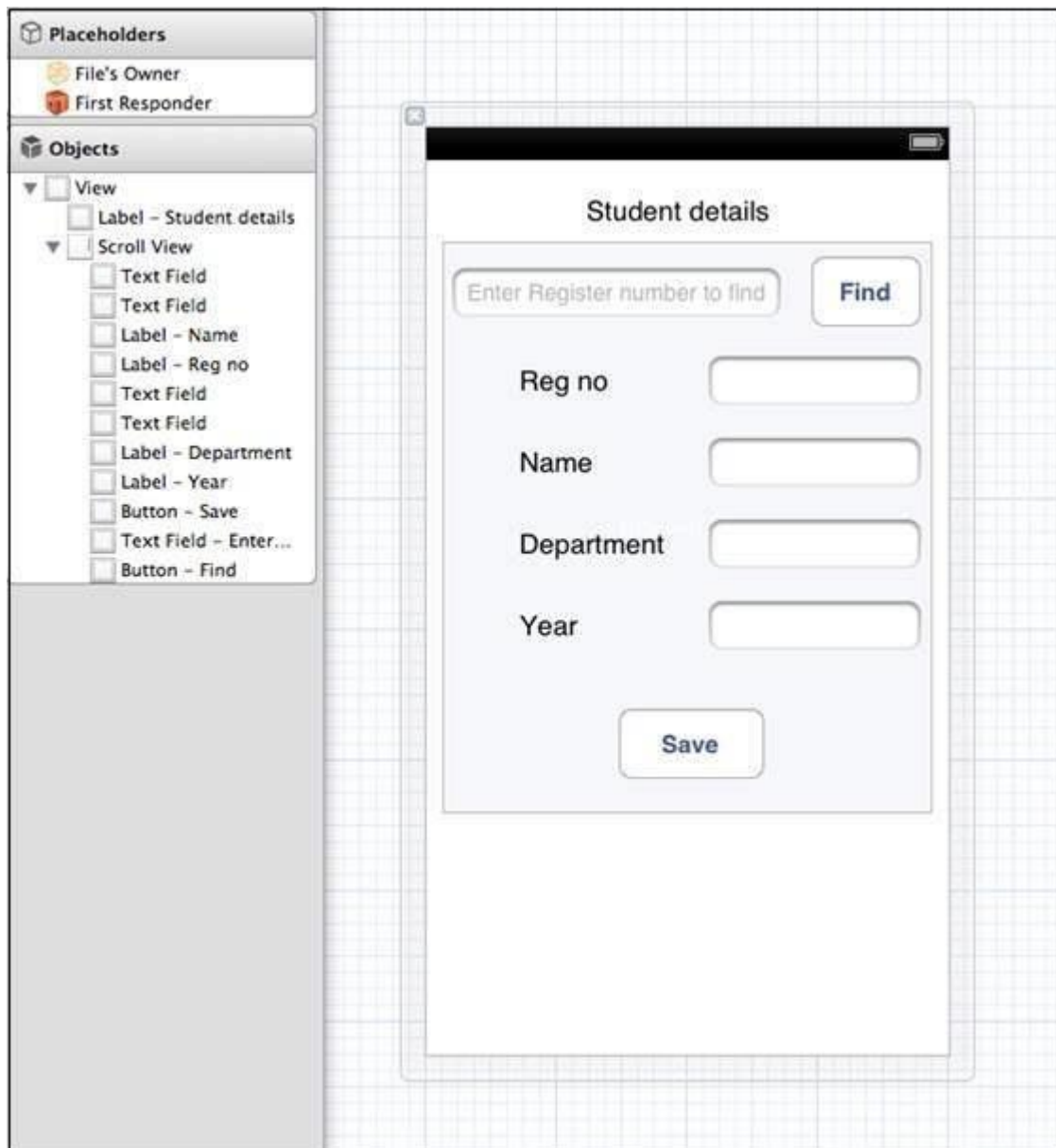
        if (sqlite3_prepare_v2(database, query_stmt, -1, &statement, NULL) == SQLITE_OK) {
            if (sqlite3_step(statement) == SQLITE_ROW) {
                NSString *name = [[NSString alloc] initWithUTF8String:
                    (const char *) sqlite3_column_text(statement, 0)];
                [resultArray addObject:name];

                NSString *department = [[NSString alloc] initWithUTF8String:
                    (const char *) sqlite3_column_text(statement, 1)];
                [resultArray addObject:department];

                NSString *year = [[NSString alloc] initWithUTF8String:
                    (const char *) sqlite3_column_text(statement, 2)];
                [resultArray addObject:year];
                return resultArray;
            } else {
                NSLog(@"Not found");
                return nil;
            }
        }
        sqlite3_reset(statement);
    }
}
return nil;
}

```

Step 8 – Update **ViewController.xib** file as follows –



Step 9 – Create IBOutlet for the above text fields.

Step 10 – Create IBAction for the above buttons.

Step 11 – Update **ViewController.h** as follows –

```
#import <UIKit/UIKit.h>
#import "DBManager.h"

@interface ViewController : UIViewController<UITextFieldDelegate> {
    IBOutlet UITextField *regNoTextField;
    IBOutlet UITextField *nameTextField;
    IBOutlet UITextField *departmentTextField;
    IBOutlet UITextField *yearTextField;
    IBOutlet UITextField *findByRegisterNumberTextField;
    IBOutlet UIScrollView *myScrollView;
}
}
```

```
-(IBAction)saveData:(id)sender;
-(IBAction)findData:(id)sender;
@end
```

Step 12 – Update **ViewController.m** as follows –

```
#import "ViewController.h"

@interface ViewController ()
@end

@implementation ViewController

- (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)
nibNameOrNil {
    self = [super initWithNibName:nibNameOrNil bundle:nibNameOrNil];

    if (self) {
        // Custom initialization
    }
    return self;
}

- (void)viewDidLoad {
    [super viewDidLoad];
    // Do any additional setup after loading the view from its nib.
}

- (void)didReceiveMemoryWarning {
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

-(IBAction)saveData:(id)sender {
    BOOL success = NO;
    NSString *alertString = @"Data Insertion failed";

    if (regNoTextField.text.length>0 &&nameTextField.text.length>0 &&
    departmentTextField.text.length>0 &&yearTextField.text.length>0 ) {
        success = [[DBManager sharedInstance]saveData:
        regNoTextField.text name:nameTextField.text department:
        departmentTextField.text year:yearTextField.text];
    } else {
        alertString = @"Enter all fields";
    }

    if (success == NO) {
        UIAlertView *alert = [[UIAlertView alloc] initWithTitle:
        alertString message:nil
        delegate:nil cancelButtonTitle:@"OK" otherButtonTitles:nil];
        [alert show];
    }
}
```

```

}

-(IBAction)findData:(id)sender {
    NSArray *data = [[DBManager sharedInstance]findByRegisterNumber:
    findByRegisterNumberTextField.text];

    if (data == nil) {
        UIAlertView *alert = [[UIAlertView alloc] initWithTitle:
        @"Data not found" message:nil delegate:nil cancelButtonTitle:
        @"OK" otherButtonTitles:nil];
        [alert show];
        regNoTextField.text = @"";
        nameTextField.text = @"";
        departmentTextField.text = @"";
        yearTextField.text = @"";
    } else {
        regNoTextField.text = findByRegisterNumberTextField.text;
        nameTextField.text = [data objectAtIndex:0];
        departmentTextField.text = [data objectAtIndex:1];
        yearTextField.text = [data objectAtIndex:2];
    }
}

#pragma mark - Text field delegate
-(void)textFieldDidBeginEditing:(UITextField *)textField {
    [myScrollView setFrame:CGRectMake(10, 50, 300, 200)];
    [myScrollView setContentSize:CGSizeMake(300, 350)];
}

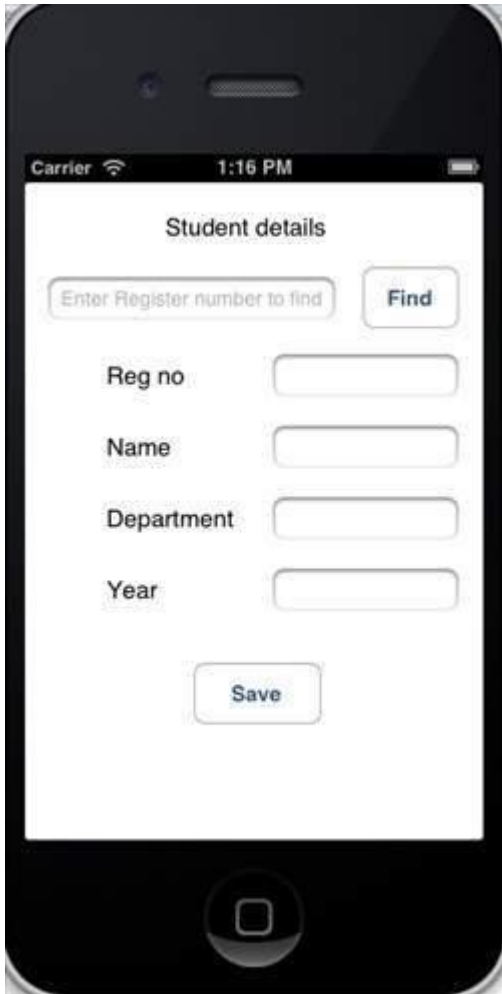
-(void)textFieldDidEndEditing:(UITextField *)textField {
    [myScrollView setFrame:CGRectMake(10, 50, 300, 350)];
}

-(BOOL) textFieldShouldReturn:(UITextField *)textField {
    [textField resignFirstResponder];
    return YES;
}
@end

```

Output

When we run the application, we'll get the following output where we can add and find the student details –



Previous Page