# <https://python.maateen.me/docs/file/>

# Introduction

ফাইল নিয়ে কাজ করতে গিয়ে আমরা তিনটা জিনিস শিখব। ফাইল ওপেন (open) করা, ওপেন করা ফাইল রিড (read) করা, রিড করার পর ফাইল রাইট (write) করা আর শেষমেষ ফাইল ক্লোজ (close) করা।

* পাইথনে ফাইল ওপেন করার জন্য open() ফাংশনটা ব্যবহার করব আমরা। এই ফাংশনটা তিনটা প্যারামিটার নিয়ে কাজ করে।ফাইলের নাম: প্রথম প্যারামিটার হল ফাইলের নাম। আমাদের স্ক্রিপ্ট ও আলোচ্য ফাইল যদি একই ডিরেক্টরিতে হয় তবে শুধু ফাইলের নামটা স্ট্রিং হিসাবে দিতে হয়। ফাইল যদি আলাদা ডিরেক্টরিতে থাকে তবে পুরো পাথটা স্ট্রিং হিসাবে দিতে হবে।
* অ্যাক্সেস মোড: দ্বিতীয় প্যারামিটার হল অ্যাক্সেস মোড। একটা ফাইলকে আমরা বিভিন্ন কাজের জন্য ওপেন করতে পারি। সেক্ষেত্রে ভিন্ন ভিন্ন কাজের জন্য ভিন্ন ভিন্ন মোডে ফাইলটা ওপেন করতে হবে। নিচে কতগুলো মোডের নাম ও কোন কাজে লাগে তা দেয়া হল।

| **মোড** | **বর্ণনা** |
| --- | --- |
| r | এটা হল ডিফল্ট মোড। কোন মোড স্পেসিফাই করে না দেয়া হলে এটা আপনা-আপনি কাজ করে। এটা শুধু রিড করার জন্য একটা ফাইলকে ওপেন করে। আর ফাইল পয়েন্টার ফাইলের শুরুতে থাকে। |
| rb | সবকিছু মোড r এর মতই। শুধু তফাৎ হল ফাইলটা বাইনারি ফরম্যাটে ওপেন করে। |
| r+ | সবকিছু মোড r এর মতই। এক্সট্রা বেনেফিট হল ফাইলটা রাইটও করা যায়। |
| rb+ | r+ আর rb এর কম্বিনেশন। |
| w | ফাইলকে রাইট মোডে ওপেন করে। ফাইলের অস্তিত্ব না থাকলে নিজ থেকেই ফাইল তৈরি করে নেয়। আর অস্তিত্ব থেকে থাকলে পুরাতন ফাইলকে ওভাররাইট করে। |
| wb | সবকিছু মোড w এর মতই। শুধু তফাৎ হল ফাইলটা বাইনারি ফরম্যাটে ওপেন করে। |
| w+ | সবকিছু মোড w এর মতই। এক্সট্রা বেনেফিট হল ফাইলটা রিডও করা যায়। |
| wb+ | w+ আর wb এর কম্বিনেশন। |
| a | ফাইলকে অ্যাপেন্ড (append) মোডে ওপেন করে। এই মোডে ফাইল পয়েন্টার ফাইলের একেবারে শেষে থাকে। ফাইল রাইট করলে তা ফাইলের শেষে যোগ হয়। ফলে আগের সব ডাটা রক্ষিত থাকে। তবে ফাইলের অস্তিত্ব না থাকলে নিজ থেকেই ফাইল তৈরি করে নেয়। |
| ab | সবকিছু মোড a এর মতই। শুধু তফাৎ হল ফাইলটা বাইনারি ফরম্যাটে ওপেন করে। |
| a+ | সবকিছু মোড a এর মতই। এক্সট্রা বেনেফিট হল ফাইলটা রিডও করা যায়। |
| ab+ | a+ আর ab এর কম্বিনেশন। |

* বাফারিং: তৃতীয় প্যারামিটার হল বাফারিং। এটা একটু উচ্চতর বিষয়। আমরা এটা সম্পর্কে পরে জানব।

কয়েকটা উদাহরণ দিলে বিষয়গুলো আরো পরিষ্কার হয়ে যাবে আমাদের কাছে। এজন্য আমরা ছোট্ট একটা কাজ করব। আমাদের স্ক্রিপ্টের সাথে একই ডিরেক্টরিতে test.txt নামে একটা ফাইল তৈরি করব। তারপর ফাইলে The name of my country is Bangladesh. কথাটা লিখে সেইভ করে রাখব। এবার কয়েকটা প্রোগ্রাম লেখা যাক।

my\_file = open('test.txt', 'r')

content = my\_file.read()

print(content)

my\_file.close()

দ্বিতীয় লাইনে ফাইলটা read() ফাংশনের (আসলে মেথড) সাহায্যে রিড করে ডাটা content ভ্যারিয়েবলে অ্যাসাইন করেছি। আর শেষ লাইনে close() ফাংশন (মেথড) ব্যবহার করে ফাইলটা ক্লোজ করে দিয়েছি। আরেকটা প্রোগ্রাম দেখা যাক।

my\_file = open('test.txt', 'w')

my\_file.write('I am Maksudur Rahman Maateen.')

my\_file.close()

এখানে write() ফাংশন (মেথড) ব্যবহার করে আমরা ফাইলটা রাইট করেছি। এই ফাংশনের ভিতর স্ট্রিং হিসাবে যা দেয়া হবে তাই রাইট হবে ফাইলে। প্রোগ্রামটা রান করলে আমরা আউটপুট কিছুই পাব না। তবে ফাইলটা যদি নরমালি ওপেন করে দেখি তাহলে দেখব সেখানে এখন The name of my country is Bangladesh. এর পরিবর্তে I am Maksudur Rahman Maateen. লেখা রয়েছে। এবার আরেকটা প্রোগ্রাম দেখা যাক।

my\_file = open('test.txt', 'a')

my\_file.write('I am from Bangladesh.')

my\_file.close()

এই প্রোগ্রামটা রান করলেও আমরা আউটপুট কিছুই পাব না। তবে ফাইলটা যদি নরমালি ওপেন করে দেখি তাহলে দেখব সেখানে এখন I am Maksudur Rahman Maateen. এর পরে I am from Bangladesh. লেখা রয়েছে। এবার আরেকটা প্রোগ্রাম দেখা যাক।

my\_file = open('test.txt', 'r')

content = my\_file.read(5)

print(content)

content = my\_file.read()

print(content)

position = my\_file.tell()

print(position)

my\_file.seek(0, 0)

content = my\_file.read()

print(content)

my\_file.close()

**আউটপুট**

I am

Maksudur Rahman Maateen.I am from Bangladesh.

51

I am Maksudur Rahman Maateen.I am from Bangladesh.

লম্বা-চওড়া একটা প্রোগ্রাম। আমরা এবার এটা ব্যাখায় যাব। দ্বিতীয় লাইনে read() এর ভিতর আমরা 5 ভ্যালু পাস করেছি। এর ফলে ফাইলের প্রথম ৫ টা ক্যারেক্টার পর্যন্ত রিড হবে শুধু। একটা কথাটা তো আমরা নিশ্চয়ই জানি, স্পেসও একটা ক্যারেক্টার। যাহোক, ষষ্ঠ লাইনে এসে আমরা tell() ফাংশন দিয়ে ফাইল পয়েন্টারের পজিশন খুঁজে বের করেছি। আর অষ্টম লাইনে seek() ফাংশন দিয়ে ফাইল পয়েন্টার আবার শুরুতে নিয়ে গিয়েছি। আরেকটা উদাহরণ দেখা যাক।

with open('test.txt', 'r') as my\_file:

content = my\_file.read()

print(content)

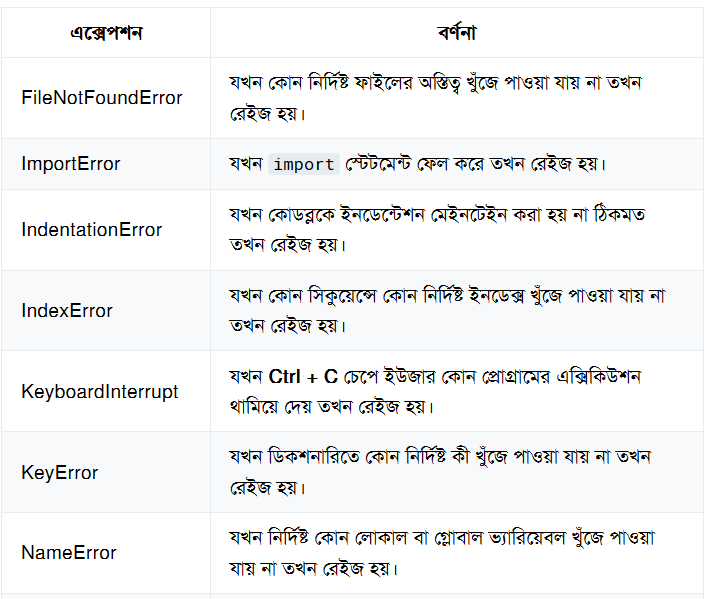
**আউটপুট**

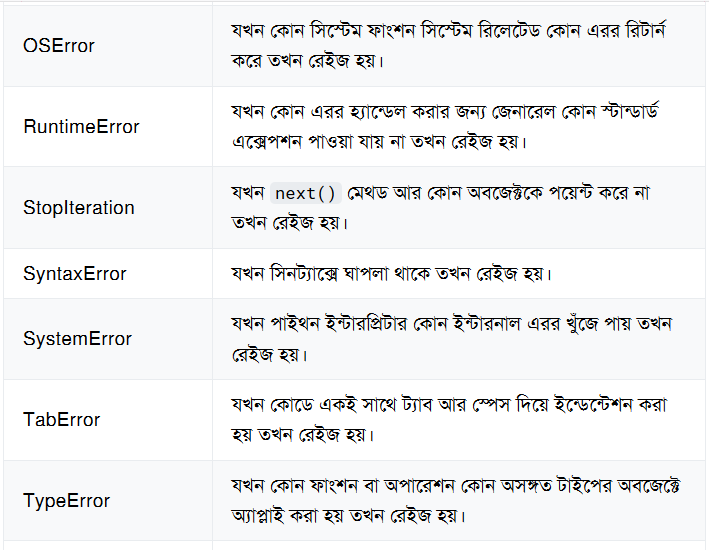
I am Maksudur Rahman Maateen.I am from Bangladesh.

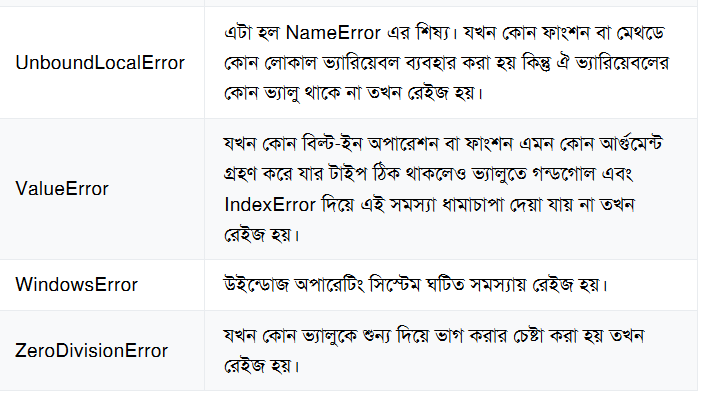
এই প্রোগ্রামে আমরা with স্টেটমেন্ট ব্যবহার করেছি। স্টেটমেন্টটা ব্যবহার করে ফাইলটা my\_file নামে ওপেন করেছি। বাদবাকি অপারেশন নরমাল। কেউ কি লক্ষ্য করেছি যে আমরা ফাইলটা ক্লোজ করিনি। আসলে এটাই এই স্টেটমেন্টের সুবিধা। কাজ শেষে বা কোন এরর পেলে নিজ থেকেই ফাইলটা ক্লোজ করে দেয়।

এই ছিল ফাইল নিয়ে আমাদের বেসিক আলোচনা। এখন বাড়ির কাজ আছে। একটা ফাইলে লুপ ঘুরিয়ে ১ থেকে ১০০ পর্যন্ত লাইন বাই লাইন রাইট করব আমরা। কি পারব তো সবাই? চেষ্টা করলে অবশ্যই পারব।

# এরর হ্যান্ডলিং







## এক্সেপশন (Exception) কি?

সহজ কথায়, এক্সেপশন হল একটা ঘটনা যা প্রোগ্রাম চলার সময় কোন সমস্যার উদগীরণ হলে ঘটে। একটা উদাহরণ দিয়ে বিষয়টা আরো পরিষ্কার করে তুলে ধরা যাক। ফাইল চাপ্টারে আমরা সবার শেষে যে প্রোগ্রামটা লিখেছিলাম সেটা আবার চালাব আমরা। তবে এবার test.txt ফাইলটা ঐ জায়গা থেকে ডিলিট করে দেয়ার পর।

with open('test.txt', 'r') as my\_file:

content = my\_file.read()

print(content)

**আউটপুট**

Traceback (most recent call last):

File "/home/ugcoder/Desktop/test.py", line 1, in <module>

with open('test.txt', 'r') as my\_file:

FileNotFoundError: [Errno 2] No such file or directory: 'test.txt'

test.txt ফাইলটা না পেয়ে FileNotFoundError থ্রো করেছে পাইথন, তারপর প্রোগ্রামের এক্সিকিউশন বন্ধ হয়ে গেছে। ফলে শেষের নির্ভেজাল স্টেটমেন্টটাও এক্সিকিউট হয় নাই। এইরকম সমস্যার নিরসন করতেই এক্সেপশন হ্যান্ডলিং বা এরর হ্যান্ডলিং।

## try … except

উপরের এররটাই এবার আমরা হ্যান্ডেল করব। আগে আমরা প্রোগ্রামটা দেখব তারপর ব্যাখ্যায় যাব।

try:

with open('test.txt', 'r') as my\_file:

content = my\_file.read()

print(content)

except:

print('The file does not exist.')

print('Made by Maateen.')

**আউটপুট**

The file does not exist.

Made by Maateen.

এবার কিন্তু আর প্রোগ্রামটা ঠুস করেই শেষ হয়ে যায়নি। এররটা হ্যান্ডেল হয়ে শেষ স্টেটমেন্ট অবধি সুন্দরভাবে প্রোগ্রাম এক্সিকিউট হয়েছে।

try...except এর ব্যাপারটা হল এরকম: নরমাল কোডগুলো try ব্লকের ভিতর থাকবে। আর কোন এক্সেপশন রেইজ হলে except ব্লকের কোড এক্সিকিউট হবে। কোন নির্দিষ্ট এক্সেপশন হ্যান্ডেল করার জন্য কোড লিখতে চাইলে except এর পরে স্পেস দিয়ে এক্সেপশনের নাম দিয়ে দিতে হয়। আবার চাইলে এক্সেপশনের থ্রো করা মেসেজ হোল্ড করে প্রিন্ট করে ইউজারকে দেখানোও যায়। একটা উদাহরণ দেখা যাক।

try:

with open('test.txt', 'r') as my\_file:

content = my\_file.read()

print(content)

except FileNotFoundError:

print('The file does not exist.')

print('Made by Maateen.')

try:

my\_list = []

print(my\_list[0])

except IndexError as e:

print(e)

**আউটপুট**

The file does not exist.

Made by Maateen.

list index out of range

এখানে আমরা এক্সেপশনের নাম ধরে এরর হ্যান্ডেল করেছি। শেষের এক্সেপশনে পাইথনের জেনারেট করা এরর মেসেজই প্রিন্ট করে দিয়েছি। আচ্ছা, একটা try ব্লকের কোডে কয়েকটা এরর হতে পারে। সেক্ষেত্রে কিভাবে এরর হ্যান্ডেল করব আমরা?

আসলে একটা try ব্লকের জন্য যত খুশি তত except ব্লক লেখা যায়। তবে প্রতিটা except ব্লকে এক্সেপশনের নাম উল্লেখ করতে হবে। একটা নির্দিষ্ট এক্সেপশন রেইজ হলেই কেবল ঐ নির্দিষ্ট except ব্লক এক্সিকিউট হবে। একটা উদাহরণ দেখা যাক।

try:

my\_file = open('test.txt')

content = my\_file.read()

i = int(content.strip())

except IOError as e:

errno, strerror = e.args

print("I/O error({0}): {1}".format(errno,strerror))

except ValueError:

print("No valid integer in line.")

except:

print("Unexpected error!")

**আউটপুট**

I/O error(2): No such file or directory

এখানে আমরা একটা try ব্লকের জন্য তিনটা except ব্লক লিখেছি। প্রথমটা IOError এর জন্য, দ্বিতীয়টা ValueError এর জন্য আর শেষেরটা ঐ দুইটা বাদে যেকোন এররের জন্য। এ ধরনের except কে পাইথনিক ভাষায় Bare Except বলা হয়। বাংলায় অবশ্য আমরা এর একটা নাম দিতে পারি - ল্যাংটা এক্সেপ্ট। আর পারতপক্ষে ল্যাংটা এক্সেপ্ট স্কিপ করা উচিত আমাদের। কারণ, বেয়ার এক্সেপ্ট সব ধরনের এররকে হাইড করে দেয় ফলে আমরা জানতেই পারব না ঠিক কোন এক্সেপশনটাকে আমরা ক্যাচ করছি। তাই দেশ ও জাতির বৃহত্তর কল্যাণের স্বার্থে আমরা সবসময় এক্সেপশনের নাম উল্লেখ করে এরর হ্যান্ডেল করব।

আমরা চাইলে সবগুলো এররকে একটা except ব্লকে সেটেল করে দিতে পারতাম।

try:

my\_file = open('test.txt')

content = my\_file.read()

i = int(content.strip())

except (IOError, ValueError):

pass

## try … except … else

else ব্লক except ব্লকের শেষে বসে। try ব্লকে কোন এক্সেপশন রেইজ না হলেই কেবল else ব্লকের কোড এক্সিকিউট হয়। একটা উদাহরণ দেখা যাক।

try:

a = 5

b = 8

print(a + b)

except ValueError as e:

print(e)

else:

print('There is no exception.')

**আউটপুট**

13

There is no exception.

কোন এক্সেপশন রেইজ হয়নি। তাই else ব্লকের কোড এক্সিকিউট হয়েছে।

## try … except … finally

finally ব্লক একেবারে শেষে বসে। আর কোন এক্সেপশন রেইজ হোক আর নাই বা হোক, finally ব্লকের কোড ঠিকই এক্সিকিউট হয়। এজন্য একে ক্লীন-আপ অ্যাকশন বলা হয়। একটা উদাহরণ দেখা যাক।

try:

with open('test.txt', 'r') as my\_file:

content = my\_file.read()

print(content)

except FileNotFoundError:

print('The file does not exist.')

finally:

print('To be or not to be that is the question.')

**আউটপুট**

The file does not exist.

To be or not to be that is the question.

test.txt ফাইলটা না থাকায় FileNotFoundError এক্সেপশন রেইজ হয়েছে। কিন্তু তারপরও finally ব্লকের কোড ঠিকই রেইজ হয়েছে।

## রেইজ এক্সেপশন

পাইথনে তো অনেক বিল্ট-ইন এক্সেপশন আছে ( <https://docs.python.org/3/library/exceptions.html> )। আমরা চাইলে এদেরকে নিজ থেকে রেইজ করতে পারি।

try:

raise NameError('Hey! It is a custom error message.')

except NameError as e:

print(e)

**আউটপুট**

Hey! It is a custom error message.

এক্সেপশন রেইজ করার জন্য raise স্টেটমেন্ট ব্যবহার করতে হয়। raise এর পরে বিল্ট-ইন এক্সেপশনের নাম দিয়ে এর সাথেই ব্রাকেটের ভিতর এরর মেসেজ স্ট্রিং হিসাবে পাস করতে হয়।

বিল্ট-ইন এক্সেপশন ছাড়াও পাইথনে ইউজার-ডিফাইন্ড এক্সেশন ব্যবহার করা যায়। কিন্তু আমরা তো এখনও ছোট। তাই এখনই আমরা এটা শিখব না। ক্লাস, অবজেক্ট, মেথড সম্পর্কে যখন আমাদের ধারণা পরিষ্কার হবে তখন আমরা ইউজার-ডিফাইন্ড এক্সেপশন পয়দা করব। ততক্ষণ হ্যাপি এরর হ্যান্ডলিং!

# read a txt file

# this code read a specific amount of cherecter

f= open("data/mh.txt","r")

print(f.read(7))

#this code read full txt file with in [] breaket,method 1

with open('mh.txt') as fobj:

    lines =fobj.readlines()

print(lines)

#this code read full txt file with try-except handelling funtion,method 2

try:

    with open('data/mh.txt') as fobj:

        contents=fobj.read()

        print(contents)

except Exception as e:

    print('file not found',e)

# read txt file with enumerate funtion

with open('data/mh.txt') as fobj:

     for x ,y in enumerate(fobj): # here enumerate use for numbering and upper for capital latter

       print(x+1,y.upper())

"""

# read a csv/excel file

import csv

with open('data/excel.csv') as obj:

    fcsv=csv.reader(fobj)

    sum=0

    for i,row in enumerate(fcsv):

        print(i,row[0],row[1])

        sum +=int(row[1]) if i>0 else 0

    print(sum)

"""

import webbrowser

def main():

    print("open sites")

    with open("sites.txt") as fobj:

        try:

            for num,url in enumerate(fobj):

                webbrowser.open\_new\_tab(url.strip())

        except Exception as e:

            print(e)

    print("\nEnjoy!")

if \_name\_ == '\_main\_':main()