

# পাইথন প্রোগ্রামিং

পর্ব  
২৭

মোহাম্মদ মিজানুর রহমান নয়ন

সাবেক বিভাগীয় প্রধান, বিসিআই ইঞ্জিনিয়ারিং ইনস্টিটিউট; সাবেক লেকচারার, ওয়ার্ল্ড ইউনিভার্সিটি অব বাংলাদেশ ও পিপলস ইউনিভার্সিটি অব বাংলাদেশ

## ফাইল/ডিরেক্টরি ম্যানেজিং অপারেশন

ফাইল/ডিরেক্টরি ম্যানেজিং অপারেশনে পাইথনের os মডিউলের বিভিন্ন ফাংশন বা মেথড ব্যবহৃত হয়। os মডিউল ব্যবহার করে ফাইল/ডিরেক্টরি সংক্রান্ত বিভিন্ন ধরনের অপারেশন সম্পন্ন করা যায়। পাইথন প্রোগ্রাম ব্যবহার করে ডায়নামিক্যালি ফাইল এবং ডিরেক্টরি ম্যানেজ করা যায়। যেসব ফাইল/ডিরেক্টরি ম্যানেজিং কার্য পাইথন প্রোগ্রামের মাধ্যমে সম্পাদন করা যায় তার একটি তালিকা দেয়া হলো—

- ফাইল তৈরি এবং ওপেন করা।
- ফাইল রিনেম করা।
- ফাইল ডিলিট করা।
- কারেন্ট ডিরেক্টরির পাথ প্রদর্শন করা।
- ডিরেক্টরি পরিবর্তন করা।
- ডিরেক্টরি তৈরি করা।
- ডিরেক্টরি রিনেম করা।

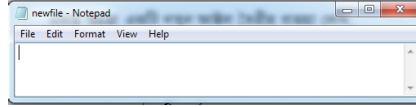
## ফাইল তৈরি এবং ওপেন করা

ফাইল তৈরি করার জন্য os মডিউলের open() মেথড ব্যবহার করা হয়। ফাইল তৈরি করার জন্য open() মেথডের সাথে কিছু ফ্লাগ ব্যবহার করা হয়, যা ফাইলটি কীভাবে ওপেন হবে তা নির্দেশ করে। এসব ফ্লাগের তালিকা দেয়া হলো—

ফ্লাগ	বর্ণনা
os.O_RDONLY	ফাইলকে রিড অনলি মুডে ওপেন করে।
os.O_WRONLY	ফাইলকে রাইট অনলি মুডে ওপেন করে।
os.O_RDWR	ফাইলকে রিড এবং রাইট মুডে ওপেন করে।
os.O_NONBLOCK	ফাইলকে ওপেন করার সময় ব্লক করে না।
os.O_APPEND	ফাইলকে ডাটা সংযুক্ত করার জন্য ওপেন করে।
os.O_CREAT	নতুন ফাইল তৈরি করে।

এবার আমরা একটি নতুন ফাইল তৈরির প্রক্রিয়া দেখি,

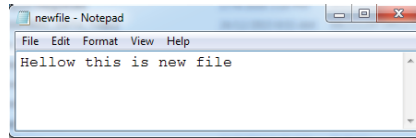
```
import os
fl=os.open("c:/newfile.txt",os.O_CREAT)
এখন c:\ ড্রাইভটি পর্যবেক্ষণ করলে দেখা যাবে যে newfile.txt নামে নতুন ফাইলটি তৈরি হয়েছে।
```



os মডিউল ব্যবহার করে ফাইলকে রিড রাইট মুডে ওপেন করার জন্য os.O\_RDONLY ফ্লাগ ব্যবহার করতে হবে। অতপর ফাইলটিতে ডাটা রাইট করার জন্য নিচের মতো প্রোগ্রাম লিখতে হবে। এখানে উল্লেখ্য, ফাইলে কোনো স্ট্রিং ডাটাকে রাইট করার জন্য প্রথমে এটিকে বাইট টাইপে কনভার্ট করে নিতে হবে।

```
import os
fl=os.open("c:/newfile.txt",os.O_RDWR)
os.write(fl, bytes("Hello this is new file", "utf-8"))
os.close(fl)
অথবা,
fl=os.open("c:/newfile.txt",os.O_RDWR)
data="Hello this is new file"
os.write(fl,data.encode('utf-8'))
os.close(fl)
```

এবার ফাইলকে ওপেন করলে আমরা ফাইলে লিখিত ডাটাগুলো দেখতে পাব।



## ফাইল রিনেম করা

পাইথনে কোনো অপারেটিং সিস্টেম ফাইলকে রিনেম করার জন্য os মডিউলের rename() মেথড ব্যবহার করা হয়। rename মেথডটি দুটি প্যারামিটার গ্রহণ করে। প্রথমটি হচ্ছে পুরনো ফাইল নেম এবং পরেরটি হচ্ছে নতুন যে নাম দেয়া হবে সেই ফাইল নেম। ফাইলের নাম ফাইল লোকেশন এবং এক্সটেনশনসহ দিতে হবে। ফাইলের নাম পরিবর্তন করার প্রোগ্রাম স্টেটমেন্ট নিচে দেয়া হলো—

```
os.rename("c:/newfile.txt", "c:/newrename.txt")
```

## ফাইল ভেরিফাই করা

কোনো ফাইল নির্দিষ্ট লোকেশনে আছে কিনা তা ভেরিফাই করে দেখার জন্য os মডিউলের isfile() মেথড ব্যবহার করা হয়। এটি ফাইল পাথকে ভেরিফাই করে দেখে ফাইলটি উল্লিখিত লোকেশনে অবস্থান করছে কিনা। ফাইলটি পাওয়া গেলে True রিটার্ন করে আর না পাওয়া গেলে False রিটার্ন করে। যেমন—

```
os.path.isfile('c:\\Python34\\Scripts\\pip3.exe')
>>> os.path.isfile('c:\\Python34\\Scripts\\pip3.exe')
True
```

উপরোক্ত উদাহরণে দেখা যাচ্ছে যে pip3.exe ফাইলটি 'c:\\Python34\\Scripts' লোকেশনে অবস্থান করছে, তাই এটি True রিটার্ন করেছে।

এবার isfile মেথডে একটি ডিরেক্টরি লোকেশন দিয়ে দেখা যাক তা কি ফলাফল প্রদান করে—

```
os.path.isfile('c:\\Python34\\Scripts\\newtest')
>>> os.path.isfile('c:\\Python34\\Scripts\\newtest')
False
```

উপরোক্ত উদাহরণের ফলাফল আমরা দেখতে পাচ্ছি False, কারণ newtest কোনো ফাইল নয়, এটি একটি ডিরেক্টরি, তাই সঠিক লোকেশনে থাকা শর্তেও isfile মেথড False রিটার্ন করেছে।

## ফাইল ডিলিট করা

কোনো অপারেটিং সিস্টেম ফাইলকে পাইথন প্রোগ্রাম ব্যবহার করে ডিলিট করা যায়। ফাইলকে ডিলিট করার জন্য os মডিউলের remove মেথড ব্যবহার করতে হবে। ফাইলকে ডিলিট করার পাইথন প্রোগ্রাম কোড নিচে দেয়া হলো—

```
os.remove("c:/newrename.txt")
```

## কারেন্ট ডিরেক্টরি লোকেশন বের করা

কারেন্ট ডিরেক্টরি লোকেশন অর্থাৎ বর্তমানে যে ডিরেক্টরিতে অবস্থান করছে তা বের করার জন্য os মডিউলের getcwd() মেথড ব্যবহার করা হয়। যেমন—

```
import os
os.getcwd() কাজ
```

ফিডব্যাক : mrm\_bd@yahoo.com