```
//ÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐ
//  FILENAME:  eText.RichText.m
//  SUMMARY:   Implementation of RichText markup formats of eText (RTF,ETF)
//  CATEGORY:  RichText
//  PROTOCOLS: <ETFDSupport>
//  INTERFACE: None
//  AUTHOR:    Rohit Khare
//  COPYRIGHT: ©1993,94 California Institue of Technology, eText Project
//ÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐ
//  Implementation Comments
//      These methods have the least visible controls; they are supplicants
//  at the feet of the mercurial god of TeXT.
//ÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐ
//  History
//  10/17/94:  Cleaned up for eText5.
//  08/05/94:  Completely Rearchitected for 5.0. RK
//ÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐ
//  Imported Interfaces
//
    #import "eText.RichText.h"

@implementation eText(RichText)
//ÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐ
//  Path Operators
//
- readRTFfromPath:(const char *)path {
    NXStream *memstream;

    memstream = NXMapFile(path, NX_READONLY);
    if (memstream){
        [self readRTF: memstream];
        NXCloseMemory(memstream, NX_FREEBUFFER);
    } else
        return nil;
    return self;
}
- writeRTFtoPath:(const char *)path {
    NXStream *memstream;

    memstream = NXOpenMemory(NULL, 0, NX_WRITEONLY);
    if (memstream) {
        [self writeRTF:memstream];
        NXSaveToFile(memstream, path);
        NXCloseMemory(memstream, NX_FREEBUFFER);
    } else
        return nil;
    return self;
}

- readETFfromPath:(const char *)path {
    NXStream *memstream;

    memstream = NXMapFile(path, NX_READONLY);
    if (memstream){
        [self readETF: memstream];
        NXCloseMemory(memstream, NX_FREEBUFFER);
    } else
        return nil;
    return self;
}
- writeETFtoPath:(const char *)path {
    NXStream *memstream;

    memstream = NXOpenMemory(NULL, 0, NX_WRITEONLY);
    if (memstream) {
        [self writeETF:memstream];
        NXSaveToFile(memstream, path);
        NXCloseMemory(memstream, NX_FREEBUFFER);
    } else
        return nil;
    return self;
}

//ÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐÐ
//  Stream Operators
//
- readRTF: (NXStream *)s {
    return [self readRichText:s];}
- writeRTF: (NXStream *)s {
    id retVal;

    strictRTF = YES;
    retVal = [self writeRichText:s];
    strictRTF = NO;
    return retVal;
}
- writeRTF: (NXStream *)s from:(int) start to:(int) end {
    id retVal;

    strictRTF = YES;
    retVal = [self writeRichText:s from:start to:end];
    strictRTF = NO;
    return retVal;
}
```

```
- readETF: (NXStream *)s {
    return [self readRichText:s];}
- writeETF: (NXStream *)s {
    strictRTF = NO; return [self writeRichText:s];}
- writeETF: (NXStream *)s from:(int) start to:(int) end {
    strictRTF = NO;
    return [self writeRichText:s from:start to:end];
}

@end
```