IOS - FILE HANDLING

File handling cannot be explained visually with the application and hence the key methods that are used for handling files are explained below. Note that the application bundle only has read permission and we won't be able to modify the files. You can anyway modify the documents directory of your application.

Methods used in File Handling

The methods used for **accessing** and **manipulating** the files are discussed below. Here we have to replace FilePath1, FilePath2 and FilePath strings to our required full file paths to get the desired action.

Check if a File Exists at a Path

```
NSFileManager *fileManager = [NSFileManager defaultManager];
//Get documents directory
NSArray *directoryPaths = NSSearchPathForDirectoriesInDomains
(NSDocumentDirectory, NSUserDomainMask, YES);
NSString *documentsDirectoryPath = [directoryPaths objectAtIndex:0];
if ([fileManager fileExistsAtPath:@""]==YES) {
    NSLog(@"File exists");
 }
```

Comparing Two File Contents

```
if ([fileManager contentsEqualAtPath:@"FilePath1" andPath:@" FilePath2"]) {
    NSLog(@"Same content");
}
```

Check if Writable, Readable, and Executable

```
if ([fileManager isWritableFileAtPath:@"FilePath"]) {
    NSLog(@"isWritable");
}
if ([fileManager isReadableFileAtPath:@"FilePath"]) {
    NSLog(@"isReadable");
}
if ( [fileManager isExecutableFileAtPath:@"FilePath"]){
    NSLog(@"is Executable");
}
```

Move File

```
if([fileManager moveItemAtPath:@"FilePath1"
toPath:@"FilePath2" error:NULL]){
    NSLog(@"Moved successfully");
}
```

Copy File

```
if ([fileManager copyItemAtPath:@"FilePath1"
toPath:@"FilePath2"  error:NULL]) {
    NSLog(@"Copied successfully");
}
```

Remove File

```
if ([fileManager removeItemAtPath:@"FilePath" error:NULL]) {
    NSLog(@"Removed successfully");
```

```
    }
```

Read File

```
NSData *data = [fileManager contentsAtPath:@"Path"];
```

Write File

```
[fileManager createFileAtPath:@"" contents:data attributes:nil];
```