Bin – идва от binary и съдържа вече компилираното .exe, което съдържа нашия код

Bin – debug е папката където се слагат файловете

File. – взимаме някакъв файл и правим нещо с него

String file = File.ReadAllText – взима целия текст в един стринг

String [] file = File.ReadAllLines(“newfile.txt”) – разбива текста по редове и го запаметява в масив

File.ReadAllBytes – връща bites масив. Можем да четем ВСЯКАКВИ файлове и после чрез библиотека да го конвертираме и т. Н.

File.WriteAllText(“output.txt”, file);

File.WriteAllText(output.txt file[0]) – записва само първия ред

Ако ползваме един файл можем да презапишем в него. След командата readalltext текста е взет и файла е затворен и готов да бъде използван отново от нас или от друг човек. При Java това не е директно направено – там се отваря един поток, по време на който никой не може да ползва файла

Винаги създава нов файл, изтрива всичко старо

Write където и да е създава нов файл

AppendAllText/Lines – добавя допълнително текс НА КРАЯ на файла

File[i] + “\r\n”

FileInfo fInfo = new FileInfo(“output.txt”)

Клас от методи като кога е бил създаден файла, директория на файла, името и т.н. Има много информация за файла. Дължината на файла е в байтове

Shift + tab

Directory.

CreateDirectory

Delete

Move(“Test”, “New Folder”)

GetFiles и GetDirectories

Byte / 1024 – kb / 1024 – mb

Клас System Exception

В catch можем да напишем (catch FormatException) примерно – ще търси само такъв тип грешка

Catch(FormatException forex)

Cw(forex.Message) – отпечатва съобщението на грешката

Може да управляваме в един try блок чрез няколко exception-а различни проблеми – при една грешка да даден определен резултат, при друга – друг – в зависимост от грешката

Try – finally

Finally блока винаги се изпълнява. Примерно опитай се да отвориш файла и накрая го затвори, независимо от резултата. Finally може да използва променлива от try блока докато само с try не можем да достъпим променлива от try извън него.