Java Randomaccessfile:

It is a predefined class in java.io package that allows users to navigate a file and read from it or write to it.

This is not possible with FileInputStream

And FileOutputStream.

Creating a RandomAccessFile:

RandomAccessFile rfile = new RandomAccessFile(“c:\\jay\\abc.txt”,”rw”);

The second argument to RandomAccessFile is the access modes.

“r” - Read Mode. Here Calling write methods will throw an IOException.

“rw” - Read and Write mode

“rwd” - Read and Write mode synchronously.All updates to file content are written to the disk synchronously.

“rws” - Read and Write mode. All the operations to file content or metadata are written to the disk synchronously.

Seeking in A Random Access File:

To read or write from a specific location, We must seek/navigate to that position using seek().

To get the current position in the file we can use RandomAccessFile and getFilePointer().

file.seek(200);

Long position = file.getFilePointer();

The current position is the index of the byte that the RandomAccessFile is currently positioned at.

How To Read Bytes From A RandomAccessFile?

RandomAccessFile file = new RandomAccessFile(“c:\\jay.abc.txt”,”rw”);

int readByte = file.read();

The read() , reads the byte located at the current position that is pointed by the RandomAccessFile pointer.

The read() method, increments the file pointer to points the next byte we read.

Read Array Of Bytes:

RandomAccessFile file = new RandomAccessFile(“c:\\jay\\emp.txt”,”rw”);

byte[] arr = new byte[1024];

int start = 0;

int length = 1024;

int bytesRead = file.read(arr,start,length);

Write Byte to a Random access File:

write(int b): To write one byte we have to use write() that takes one integer argument. The byte will be written to the current file position. The previous byte will be oveerites.

Calling write() will automatically advanced the current position of file by 1.

RandomAccessFile file = new RandomAccessFile(“c:\\jay\\employee.txt”,”rw”);

file.write(65);

Write Array Of Bytes:

write(byte[] arr) - This is an overloaded write method that takes one argument i.e a Byte array. This byte array will be written to the file from the current position.

byte [] arr = “Hello World”.getBytes();

file.write(arr);

Write a part of your byte array using RandomAccessFile.

file.write(arr,2,5);

Here the 2nd argument is the index number from where We have to start writing.

The third argument is the total number of bytes from that specific index.

close(): Using this method we close the instance of RandomAccessFile once we are done with the instance.

file.close();

try(RandomAccessFile file = new RandomAccessFile(“c:\\java\\A.txt”,”rw”)){

//Read or write operation will be performed here.

}

File:

The java.io.File class