Random Access of Binary Files

- There are two ways of accessing a file

1. Sequential Access - Read values in files one by one
2. Random Access - We can go to any byte of file and read it

- Random access works for binary files as well for example

```
with open('my.data','wb') as f:
    f.write(b'abcdefghij')
```

```
C:\Users\Abdul Bari\Desktop\MyPython>python BinWrite.py

C:\Users\Abdul Bari\Desktop\MyPython>dir
 Volume in drive C is OS
 Volume Serial Number is 3E57-713D

 Directory of C:\Users\Abdul Bari\Desktop\MyPython

03/25/2022  01:46 AM    <DIR>          .
03/25/2022  01:46 AM    <DIR>          ..
03/25/2022  01:46 AM                63 BinWrite.py
03/25/2022  01:46 AM                10 my.data
               2 File(s)             73 bytes
               2 Dir(s)  418,479,534,080 bytes free

C:\Users\Abdul Bari\Desktop\MyPython>type my.data
abcdefghij
C:\Users\Abdul Bari\Desktop\MyPython>
```

- Whenever a file is opened its pointer will be at index 0
- tell( ) method will tell you the current position of file pointer
- seek( ) is the method , which will move the file pointer to a certain given position in a file for example

```
with open('my.data','rb') as f:
    print(f.read(2).decode)
```

```
C:\Users\Abdul Bari\Desktop\MyPython>python BinRead.py
ab
```

- Moving the pointer

```
with open('my.data','rb') as f:
    print(f.read(2).decode())
    f.seek(3,0)
    print(f.read(1).decode())
```

```
C:\Users\Abdul Bari\Desktop\MyPython>python BinRead.py
ab
d
```

- Moving the pointer from current position

```
with open('my.data','rb') as f:
    print(f.read(2).decode())
    f.seek(3,1)
    print(f.read(1).decode())
```

```
C:\Users\Abdul Bari\Desktop\MyPython>python BinRead.py
ab
f
```

- Moving the pointer from end of file

```
with open('my.data','rb') as f:
    print(f.read(2).decode())
    f.seek(-3,2)
    print(f.read(1).decode())
```

```
C:\Users\Abdul Bari\Desktop\MyPython>python BinRead.py
ab
h
```