Rangamati Nir,DUET,Gazipur-1707
info.sosbd@gmail.com
09678 677 677, 01751 402449

📘 পাস বুক- বেসিক অ্যাপ্লিকেশন ডেভেলপমেন্ট ইউজিং পাইথন

⏱️ পড়ার সময় ঃ মাত্র ৮ ঘণ্টা

🎯 লক্ষ্য ঃ ৩৮ মার্কস

🔧 **ডিপ্লোমা ইঞ্জিনিয়ারিং শিক্ষার্থীদের জন্য দ্রুত রিভিশনের পারফেক্ট সলিউশন!**

📖 পৃষ্ঠা সংখ্যা ঃ

☞ ই-বুক ভার্সন ঃ ৮০ পৃষ্ঠা

☞ প্রিন্টযোগ্য কাগজে ঃ ২৭ পৃষ্ঠা

📝 পরীক্ষা সংক্রান্ত তথ্য ঃ

☞ মোট মার্কস ঃ ৬০

☞ পাস মার্কস ঃ ২৪

📊 প্রশ্ন ও মার্কস বিশ্লেষণ ঃ

✅ অতিসংক্ষিপ্ত প্রশ্ন ঃ

মোট পড়তে হবে ঃ ৪৪ টি

নিশ্চিত কমন ঃ ৫ টি

নিশ্চিত মার্কস ঃ ৫ (প্রতি প্রশ্নে ১ মার্কস)

সংক্ষিপ্ত প্রশ্ন ঃ

মোট পড়তে হবে ঃ ৩১ টি
নিশ্চিত কমন ঃ ৫ টি
নিশ্চিত মার্কস ঃ ১০ (প্রতি প্রশ্নে ২ মার্কস)

রচনামূলক প্রশ্ন ঃ

মোট পড়তে হবে ঃ ১৯ টি
নিশ্চিত কমন ঃ ৩ টি
নিশ্চিত মার্কস ঃ ১৮ (প্রতি প্রশ্নে ৬ মার্কস)

কিভাবে পড়বেন এই পাস বুক ঃ

Step 1- অধ্যায় বাছাই (সবার আগে শেষ করুন) ঃ

অধ্যায়	বিষয়	কারণ
১	পাইথন ফাংশনস	অতি ও সং আসে
৫,৬	অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং এর ফোর পিলার এবং পাইথন ইটারেটর, জেনারেটর এবং ডেকোরেটর	অতি, সং ও রচনা আসে
৭, ১০	পাইথন-এর এক্সসেপশন এবং এরর হ্যান্ডিলিং এবং পাইথন রেগুলার এক্সপ্রেশন	অতি ও সং, রচনামূলক আসে
২, ৩, ৪, ৮, ৯	অবশিষ্ট অধ্যায়	অতি ও রচনামূলক আসে
	Program	রচনা আসে

📅 স্টাডি প্ল্যান (মোট সময় ঃ ৮ ঘণ্টা) ঃ

⏱️ ১ ঘণ্টা ⟶ অধ্যায়-১ (পাইথন ফাংশনস)

⏱️ ২ ঘণ্টা ⟶ অধ্যায়-৫-৬ (অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং এর ফোর পিলার এবং পাইথন ইটারেটর, জেনারেটর এবং ডেকোরেটর)

⏱️ ২ ঘণ্টা ⟶ অধ্যায়- ৭, ১০ (পাইথন-এর এক্সসেপশন এবং এরর হ্যান্ডিলিং এবং পাইথন রেগুলার এক্সপ্রেশন)

⏱️ ২ ঘণ্টা ⟶ অধ্যায়-২, ৩, ৪, ৮, ৯ (অবশিষ্ট অধ্যায়)

⏱️ ১ ঘণ্টা ⟶ Program

🎯 পাস বুকের প্রধান লক্ষ্য ঃ

✅ দ্রুত রিভিশনের সুবিধা- পরীক্ষার আগে সময় বাঁচায়

✅ গুরুত্বপূর্ণ প্রশ্ন- উত্তরের সংক্ষিপ্ত বিন্যাস

✅ সারাংশভিত্তিক সাজানো- এক নজরে পুরো বিষয়টা দেখে ফেলা

✅ মূল টপিক ফোকাস- আত্মবিশ্বাস গড়ে তোলার সহায়ক

📌 বিশেষ দ্রষ্টব্য ঃ

☞ এটি একটি পূর্ণাঙ্গ টেক্সটবুক নয়- বরং সঠিক সময় ব্যবস্থাপনার রিভিশন গাইড।

☞ পরীক্ষার আগের “last moment” প্রস্তুতির জন্য উপযুক্ত সহকারী!

বেসিক অ্যাপ্লিকেশন ডেভেলপমেন্ট ইউজিং পাইথন

অধ্যায়	অধ্যায়ের নাম	পৃষ্ঠা
১	পাইথন ফাংশনস	৩-৬
২	ফাইল অপারেশন ইন পাইথন	৭-১১
৩	মডিউল, প্যাকেজ এবং অ্যাপ্লিকেশন সফটওয়্যার	১২-১৫
৪	অবজেক্ট ওরিয়েন্টেড প্রোগ্রমিং-এর মৌলিক ধারণা	১৬-২২
৫	অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং এর ফোর পিলার	২৩-২৯
৬	পাইথন ইটারেটর, জেনারেটর এবং ডেকোরেটর	৩০-৩২
৭	পাইথন-এর এক্সসেপশন এবং এরর হ্যান্ডিলিং	৩৩-৩৬
৮	পাইথন-এর লগিং	৩৭-৪০
৯	ইউনিট টেস্টিং	৪১-৪৩
১০	পাইথন রেগুলার এক্সপ্রেশন	৪৪-৪৮
১১	অ্যাপ্লিকেশন সফটওয়্যার	৪৯-৫৫
	বিগত সালের প্রশ্ন	৫৬-৫৬

Softmax Publications

অধ্যায়-১ পাইথন ফাংশনস

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

**** ১। ফাংশন (Function) বলতে কী বুঝ?** বাকাশিবো- ২০২৩

উত্তর ঃ ফাংশন হচ্ছে কোনো নির্দিষ্ট কোড ব্লক যা নির্দিষ্ট কিছু কাজের জন্য ব্যবহৃত হয়। যার একটি নির্দিষ্ট নাম থাকে, যা এক বা একাধিক স্টেটমেন্টের সমন্বয়ে গঠিত এবং নির্দিষ্ট কোনো সমস্যা সমাধানের জন্য ব্যবহৃত হয়।

***** ২। Built-in Function বা Library Function কাকে বলে?** বাকাশিবো- ২০১৯, ২০

উত্তর ঃ যে সকল Function এর কাজ Developer কর্তৃক পূর্বে থেকেই নির্দিষ্ট কিছু কাজ সম্পন্ন করার জন্য তৈরি করা, তাকে Built-in বা Library Function বলে।

***** ৩। Calling ও Called Function কাকে বলে?** বাকাশিবো- ২০২১

উত্তর ঃ Calling Function ঃ একটি ফাংশন অপর একটি ফাংশনকে যখন ব্যবহার করে, তাকে Calling Function বলে।

Called Function ঃ ব্যবহারকারী তার নিজের প্রয়োজনে যে Function কে Call করে, তাকে Called Function বলে।

***** ৪। ফাংশন প্রটোটাইপ (Function Prototype) বলতে কী বুঝ?** বাকাশিবো- ২০১৯

উত্তর ঃ নির্ধারিত ফাংশনটি কী ধরনের কাজ করবে, কয়টি ডাটা গ্রহণ করবে তা Compiler কে জানানোর প্রক্রিয়াকে ফাংশন প্রটোটাইপ বলে।

**** ৫। Function Argument বলতে কী বুঝ?** বাকাশিবো- ২০১৭, ২৩

উত্তর ঃ যখন কোনো ফাংশন কল করা হয় তখন ফাংশনের Parameter এর সংখ্যা তাদের Data Type এবং সংখ্যা অনুযায়ী Value পাস করা হয়। এই Value কে Function Argument বলে।

৬। Recursive Function কাকে বলে?

উত্তর ঃ যে Function নিজেই নিজেকে Call করে তাকে Recursive Function বলে এবং এই Call করার প্রক্রিয়াকে Recursion বলে।

*** ৭। ল্যাম্বডা (Lambda) ফাংশন কী?**

উত্তর ঃ পাইথনে Lambda Operator ব্যবহার করে এক লাইনের যে ফাংশন তৈরি করা হয়, তাকে ল্যাম্বডা ফাংশন বা অ্যানোনিমাস ফাংশন বলে।

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। ফাংশন ব্যবহারের সুবিধাগুলি লেখ।** বাকাশিবো- ২০১৮, ২০

উত্তর ঃ

১) একই কোড বার বার লেখার কোনো প্রয়োজন হয় না।
২) প্রোগ্রামের সাইজ ছোট হয়।
৩) প্রোগ্রাম লিখতে তুলনামূলক কম সময় লাগে।
৪) প্রোগ্রামের ভুল সহজেই সংশোধন করা যায়।
৫) Program কে ফাংশন হিসেবে লাইব্রেরীতে জমা করেও পরবর্তীতে তা ব্যবহার করা যায়।

***** ২। User Defined Function ব্যবহারের সুবিধাগুলি লেখ।**

উত্তর ঃ বাকাশিবো- ২০২১, ২৩

১) বড় Program কে ছোট ছোট মডিউলে ভাগ করা যায়।
২) এক Program এর ফাংশন অন্য Program এ ব্যবহার করা যায়।
৩) Program এর আকার ছোট হয়।
৪) Program লিখতে কম সময় লাগে।
৫) Program Execution এর সময় কম লাগে।
৬) Program এর ভুল ত্রুটি নির্ণয় সহজ হয়।
৭) Recursive Function ব্যবহার করে নিজে নিজেই Access করতে পারে।

***** ৩। Built-in Function ও User Defined Function এর মাঝে পার্থক্য লেখ।** বাকাশিবো- ২০২৪

উত্তর ঃ Built-in Function ও User Defined Function এর মাঝে নিম্নরূপ ঃ

Built-in Function	User Defined Function
১) যে সকল Function এর কাজ Developer কর্তৃক পূর্বে থেকেই নির্দিষ্ট কিছু কাজ সম্পন্ন করে, তাকে Built-in বা Library Function বলে।	১) ব্যবহারকারী তার প্রয়োজনে যে সকল Function তৈরি করে, তাকে User Defined Function বলে।
২) এই সকল Function এর নাম পরিবর্তন করা যায় না।	২) এই সকল Function এর নাম User তার ইচ্ছেমতো দিতে পারে।
৩) প্রতিটি প্রোগ্রামেই Built-in Function ব্যবহার করতে হয়।	৩) প্রতিটি প্রোগ্রামে User Defined Function ব্যবহার করা বাধ্যতামূলক নয়।
৪) এই Function ব্যবহারে প্রোগ্রামের আকার নির্দিষ্ট থাকে।	৪) এই Function এর ব্যবহারে প্রোগ্রামের আকার অপেক্ষাকৃত ছোট হয়।

৫) Function গুলি Compiler এ পূর্ব নির্ধারিত থাকে।	৫) Function গুলি Compiler এ পূর্ব নির্ধারিত থাকে না।
৬) **উদাহরণ ঃ** all( ), any( ), len( ), range( ), print( ) ইত্যাদি।	৬) **উদাহরণ ঃ** add( ), sub( ), mul( ), div( ) ইত্যাদি।

**** ৪। Pass by Value ও Pass by Reference বর্ণনা কর।**

বাকাশিবো- ২০২০, ২৩

উত্তর ঃ **Pass by Value ঃ** অন্য Variable এর সাথে ফাংশন Parameter এর Value কপি করার প্রক্রিয়াকে Pass by Value বলে। এই ক্ষেত্রে Value Pass করার সময়, ফাংশনের ভিতরের পরিবর্তনগুলো মূল Value তে কোনো পরিবর্তন হয় না। ফলে Value pass করে প্রকৃত Parameter এর একটি Copy তৈরি করে।

Pass by Reference ঃ Function এ প্রকৃত Parameter Pass করার প্রক্রিয়াকে Pass by Reference বলে। এর ফলে ফাংশনের ভিতরে করা পরিবর্তনগুলো মূল Value টিতে প্রতিফলিত হয়।

যখন ইমিউটেবল প্যারামিটার ফাংশনে Pass করা হয় তখন তা Pass by Value এর ন্যায় আচরণ করে। অন্যদিকে যখন মিউটেবল প্যারামিটার ফাংশনে Pass করা হয় তখন তা Pass by Reference এর ন্যায় আচরণ করে।

৫। Local Variable ও Global Variable এর মাঝে পার্থক্য লেখ।

উত্তর ঃ Local Variable ও Global Variable এর মাঝে পার্থক্য নিম্নরূপ ঃ

Local Variable	**Global Variable**
১) যে সকল Variable এর ব্যাপ্তি শুধুমাত্র কোনো নির্দিষ্ট ফাংশনে সীমাবদ্ধ, তাকে Local variable বলে।	১) যে সকল Variable এর ব্যাপ্তি ঐ Program এর সকল Function ব্যবহার করতে পারে, তাকে Global Variable বলে।
২) এই সকল Variable কে ঐ নির্দিষ্ট Function এর ভেতরে ঘোষণা করতে হয়।	২) এই সকল Variable কে Global Section এ ঘোষণা করতে হয়।
৩) এই Variable কে অন্য কোনো Function সরাসরি ব্যবহার করতে পারে না।	৩) এই Variable কে অন্য সকল Function ব্যবহার করতে পারে।
৪) দুই বা ততোধিক ফাংশনে একই নাম এবং Data Type এর local Variable ব্যবহার করা যায়।	৪) যেকোনো Function এই ধরনের Variable কে ব্যবহার করতে পারে।

SOS রচনামূলক প্রশ্নোত্তর ঃ

*** ১। বিভিন্ন প্রকার আরগুমেন্ট বর্ণনা কর।**

উত্তর ঃ Argument ঃ কোনো User defined function এর মাঝে Parenthesis( ) এর মাঝে কোনো Variable ঘোষণা করা হলে তাকে Parameter বলে। যখন কোনো ফাংশনকে Call করা হয় তখন ফাংশনের Parameter হিসেবে যে Value পাঠানো হয়, তাকে Argument বলে।

Function এ Argument কে চার ভাগে ভাগ করা যায়। যথা ঃ

১) Required Argument
২) Keyword Argument
৩) Default Argument এবং
৪) Variable Length Argument

১.Required Argument ঃ যে সকল Argument কোনো ফাংশনের সঠিক Positional Order এ Pass করা হয়, তাকে Required বা Positional Argument বলে।

যেমন ঃ

```
def  number(a, b, c):
     print (a, b, c)
number (10, 15, 20)    //function call
```

Output : 45

২.Keyword Argument ঃ যে সকল Argument সমূহ ফাংশনে Positional Order এ পাস না করে ফাংশনে তাদের

মান সরাসরি Pass করা হয়, তাকে Keyword Argument বলে।

যেমন ঃ

```
def  addition(a, b, c):
     return a+b+c
print (addition(a = 5, b = 10, c = 15))
```

Output : 30

৩. **Default Argument** ঃ প্রোগ্রামে Function বর্ণনার সময় তার কোনো Argument Variable এর সাথে সংগতিপূর্ণ মান নির্ধারণ করে দেওয়ার প্রক্রিয়াকে Default Argument বলে।

যেমন ঃ

```
def num(a, b = 10, c = 20):
     print (a, b, c)
num(10)    // function call
```

Output : 10, 10, 20

৪. **Variable Length Argument** ঃ এই ক্ষেত্রে ব্যবহারকারী যে কোনো সংখ্যক Argument নিয়ে কাজ করতে পারে।

যেমন ঃ

```
def num(*numbers):
     print (numbers)
num(10, 20, 30, 40)    //function call
```

Output : (10, 20, 30, 40)

অধ্যায়-২ ফাইল অপারেশন ইন পাইথন

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। ফাইল (File) কী?** বাকাশিবো- ২০১৮

উত্তর ঃ Secondary Storage Device যেমন Hard Disk, Floppy Disk, Compact Disk, Digital Video Disk ইত্যাদির এমন একটি Space, যেখানে কোনো Data স্থায়ী বা অস্থায়ীভাবে জমা রাখা যায়, তাকে File বলে।

*** ২। File Operation বলতে কী বুঝ?** বাকাশিবো- ২০২৪

উত্তর ঃ কোনো File এ Input দেওয়া এবং File হতে Output নেওয়ার জন্য যে সকল কার্যাবলি সম্পন্ন হয়, তাকে File Operation বলে।

***** ৩। input( ) ফাংশনের সিনট্যাক্স লেখ।** বাকাশিবো- ২০১৮, ১৯, ২০

উত্তর ঃ Syntax :

```
input([prompt])
```

prompt = সেই Value, যার মাধ্যমে Input গ্রহণ করা হয়।

উদাহরণ ঃ

```
input("Softmax Online School")
```

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

*** ১। File Operation এ ব্যবহৃত Function গুলোর কাজ লেখ।**

বাকাশিবো- ২০২৪

উত্তর ঃ File Operation এর বিভিন্ন কার্যাবলি সম্পাদনের জন্য বিভিন্ন Function ব্যবহৃত হয়।

Function	**কাজ**
close( )	Open কৃত File বন্ধ করতে।
read( )	File এর সকল Character পড়তে পারবে।
readlines( )	সম্পূর্ণ File কে Read করতে এবং File হতে একটি লাইন Return করে।
seek( )	File Pointer কে নির্দিষ্ট স্থানে নেওয়ার জন্য এই Function ব্যবহৃত হয়।
seekable( )	যদি File টি Random Access Support করে তবে True Return করে অন্যথায় False Return করে।
tell( )	File এর বর্তমান অবস্থান নির্দেশ করতে ব্যবহৃত হয়।
truncable( )	File এর Size নির্দিষ্ট করতে ব্যবহৃত হয়।
write( )	File এ কোনো কিছু লেখার জন্য এটি ব্যবহৃত হয়।
writable( )	যদি File এ Write করা যায় তবে True Return করে, অন্যথায় False Return করে।
writelines( )	File এ এক সেট লাইন Write করে।

***** ২। বিভিন্ন প্রকার File Access Mode এর ব্যবহার লেখ।**

উত্তর ঃ পাইথনে Open কৃত File এ বিভিন্ন Operation করার জন্য বিভিন্ন Mode এ File কে Open করতে হবে।

১) **Read Mode ঃ** পূর্বে তৈরিকৃত ফাইলে Read Operation সম্পন্ন করার জন্য এই Mode ব্যবহৃত হয়।

উদাহরণ ঃ

file = open("file_name.txt", "r")

এতে File Pointer টি File এর শুরুতেই অবস্থান করবে।

২) **Write Mode ঃ** File এ Write Operation সম্পন্ন করার জন্য এই Mode ব্যবহৃত হয়, তবে ফাইলে পূর্বের Text Overwrite হয়।

উদাহরণ ঃ

file = open("file_name.txt", "w")

৩) **Append Mode ঃ** পূর্বের তৈরিকৃত কোনো File এ নতুন কোনো Text যুক্ত করার জন্য এই Mode ব্যবহৃত হয়।

উদাহরণ ঃ

file = open("file_name.txt", "a")

৪) **Create Mode ঃ** নির্দিষ্ট File তৈরি করতে এবং ফাইলটি পূর্বনির্ধারিত থাকলে Error প্রদর্শন করবে।

উদাহরণ ঃ

file = open("file_name.txt", "x")

SOS রচনামূলক প্রশ্নোত্তর ঃ

***** ১। Opening ও Closing File বর্ণনা কর।**

অথবা, **পাইথন ভাষার ফাইল খোলা ও বন্ধ করার পদ্ধতি বর্ণনা কর।**

বাকাশিবো- ২০১০৯, ২০

উত্তর ঃ Opening File ঃ open( ) ফাংশন ঃ পাইথনে File Open করার জন্য এই Function ব্যবহৃত হয়।

Syntax :

file object = open(file_name, [access_mode], [buffering])

open( ) ফাংশনে ৩টি Parameter বিদ্যমান। যথা ঃ

১) **ফাইলের নাম ঃ** এটি open( ) ফাংশনের প্রথম প্যারামিটার। যদি পাইথন Script ও file কম্পিউটারে একই Directory তে থাকে, তবে শুধুমাত্র File এর নাম দিলেই হবে।

যেমন ঃ

file object = open("fil_name.txt")

আর যদি ভিন্ন Directory তে হয় তার সম্পূর্ণ Path ব্যবহার করতে হবে।

যেমন ঃ

file object = open("c:\user\Rubel\sos.txt")

২) **Access Mode ঃ** নতুন File তৈরি করতে হবে, নাকি পুরাতন File Operation সম্পন্ন হবে তা এই Parameter এর উপর নির্ভরশীল।

Access Mode হিসেবে ৩টি Mode ব্যবহৃত হয়।

i) Read Mode -"r"
ii) Write Mode –"w"
iii) Append Mode –"a"

৩) **Buffering** ঃ File Access করার সময় যদি Buffering Value 0 হয় তবে কোনো Buffering সংঘটিত হবে না। কিন্তু যদি Buffering Value 1 হয় হবে Buffering সংঘটিত হয়।

Closing File ঃ ফাইলে Read/Write Operation সম্পন্ন করার পর Open কৃত File অবশ্যই বন্ধ করতে হবে। তা না হলে File Open থাকার কারণে Memory এর অপচয় হবে এবং প্রোগ্রামের Performance তুলনামূলকভাবে কমবে।

ফাইল বন্ধ করার জন্য close( ) ফাংশন ব্যবহৃত হয়।

Syntax : fileName.close( )

অধ্যায়-৩ মডিউল, প্যাকেজ এবং অ্যাপ্লিকেশন সফটওয়্যার

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। Application Software বলতে কী বুঝায়?**

উত্তর ঃ Application Software এমন একটি Computer Program যা কোনো নির্দিষ্ট কার্য সম্পাদন করে। অর্থাৎ কোনো Application এর সাহায্যে নির্দিষ্ট কোনো কাজ করা জন্য যে Software ব্যবহার করা হয়, তাকে Application Software বা সংক্ষেপে Apps বলা হয়।

উদাহরণ ঃ Microsoft Word, PowerPoint, Firefox, Google Chrome, Media Player ইত্যাদি।

***** ২। মডিউল (Module) কী?**

উত্তর ঃ মডিউল হচ্ছে কিছু কোডের সমষ্টি যেখানে বেশ কিছু ফাংশন, Variable বা Data থাকে এবং যেগুলোকে Access করে প্রয়োজন অনুযায়ী অন্য পাইথন প্রোগ্রামে ব্যবহার করা যায়। ম্যাথেমেটিক্স সম্পর্কিত কাজের জন্য math মডিউল, ইনপুট-আউটপুট সম্পর্কিত কাজের জন্য sys মডিউল, ডেট ও টাইম সম্পর্কিত কাজের জন্য টাইম মডিউল ব্যবহার করা হয়।

***** ৩। প্যাকেজ (Package) বলতে কী বুঝায়?** বাকাশিবো- ২০২৪

উত্তর ঃ Package হলো module structure করার একটি উপায়। কতগুলো module বা program একত্রে করে একটি folder এ সংরক্ষণ করা হলে, তাকে Package বলে।

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। Module তৈরি করার Syntax ও উদাহরণ লেখ।**

উত্তর ঃ নতুন একটি প্রোগ্রাম এ মডিউল ব্যবহার করতে চাইলে তা প্রথমেই import করে নিতে হবে।

Syntax: import module_name

উদাহরণ ঃ import random

```
a = random.randint (1, 1000)
print (a)
```

উপরোক্ত Program এ a নামক variable এ একটি random Number store করতে চাচ্ছি। যে Random Number টি হবে 1 থেকে 1000 এর মধ্যে। Random Number তৈরির ফাংশন না লিখে built-in module, random কে import করে randint ফাংশনকে ব্যবহার করে random Number পাচ্ছি।

অথবা, from math import sqrt

```
print (sqrt (25))
```

উদাহরণ ঃ today = datetime.datetime.today()

print (today)

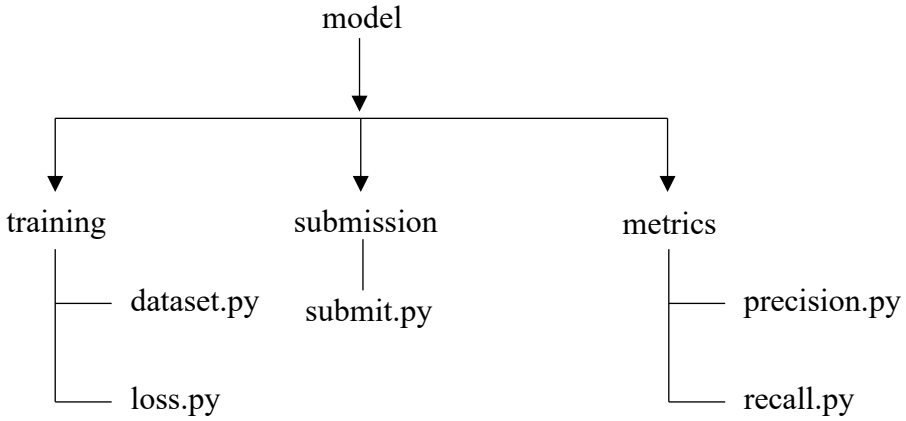
***** ২। Package তৈরির Syntax ও উদাহরণ লেখ।**

উত্তর ঃ import package Name.module_name

উদাহরণ ঃ i) import sound.effect.echo

ii) import sound.effect.reverse

iii) import sound.effect.equlaizer



উপরের উদাহরণে submit.py program ব্যবহার করতে চাইলে সম্পূর্ণ package কে import করতে পারি।

যেমন ঃ import model

আবার, চাইলে sub package ও import করতে পারি।

যেমন ঃ import model.submission.submit.py

অথবা, from model.submission import submit

অধ্যায়-৪ অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং এর মৌলিক ধারণা

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

*** ১। Instance Method কী?**

উত্তর ঃ যে Method একটি ক্লাসের Instance এর ভিতরে অবস্থান করে, তাকে Instance Method বলে।

Note: Instance Variable: যে সকল Variable একটি Class এর ভিতরে কিন্তু Constructor, Method অথবা Block এর বাহিরে Declare করা হয় তাদের Instance Variable বলে।

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। OOP এর বৈশিষ্ট্যগুলো কী কী?**

TSS-AP-2023, DUET: 11-12, BPDB-04, NTRCA-14, 16, 19, 20, BOF-18, 21, Uttara Bank-19, BCC-AP-19, বাণিজ্য মন্ত্রণালয়- ২১

উত্তর ঃ OOP এর বৈশিষ্ট্যগুলো নিচে উল্লেখ করা হলো ঃ

i) Class
ii) Object
iii) Data Abstraction
iv) Inheritance
v) Encapsulation
vi) Polymorphism

vii) Message Passing viii) Static binding
ix) Dynamic binding x) Open Recursion

***** ২। OOP এর সুবিধা ও অসুবিধাগুলো লেখ।**

উত্তর ঃ সুবিধা ঃ

i) বাস্তব ভিত্তিক সমস্যার সহজ সমাধান করা যায়।

ii) ভুল ত্রুটি নির্ণয় তুলনামূলকভাবে সহজ।

iii) Data Hiding সুবিধা বিদ্যমান।

iv) Security বেশি।

v) Inheritance এর কারণে পুরাতন Class এর পরিধি বাড়িয়ে নতুন Class তৈরি করা যায়।

vi) প্রয়োজনে Data বা function যুক্ত করা যায়।

vii) Multiple Programming Support করে।

viii) Networking Program করা যায়।

অসুবিধা ঃ

i) Program টেস্টিং, ডিবাগিং তুলনামূলকভাবে সহজ।

ii) এক method হতে অন্য method এ যোগাযোগের সময় বেশি।

iii) Program নির্বাহ হতে সময় বেশি লাগে।

iv) নতুন ব্যবহারকারীদের কাছে অপরিচিত এবং বুঝা কষ্টকর।

***** ৩। Class কী? পাইথনে Class তৈরির পদ্ধতি লেখ।**

উত্তর ঃ Class: Class হচ্ছে User defined data type. অর্থাৎ ব্যবহারকারী তার প্রয়োজনমত Class তৈরি করতে পারবে। একটি ক্লাসের ভিতরে Properties ও method থাকে।

ক্লাসের ভেতরের Variable কে বলা হয় member variable বা Properties এবং ফাংশনকে বলা হয় member function বা method বা behavior.

Class তৈরির Syntax বা Structure নিম্নরূপ ঃ

Syntax:

```
class ClassName:
<Statement 1>
<statement 2>
  . .
  . .
  . .
<statement N>
```

class কীওয়ার্ড ব্যবহার করে Class তৈরি করা হয়। এরপর Class এর একটি বৈধ নাম ব্যবহার করতে হয়। তারপর থাকে কোলন। একটি ক্লাসের মধ্যে বিভিন্ন Method (function) এবং Variable (Property) থাকতে পারে। যাদেরকে অবশ্যই Intented block এর মধ্যে লিখতে হয়।

**** ৬। Object কী ? পাইথনে Object তৈরির পদ্ধতি লেখ।**

উত্তর ঃ একটি Class এর Instance কে Object বলে। যা হচ্ছে Data এবং Method এর সমষ্টি। অর্থাৎ class এ ব্যবহৃত প্রতিটি Variable কে এক একটি Object বলে।

Syntax:

ObjectName = ClassName ()

উদাহরণ:

obj = Student ()

এখানে obj = object_Name

এবং Student = className

SOS রচনামূলক প্রশ্নোত্তর ঃ

***** ১। ক্লাস ব্যবহার করে দ্বিঘাত সমীকরণের মূল নির্ণয়ের প্রোগ্রাম লেখ।**

বাকাশিবো- ২০২৩

উত্তর ঃ

```
class quadratic:
    def __init__(self,a,b,c):
        import math
        d = (b*b) - (4*a*c)
        if(d>0):
```

```
            x1 = (-b +
math.sqrt(d))/(2*a)
            x2 = (-b -
math.sqrt(d))/(2*a)
            print('The roots are :
',x1,x2)
        elif(d==0):
            x = (-b)/(2*a)
            print('The root is : ',x)
        else:
            print('The roots are
imaginary: ')
a = int(input('Enter the value of a :
'))
b = int(input('Enter the value of b :
'))
c = int(input('Enter the value of c :
'))
q = quadratic(a,b,c)
```

*** ২। ক্লাস ব্যবহার করে তিনটি সংখ্যার মাঝে বড় সংখ্যা নির্ণয়ের প্রোগ্রাম লেখ। বাকাশিবো- ২০২৪

উত্তর ঃ

```
class largest:
    def __init__(self,a,b,c):
        if(a>b):
            if(a>c):
                print('a is largest')
        elif(b>a):
            if(b>c):
                print('b is largest')
            else:
                print('c is largest')
a = int(input('enter 1st number : '))
b = int(input('enter 2nd number : '))
c = int(input('enter 3rd number : '))
l = largest(a,b,c)
```

*** ৩। ক্লাস ব্যবহার করে তিনটি সংখ্যার মাঝে ছোট সংখ্যা নির্ণয়ের প্রোগ্রাম লেখ।

উত্তর ঃ

```
class triangle:
    def __init__(self,a,b,c):
```

```
        import math
        if((a+b)>c and (b+c)>a and
(a+c)>b):
            s = (a+b+c)/2
            area = math.sqrt(s*(s-
a)*(s-b)*(s-c))
            print('The area is :
%0.2f'%area)
        else:
            print('The triangle is
impossible')
a = int(input('Enter 1st value : '))
b = int(input('Enter 2nd value : '))
c = int(input('Enter 3rd value : '))
t = triangle(a,b,c)
```

অধ্যায়-৫ অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং এর ফোর পিলার

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। OOP এর ফোর পিলার বলতে কী বুঝ?**

উত্তর ঃ Polymorphism, Inheritance, Encapsulation, Data Abstraction হচ্ছে এর অন্যতম গুরুত্বপূর্ণ বৈশিষ্ট্য। এই বৈশিষ্ট্যগুলোকে একত্রে OOP এর ফোর পিলার বলা হয়।

***** ২। Polymorphism বলতে কী বুঝায়?**

DUET: 2015-16, NTRCA- 14, BB- 14, PSC- 17, বাকাশিবো- ২০২৩

উত্তরঃ Poly অর্থ বহু এবং morphism অর্থ রূপ। অর্থাৎ Polymorphism অর্থ বহুরূপ। ভিন্ন ভিন্ন অবস্থায় একই Object এর ভিন্ন ভিন্ন রূপ প্রদান করার প্রক্রিয়াকে Polymorphism বলে। যেমন ঃ Method Overloading, Method Overriding.

***** ৩। Inheritance বলতে কী বুঝায়?**

DUET: 07-08, Islami bank- 19, NTRCA- 14, PSC- 17, বাকাশিবো- ২০২৩

উত্তরঃ Inheritance এর বাংলা অর্থ উত্তরাধিকার সূত্রে পাওয়া। একটি ক্লাসের বৈশিষ্ট্য অন্য একটি ক্লাসে ব্যবহার করাকে Inheritance বলে।

ফলে পুরাতন ক্লাসের পরিধি বাড়িয়ে নতুন ক্লাস তৈরি করা যায়। পুরাতন ক্লাসকে Base Class বা Super Class এবং নতুন ক্লাসকে Sub Class বা Derived Class বলে।

***** ৪। Super Class কী?**

উত্তরঃ Python programming এ একটি ক্লাসের বৈশিষ্ট্য অন্য ক্লাসে ব্যবহার করাকে Inheritance বলে। যে ক্লাস এর বৈশিষ্ট্য শেয়ার করা হয়, সেই মূল ক্লাসকে Super বা Base বা Parent Class বলে।

***** ৫। Sub Class কী?**

উত্তরঃ Python programming এ একটি ক্লাসের বৈশিষ্ট্য অন্য ক্লাসে ব্যবহার করাকে Inheritance বলে। পুরাতন ক্লাসের বৈশিষ্ট্য শেয়ার করে যে নতুন ক্লাস তৈরি করা হয়, সে নতুন ক্লাস কে Sub বা Derived বা Child Class বলে।

**** ৬। Overridden Method কী?**

উত্তরঃ Base class এর Define কৃত কোনো Method কে নিয়ে Derived class এ কাজ করা হলে Base class এর ঐ Method কে Overridden Method বলে।

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। OOP এর ফোর পিলার সংক্ষেপে লেখ।** বাকাশিবো- ২০২৩

উত্তরঃ **Data Abstraction:** Data Abstraction হচ্ছে এক ধরনের Data Type বর্ণনা কৌশল, যেখানে কিছু Data থাকবে এবং সেই Data গুলোকে Access করার জন্য কিছু Method থাকে। কিন্তু ক্লাসের অভ্যন্তরে Data এবং Function বা Method কে ক্লাসের বাইরে হতে অবাঞ্চিত Access হতে সুরক্ষা প্রদান করে।

Encapsulation: Encapsulation এমন একটি প্রক্রিয়া যার মধ্যমে কোনো Object এর Data ও Method সমূহের বাইরের অবাঞ্চিত Access হতে রক্ষা করা যায় এবং Data ও Method সমূহকে একত্রিত করে Class তৈরি করা যায়।

Polymorphism: Poly অর্থ বহু এবং Morphism অর্থ রূপ। অর্থাৎ Polymorphism অর্থ বহুরূপ। ভিন্ন ভিন্ন অবস্থায় একই Object এর ভিন্ন ভিন্ন রূপ প্রদান করার প্রক্রিয়াকে Polymorphism বলে। যেমন ঃ Method Overloading, Method Overriding.

Inheritance: Inheritance এর বাংলা অর্থ উত্তরাধিকার সূত্রে পাওয়া। একটি ক্লাসের বৈশিষ্ট্য অন্য একটি ক্লাসে ব্যবহার করাকে Inheritance বলে। ফলে পুরাতন ক্লাসের পরিধি বাড়িয়ে নতুন ক্লাস তৈরি করা যায়। পুরাতন ক্লাসকে Base Class বা Super Class এবং নতুন ক্লাসকে Sub Class বা Derived Class বলে।

***** ২। Polymorphism বলতে কী বুঝ? সংক্ষেপে লেখ।**

Competition Commotion (P)- 19

উত্তরঃ Poly অর্থ বহু এবং Morphism অর্থ রূপ। অর্থাৎ Polymorphism অর্থ বহুরূপ। ভিন্ন ভিন্ন অবস্থায় একই Object এর ভিন্ন ভিন্ন রূপ প্রদান করার প্রক্রিয়াকে Polymorphism বলে। যেমন ঃ Method Overloading, Method Overriding.

উদাহরণ ঃ

```
i) a = 5      ii) a = "Softmax"
   b = 10         b = "Online"
   print (a+b)    c = "School"
                  print (a + " " + b " " + c)
Output: 15     Output: Softmax Online School
```

i) নং উদাহরণে + Operator দ্বারা দুইটি সংখ্যা যোগ করা হয়েছে।

ii) নং উদাহরণে + Operator দ্বারা তিনটি String কে Concatenate বা জোড়া লাগানো হয়েছে।

Polymorphism এর সংজ্ঞা থেকে আমরা জানি একই Object এর ভিন্ন ভিন্ন রূপ। যা "+" Operator দুই ক্ষেত্রে দুই রকমের বৈশিষ্ট্য প্রদর্শন করছে। ইহা Polymorphism এর উৎকৃষ্ট উদাহরণ।

SOS রচনামূলক প্রশ্নোত্তর ঃ

***** ১। OOP এর ফোর পিলার সম্পর্কে বর্ণনা কর।** বাকাশিবো- ২০২৩

উত্তরঃ Polymorphism, Inheritance, Encapsulation, Data Abstraction হচ্ছে OOP এর অন্যতম গুরুত্বপূর্ণ বৈশিষ্ট্য। এই বৈশিষ্ট্যগুলোকে একত্রে OOP এর ফোর পিলার বলা হয়। নিচে এদের বর্ণনা দেওয়া হল ঃ

Inheritance: Inheritance এর বাংলা অর্থ উত্তরাধিকার সূত্রে পাওয়া। একটি ক্লাসের বৈশিষ্ট্য অন্য একটি ক্লাসে ব্যবহার করাকে Inheritance বলে। ফলে পুরাতন ক্লাসের পরিধি বাড়িয়ে নতুন ক্লাস তৈরি করা যায়। পুরাতন ক্লাসকে Base class বা Super class এবং নতুন ক্লাসকে Sub class বা Derived class বলে। উদাহরণ ঃ

```
class Student:
 def info(self):
  print("Computer Technology")
class Result (Student):
 def gpa(self):
  print("gpa is 3.80")
r = Result ()
r.gpa ()
r.info ()

Output:
```

```
gpa is 3.80
Computer Technology
```

Polymorphism: Poly অর্থ বহু এবং Morphism অর্থ রূপ। অর্থাৎ Polymorphism অর্থ বহুরূপ। ভিন্ন ভিন্ন অবস্থায় একই Object এর ভিন্ন ভিন্ন রূপ প্রদান করার প্রক্রিয়াকে Polymorphism বলে। যেমন ঃ Method Overloading, Method Overriding.

উদাহরণ ঃ

i) a = 5 ii) a = "Softmax"
b = 10 b = "Online"
print (a+b) c = "School"
print (a + " " + b " " + c)

Output: 15 Output: Softmax Online School

i) নং উদাহরণে + Operator দ্বারা দুইট সংখ্যা যোগ করা হয়েছে।

ii) নং উদাহরণে + Operator দ্বারা তিনটি String কে Concatenate বা জোড়া লাগানো হয়েছে।

Polymorphism এর সংজ্ঞা থেকে আমরা জানি একই Object এর ভিন্ন ভিন্ন রূপ। যা + Operator দুই ক্ষেত্রে দুই রকমের বৈশিষ্ট্য প্রদর্শন করছে। ইহা Polymorphism এর উৎকৃষ্ট উদাহরণ।

Abstraction: Data Abstraction হচ্ছে এক ধরনের Data Type বর্ণনা কৌশল, যেখানে কিছু Data থাকবে এবং সেই Data গুলোকে Access করার জন্য কিছু Method থাকে। কিন্তু ক্লাসের

অভ্যন্তরে Data এবং Function বা Method কে ক্লাসের বাইরে হতে অবাঞ্চিত Access হতে সুরক্ষা প্রদান করে। উদাহরণ ঃ

```
i) def Add( ):
   a = int (input ("Enter First Number:"))
   b = int (input ("Enter Second Number:"))
   sum = a+b
   print ("Summation is : ", sum)
```

(i) নং file add.py নামে save করি।

```
ii) def Sub( ):
   a = int (input ("Enter First Number:"))
   b = int (input ("Enter Second Number:"))
   sub = a-b
   print ("Subtraction is : ", sub)
```

(ii) নং file sub.py নামে save করি।

```
iii) import add, sub
   add. Add( )
   sub. Sub( )
```

Abstraction কে দুই ভাবে Implement করা যায়।

i) Abstract class
ii) Interface

[**Note:** Abstract class: যে Class এ এক বা একাধিক Abstract থাকে, তাকে Abstract class বলে।]

Encapsulation: Encapsulation এমন একটি প্রক্রিয়া যার মাধ্যমে কোনো Object এর Data ও Method সমূহের বাইরের অবাঞ্চিত Access হতে রক্ষা করা যায় এবং Data ও Method সমূহকে একত্রিত করে Class তৈরি করা যায়।

উদাহরণ ঃ Python এ কোনো Variable অথবা Method create করলে তা Publicly বা Globaly declare হয়, অর্থাৎ কোনো একাটি Class এর ভিতরে Variable declare করলে তা Class এর বাহিরের যে কোনো জায়গা থেকেও Access করা যায়। যার ফলে Data এর Privacy থাকে না। Python এ Encapsulation এর মাধ্যমে Variable এর Access কে Control করা যায়।

Python এ ২ ভাবে করা যায়। i) Protected এবং ii) Private

i) Protected: Variable কে Protected মোড declare করলে শুধুমাত্র ঐ Class method এবং Object দ্বারা Variable কে Access করা যায়।

ii) Private: Variable কে Private মোড declare করলে শুধুমাত্র ঐ Class এর Method দ্বারা Variable কে Access করা যায়।

```
class A:
  _x = 5
  __y = 15   [Variable এর আগে Single underscore (_) দিলে Protected আর Double underscore (_,_) দিলে
                Private হয়।]
def Show (self):
```

print (“x = ”, self._x)
print (“y = ”, self.__y)
ob1 = A ( ) [Creating object of class A]
ob1. Show ( ).....x = 5
y = 15
print (“x in out of class = ”, obj. _x).....x in out of class = 5
print (“y in out of class = ”, obj. __y).....[A object has no attribute _,_y] Error massage
print (“x = ”, _x).....x is not defined
print (“y = ”, __y).....y is not defined

***** ২। Inheritance এর প্রকারভেদগুলো বর্ণনা কর।**

উত্তরঃ Inheritance এর বাংলা অর্থ উত্তরাধিকার সূত্রে পাওয়া। একটি ক্লাসের বৈশিষ্ট্য অন্য একটি ক্লাসে ব্যবহার করাকে Inheritance বলে। ফলে পুরাতন ক্লাসের পরিধি বাড়িয়ে নতুন ক্লাস তৈরি করা যায়। পুরাতন ক্লাসকে Base class বা Super class এবং নতুন ক্লাসকে Sub class বা Derived class বলে।

Inheritance কে পাঁচভাগে ভাগ করা হয়। যথা-

Single Inheritance: একটি মাত্র Parent class এর বৈশিষ্ট্য একটি মাত্র Child class এ Inherit করাকে Single Inheritance বলে।

Syntax:

```
class parent_class:
    # parent class statement
class child_class (parent_class):
    # child class statement
```

Multilevel Inheritance: যদি একটি Derived class এর আরও এক বা একাধিক Derived class থাকে, তাকে Multilevel Inheritance বলে।

Base class → Derived class (Intermediate class) → Derived class।

Syntax:

```
class parent_class1:
    # parent class1 statement
class parent_class2 (parent_class1):
    # parent_class statement
class child_class (parent_class2):
    # child class statement
```

Multiple Inheritance: একাধিক Base class এর বৈশিষ্ট্য যদি একটিমাত্র Derived class Inherit করে, তবে তাকে Multiple Inheritance বলে।

Syntax:

```
class parent_class1:
    # parent class1 statement
class parent_class2:
    # parent class2 statement
class child_class (parent_class1, parent_class2):
    # child class statement
```

Hierarchical Inheritance: একটি Base class হতে যদি একাধিক Derived class তৈরি করা হয়, তাকে Hierarchical Inheritance বলে।

Syntax:

```
class parent_class:
        # parent class statement
class child class_1 (parent_class):
    # child class_1 (parent_class):
class child class_2 (parent_class):
    # child class_2 (parent_class):
```

Hybrid Inheritance: যে পদ্ধতিতে দুই বা ততোধিক Inheritance পদ্ধতি ব্যবহার করে নতুন আরও একটি Inheritance তৈরি করা হয়, তাকে Hybrid Inheritance বলে।

অধ্যায়-৬ পাইথন ইটারেটর, জেনারেটর এবং ডেকোরেটর

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। ইটারেটর (Iterator) বলতে কী বুঝায়?** বাকাশিবো- ২০২৪

উত্তরঃ ইটারেটর এমন এক ধরনের Object যা একটি গণনাযোগ্য মান ধারণ করে। এটি List, Tuple, Set, Dictionary ইত্যাদি Object কে Iterate করতে ব্যবহৃত হয়। এ সকল Data Type Iterate করা যায় বলে, এদেরকে Iterable Object বলে।

***** ২। ডেকোরেটর (Decorator) কাকে বলে?** বাকাশিবো- ২০২৩

উত্তরঃ ডেকোরেটর সাধারণ ধরনেরই একটি ফাংশন যা অন্য একটি ফাংশনকে Modify করে অর্থাৎ তার কাজকে বর্ধিত বা পরিবর্তিত করে। যদি কখনো এমন প্রয়োজন হয় যে, একটি ফাংশনের ফাংশনালিটি একটু পরিবর্তন বা পরিবর্ধন করার প্রয়োজন, কিন্তু সেই ফাংশনের কোড আমরা পরিবর্তন করতে চাচ্ছিনা, তখন ডেকোরেটর ব্যবহার করে আমরা সেই কাজটি করতে পারবো।

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। ইটারেটর এর প্রসিডিউর কার্যনীতি সংক্ষেপে লেখ।**

উত্তরঃ ইটারেটর এমন এক ধরনের Object যা একটি গণনাযোগ্য মান ধারণ করে। এটি List, Tuple, Set, Dictionary ইত্যাদি Object কে করতে Iterate ব্যবহৃত হয়। এ সকল Data Type Iterate করা যায় বলে, এদেরকে Iterable Object বলে। ইটারেটরে দুইটি মেথড ব্যবহার করা হয়। যথা- ১) iter( ) - যদি আমরা কোনো ইটারেবল Object কে পাস করি তবে তা একটি ইটারেটর Return করে।

২) next( ) - একটি ইটারেটরের পরবর্তী Element Access করতে ব্যবহৃত হয়।

```
my_list = [4, 7, 0]
iterator = iter(my_list)
print(next(iterator))
print(next(iterator))
print(next(iterator))
```

***** ২। জেনারেটরের প্রসিডিউর কার্যনীতি সংক্ষেপে লেখ।**

উত্তরঃ জেনারেটর একটি ফাংশন। এই ফাংশনের কাজ হলো Yield statement ব্যবহার করে Sequence তৈরি করা। জেনারেটর ও এক ধারনের ইটারেটর। Yield keyword ব্যবহার করে সাধারণত প্রয়োজনীয়

জেনারেটর ফাংশন তৈরি করা হয়। এটি সাধারণত একটি নিয়মিত পাইথন ফাংশনকে জেনারেটরে রূপান্তর করতে ব্যবহৃত হয়।

```
def generatorFun( ):
    yield 1
    yield 2
    yield 3
for value in generatorFun( ):
    print(value)
```

***** ৩। ডেকোরেটরের প্রসিডিউর কার্যনীতি সংক্ষেপে লেখ।**

উত্তরঃ ডেকোরেটর সাধারণ ধরনেরই একটি ফাংশন যা অন্য একটি ফাংশনকে Modify করে অর্থাৎ তার কাজকে বর্ধিত বা পরিবর্তিত করে। যদি কখনো এমন প্রয়োজন হয় যে, একটি ফাংশনের ফাংশনালিটি একটু পরিবর্তন বা পরিবর্ধন করার প্রয়োজন, কিন্তু সেই ফাংশনের কোড আমরা পরিবর্তন করতে চাচ্ছিনা, তখন ডেকোরেটর ব্যবহার করে আমরা সেই কাজটি করতে পারবো।

```
def show(func):
    def inner( ):
        print("Python")
        func( )
    return inner
def display( ):
    print("Softmax Online School")
decorated_func = show(display)
```

decorated_func( )

SOS রচনামূলক প্রশ্নোত্তর ঃ

**** ১। ইটারেটর ব্যবহার করে Fibonacci series এ মান বের করার Python program লেখ।**

উত্তরঃ

```
def fib(limit):
    a, b = 0, 1
    while a < limit:
        yield a
    a, b = b, a + b
    x = fib(10)
    print(next(x))
    print(next(x))
    print(next(x))
    print(next(x))
    print(next(x))
```

অধ্যায়-৭ পাইথন-এর এক্সসেপশন এবং এরর হ্যান্ডিলিং

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। Bug ও Debug বলতে কী বুঝায়?** বাকাশিবো- ২০২৩, ২৪

উত্তরঃ **Bug:** Program এর যেকোনো ধরনের Error বা ত্রুটিকে Bug বলে।

Debug: Bug দূর করে Program কে সংশোধন করাকে Debug বলে।

***** ২। Syntax error বলতে কী বুঝায়?**

উত্তরঃ পাইথন Program এ গঠনগত সকল Error কে Syntax error বলে। যেমন ঃ কমা, কোলন, যেকোনো ব্রাকেট, ভুল Indentation ইত্যাদি সকল কিছুই Syntax error.

***** ৩। Runtime error বলতে কী বুঝায়?**

উত্তরঃ কোনো Program Run করানোর পরে যদি কোনো Error Message দেখায় তবে ঐ Error কে Runtime error বলে। Runtime error তখনই আসে যখন এমন কোনো Operation চালানো হয় যার বাস্তব কোনো সমাধান নেই।

যেমন ঃ কোনো সংখ্যাকে 0 দিয়ে ভাগ করা, user কর্তৃক ভুল ইনপুট।

**** ৪। Logical error বলতে কী বুঝায়?**

উত্তরঃ একটি program এর compilation এবং execution এর সময়, নির্দিষ্ট ইনপুট দেওয়া হলেও ক্ষেত্রবিশেষে অনাকাঙ্খিত Output পাওয়া যায়। এই ধরনের error গুলি যা ভুল Output প্রদান করে, তাকে Logical error বলে।

যেমন ঃ

```
x = int (input ("Enter a number"))
y = int (input ("Enter a number"))
z = x + y/2
print ("The average is : ", z)
```

এখানে z = (x + y)/2 হলে যে কাজটা করতে চাচ্ছি তা সঠিক হতো। কিন্তু z = x + y/2 এর কারণে ভুল Output প্রদান করবে। যা এক ধরনের Logical error.

**** ৫। বিল্ট-ইন এক্সসেপশনস বলতে কী বুঝায়?** বাকাশিবো- ২০২৩

উত্তরঃ পাইথনে Exception Handling করার জন্য যে সকল Exception Class গুলো Python developer কর্তৃক সংযুক্ত আছে, তাদেরকে Built-in exception বলে।

যেমন ঃ ZeroDivision error, EOFError, KeyError, Name Error, Runtime Error, Type Error, Value Error ইত্যাদি।

**** ৬। রেইসিং (Raising) কী?**

উত্তরঃ Raise কী-ওয়ার্ড Exceptions বা error গুলোকে উত্থাপন করতে ব্যবহৃত হয়। এটি Program এর Control Flow বন্ধ করে। এটি একটি Exception Handler.

যেমন ঃ

```
a = 5
if( a%2! = 0):
raise Exception ("The Number Should not be an Odd integer")
```

উপরের Program এ আমরা একটি পূর্ণসংখ্যা জোড় না বিজোড় তা চেক করব। পূর্ণসংখ্যা বিজোড় হলে Exception উত্থাপিত হবে।

**** ৭। Assert বলতে কী বুঝায়?**

উত্তরঃ পাইথনে কোনো Statement এর Condition সত্যকে মূল্যায়ন করলে কার্য সম্পাদন চালিয়ে যাওয়ার জন্য Assert Statement ব্যবহার করা হয়। যদি শর্ত মিথ্যা হয়, তবে নির্দিষ্ট ত্রুটি Assert error exception উত্থাপন হয়।

Syntax :

```
assert < condition >, < error message >
```

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। পাইথন এক্সসেপশন বলতে কী বুঝায়? উদাহরণসহ লেখ।**

উত্তরঃ **পাইথন এক্সসেপশন ঃ** এক্সসেপশন তখনই সংগঠিত হয়, যখন Program সার্বিকভাবে Run করে, কিন্তু Code Result এ আসে। তাই Error program এর Execution বন্ধ করে না, কিন্তু এর স্বাভাবিক প্রবাহের পরিবর্তন ঘটায়। ইহাকে Run time error ও বলে।

উদাহরণ ঃ

```
a = 5
b = 0
try:
   if (a<0 or b<0):
      raise ZeroDivisionError
   print(a/b)
except ZeroDivisionError:
   print("Enter Valid Value")
```

**** ২। পাইথন প্রোগ্রামে কী কী কারণে এক্সসেপশন তৈরি হয়?**

উত্তরঃ পাইথম প্রোগ্রামে এক্সসেপশন তৈরি হবার কারণ গুলো নিচে উল্লেখ করা হলো ঃ

১) Divided by zero (কোনো সংখ্যাকে 0 দ্বারা ভাগ করা)

২) এমন কোনো Identifier (Variable, Method ইত্যাদি) যা define করা হয়নি।

৩) এমন কোনো Operation চালানোর চেষ্টা করা, যা সম্ভব না। (যেমন ঃ Integer সংখ্যার সাথে String Concatenate করতে চাওয়া)

৪) Empty List, Dictionary ইত্যাদির Element Access করতে চাওয়া।

৫) তৈরি করা নেই এমন কোনো File Access করার চেষ্টা করা।

***** ৩। পাইথন এর try ও except exceptions সংক্ষেপে লেখ।**

বাকাশিবো- ২০২৪

উত্তরঃ নিচে পাইথন এর try ও except exceptions আলোচনা করা হলো ঃ

```
a = 5
b = 0
try:
    if (a<0 or b<0):
        raise ZeroDivisionError
    print(a/b)
except ZeroDivisionError:
    print("Enter Valid Value")
```

পাইথনে নির্দিষ্ট শর্তটি সত্য হলে কাজ চালিয়ে যাওয়ার জন্য assert statement ব্যবহৃত হয়।

আর যদি শর্তটি মিথ্যা হয়, তবে নির্দিষ্ট Message print করে Assertion Error Exception উত্থাপন করে।

***** ৪। পাইথনে try, except, else, finally এর syntax লেখ।**

উত্তরঃ **try** ঃ excepted error সংগঠিত হয়েছে কিনা তা পরিক্ষা করে।

except ঃ error Handle করতে ব্যবহৃত হয়।

else ঃ যদি কোনো Exception সংগঠিত না হয়, তবে, এই Block execute হয়।

Finally ঃ exception সংগঠিত হোক বা না হোক তবুও এই Block Execute হয়।

Syntax ঃ

```
try :
    # some code
except :
    # optional block
    # handling of exception (if required)
else :
    #execute if no exception
finally :
    # some code..... (always executed)
```

***** ৫। try ও except ব্যবহার করে পাইথনে program লেখ।**

বাকাশিবো- ২০২৩

উত্তরঃ

```
def divide(a, b):
    try:
        c = a / b
        print(" Result is :", c)
    except ZeroDivisionError:
        print("Sorry ! Zero Division
Error")
divide(7, 3)
divide(7, 0)
```

অধ্যায়-৮ পাইথন এর লগিং

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। পাইথন লগিং (Logging) কী?**

উত্তরঃ একটি Program যখন চলমান থাকে, তখন অনেকগুলো Event সংগঠিত হয়। এই Event গুলি Track করে Record রাখাই লগিং। Program চলার সময় এটি বিভিন্ন ধরনের কাজ করে, এতে বিভিন্ন ধরনের Error সংগঠিত হতে পারে যা Exception এর মাধ্যমে Handle করা হয়। এ সকল Event মেসেজ আকারে লগে লিখে রাখা হয়, যাতে প্রোগ্রাম Debug করতে সুবিধা হয়।

***** ২। লগিং ফাইল কাকে বলে?**

উত্তরঃ লগিং সংক্রান্ত সকল Information যে ফাইলে সংরক্ষণ করা হয়, তাকে লগিং ফাইল বলে। এই ফাইল এর মাধ্যমে Error নির্ণয় ও সমাধান করা যায়।

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। লগিং এর বিভিন্ন লেভেলগুলো উল্লেখ কর।** বাকাশিবো- ২০২৩, ২৪

উত্তরঃ পাইথন এ লগ মেসেজ প্রদানের জন্য পাঁচ ধরনের built-in লেভেল বিদ্যমান। যথা ঃ

১) **Debug:** সমস্যা নির্ণয়ের উপর ভিত্তি করে বিভিন্ন বার্তা বা Message Display করার জন্য ব্যবহৃত হয়।

২) **Info:** ব্যবহারকারীর প্রত্যাশা অনুযায়ী কাজ সঠিকভাবে সম্পাদন হচ্ছে কিনা, তা নিশ্চিত করতে এটি ব্যবহৃত হয়।

৩) **Warning:** এই Level টি নির্দেশ করে, অপ্রত্যাশিত কিছু ঘটেছে বা অদূর ভবিষ্যতে কিছু সমস্যা ঘটতে পারে।

৪) **Error:** এই Level দ্বারা বুঝায় Software টি কিছু কাজ সম্পাদন করতে পারেনি, এমন Message প্রদান করে।

৫) **Critical:** এই Level নির্দেশ করে যে Program টি নিজেই চলতে চলতে অক্ষম হতে পারে, এমন গুরুত্বপূর্ণ ত্রুটির বার্তা দেয়।

Level	Numeric value
NOTSET	0
DEBUG	10
INFO	20
WARNING	30
ERROR	40
CRITICAL	50

SOS রচনামূলক প্রশ্নোত্তর ঃ

**** ১। লগিং এর আউটপুট ফরমেটিং এবং কনফিগারেশন বর্ণনা কর।**

উত্তরঃ একটি ফাইলে Event রেকর্ড করতে Logging module ব্যবহার করা যায়। এজন্য Library হতে Logging module টিকে import করতে হয়।

১) প্রথমেই logger তৈরি এবং configure করতে হবে। এর অনেকগুলো Parameter বা Argument থাকতে পারে। যে File এ Event গুলো রেকর্ড করতে চাই, তার নামটি অবশ্যই Pass করতে হয়।

২) এখানে Logger এর Format ও সেট করা যায়। Default ভাবে File টি যদি append mode এ কাজ করে (a), তবে আমাদের প্রয়োজনে write mode (w) এ পরিবর্তন করতে পারি।

৩) logger এর level set করতে সরাসরি logger এর level গুলি ব্যবহার করা যায় অথবা নির্ধারিত Numeric Value ব্যবহার করতে পারি। প্যারামিটার হিসেবেও এই সকল Attribute ব্যবহার করা যেতে পারে।

৪) ব্যবহারকারী তার প্রয়োজনীয় Attribute অনুসারে প্রয়োজনীয় Attribute গুলো Python library হতে নির্বাচন করতে পারে।

```
import logging
logging.basicConfig(level=logging.DEBUG)
logging.debug('This will get logged')
```

Logging Configuration: logging এর প্রধান কাজ হচ্ছে ব্যবহারকারীর প্রয়োজনে একটি ফাইলে রেকর্ড ইভেন্টসমূহ সংরক্ষণ করা। Logging module basic config (**kwargs) method ব্যবহার করে logging configuration করা হয়।

basic config ( ) এর জন্য ব্যবহৃত প্যারামিটার ঃ

১) level: নির্দিষ্ট level set করতে ব্যবহৃত হয়।

২) filename: একটি ফাইল নির্দিষ্ট করে, যেখানে সকল event log হবে।

৩) filemode: file কে যে mode এ open করব। Default হিসেবে append (a) mode থাকে।

৪) format: log message এর format প্রদান করে।

```
import logging
logging.basicConfig(filename='basic.log
', filemode = 'w', format='%(process)d-
%(levelname)s-%(message)s')
logging.info('This will write in
basic.log')
logging.warning('This will also write
in basic.log')
logging.debug('This will not write in
basic.log')
```

অধ্যায়-৯ ইউনিট টেস্টিং

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। ইউনিট টেস্টিং (Unit Testing) কী?**

বাকাশিবো- ২০২৪

উত্তরঃ একটি Software code এর Unit হলো Function বা Method. এই ইউনিটকে টেস্ট করার প্রক্রিয়াকে ইউনিট টেস্টিং বলে। এই টেস্ট কোডগুলো লেখার জন্য বিভিন্ন ধরনের Frame work, Tools ইত্যাদি ব্যবহার করা হয়। যেমন ঃ Data flow Testing, Control flow Testing, Branch Coverage Testing, Statement Coverage Testing, Decision Coverage Testing ইত্যাদি।

[**Note:** Framework: Software তৈরিতে Framework হলো একটি সাহায্যকারী কাঠামো, যার উপর ভিত্তি করে অন্য Software তৈরি ও উন্নয়ন করা যায়। Software উন্নয়নকে সুবিধাদানের জন্যই মূলত তৈরি করা হয়।]

*** ২। ইউনিট টেস্টিং এর প্রয়োজনীয়তা কী?** বাকাশিবো- ২০২৩

উত্তরঃ Manual Testing এর ক্ষেত্রে বিশাল বড় বড় কোড বা বিভিন্ন ধরনের Function সমূহকে আলাদা আলাদাভাবে টেস্ট করা অনেকটা কষ্টসাধ্য এবং সময়সাপেক্ষ ব্যাপার। এই অসুবিধা দূর করে দক্ষতার সাথে

কোডসমূহকে টেস্ট করার জন্য ইউনিট টেস্টিং খুবই কার্যকরী একটি উপায়। এই ক্ষেত্রে বিভিন্ন Framework, Tools ইত্যাদি ব্যবহৃত হয়।

**** ৩। পাইথন বিল্ট-ইন ইউনিট টেস্ট ইউনিটগুলো নাম লেখ।**

উত্তরঃ পাইথন বিল্ট-ইন ইউনিট টেস্ট ইউনিটগুলো হলো ঃ unittest, unittest2, mock, nose, tox ইত্যাদি।

*** ৪। বহুল ব্যবহৃত Python module গুলোর নাম লেখ।**

উত্তরঃ unittest, DocTest, Pytest, Robot, Nose ইত্যাদি।

**** ৫। Unittest framework ব্যবহার করে Testing এর কী কী ফলাফল পাওয়া যায়?**

উত্তরঃ **i) Ok:** যদি সকল test pass করে, তবে Ok Return করে।

ii) Failure: যদি কোনো Test fail করে, তবে Assertion Error Exception Raise করে।

iii) Error: Assertion Error এর পরিবর্তে অন্য কোনো Error সংঘটিত হলে।

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। পাইথনে ইউনিট টেস্টিং এর গুরুত্ব সংক্ষেপে লেখ।**

উত্তরঃ পাইথনে ইউনিট টেস্টিং এর গুরুত্ব ঃ

i) কোড সেকশনকে আলাদা করা।

ii) প্রতিটি কোড সঠিক কিনা, তা যাচাই করা।

iii) কোডের সঠিকতা আলাদা আলাদাভাবে যাচাই করা।

iv) প্রতিটি মেথডকে আলাদা আলাদাভাবে পরিক্ষা করা।

v) Develop cycle এর প্রথম দিকে Bug গুলো সঠিক করতে।

vi) কোড দ্রুত পরিবর্তন করতে সক্ষম করতে।

vii) Code পুনরায় ব্যবহার করতে সাহায্য করে।

viii) অতি কম সময়ের মধ্যে Bug খুঁজে বের করা যায়।

ix) URL (Uniform Resource Locator) সংস্করণ করতে হবে কিনা তা সাথে সাথেই বুঝা যায়।

**** ২। ইউনিট টেস্টিং এর সুবিধাগুলো কী কী?**

উত্তরঃ i) ফাংশনটি কীভাবে কাজ করে তা যে কেউ কোড দেখার আগেই Unit test দেখে ধারণা করতে পারবে।

ii) Unit test এর case গুলো দেখলে বুঝা যায় কোনো বিশেষ case মিস হয়েছে কিনা, যদি মিস হয় তবে সেই Case যুক্ত করে test Run করলে সেই case এর জন্য ফাংশনটি ঠিকভাবে Output দিচ্ছে কিনা।

iii) কেউ যদি ফাংশনে কোনো পরিবর্তন করে তাহলে অনেক সময় ফাংশনে অনাকাঙ্খিত bug আসতে পারে। সেক্ষেত্রে যদি ফাংশনের Unit test ঠিকমত লেখা থাকে, তবে bug না আসার সম্ভাবনা থাকে। কারণ bug ঢুকলে Unit test fail করবে।

SOS রচনামূলক প্রশ্নোত্তর ঃ

***** ১। Unit testing এর Structure ব্যাখ্যা কর।** বাকাশিবো- ২০২৩

উত্তরঃ একটি Software code এর Unit হলো Function বা Method. এই ইউনিটকে টেস্ট করার প্রক্রিয়াকে ইউনিট টেস্টিং বলে।

Basic test structure : Unit test কে দু'ভাবে বর্ণনা করা যায়। যথা ঃ

১) কোড ব্যবহার করে ২) নিজেই test করে।

```
import unittest
class TestingSum(unittest.TestCase):
    def test_sum(self):
        self.assertEqual(sum([2, 3,
5]), 10, "It should be 10")
    def test_sum_tuple(self):
        self.assertEqual(sum((1, 3,
5)), 10, "It should be 10")
if __name__ == '__main__':
```

```
    unittest.main()
```

Unit testing এর সম্ভাব্য ফলাফল তিন ধরনের হতে পারে। যথা ঃ

i) Ok: যদি সকল test pass করে, তবে Ok Return করে।

ii) Failure: যদি কোন Test fail করে, তবে Assertion Error Exception Raise করে।

iii) Error: Assertion Error এর পরিবর্তে অন্য কোনো Error সংঘটিত হলে।

**** ২। পাইথন ইউনিট টেস্ট এর রিকোয়ারমেন্টস বর্ণনা কর।**

উত্তরঃ **Unit test:** Code test করার জন্য প্রয়োজনীয় সকল tools, unit test এ বিদ্যমান। এটি পাইথন এর Default unit testing framework, যা Standard library এর অন্তর্ভুক্ত। Unit test ব্যবহার করে কীভাবে প্রোগ্রাম লেখা, টেস্ট ও নির্বাহ করা হয় তা নিম্নে দেখানো হলো ঃ

```
def add(x,y):
    return x+y
def check_even(num):
    if(num%2==0):
        return True
    else:
```

```
        return False
```

উপরের ফাংশন দুইটি টেস্ট করার জন্য একই Directory তে test code লিখতে হবে।

```
import unittest
from test import add,check_even
class MyTest(unittest.TestCase):
    def test_add(self):
        self.assertEqual(add(4,5),9)
    def test_check_even(self):
```

উপরের প্রোগ্রামে add ( ) ফাংশন দুটি সংখ্যা যোগ করে। তাই test করে দেখেছি যে add ( ) ফাংশনে 4, 5 pass করলে Return কৃত মান 9 এর সমান হয় কিনা।

check_even( ) ফাংশনটি চেক করে দেখে যে, কোনো সংখ্যা জোড় কিনা? জোড় হলে True Return করবে, বিজোড় হলে False Reutrn করবে। check_even( ) ফাংশনে 4 কে pass করে দেখেছি তা True Return করে কিনা। এই পদ্ধতি ব্যবহার করে, ফাংশনের কোড test করা যায়।

অধ্যায়-১০

পাইথন রেগুলার এক্সপ্রেশন

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। পাইথন রেগুলার এক্সপ্রেশন (Regular Expression) কী?**

বাকাশিবো- ২০২৪

উত্তরঃ সহজ ভাষায় Regular Expression হলো ক্যারেক্টারের সিকুয়েন্স যা একটি Search pattern কে নির্দেশ করে। যে Special Character Sequence ব্যবহার করে অনেকগুলো String হতে নির্দিষ্ট কোনো Pattern String খুঁজে বের করা হয়, তাকে রেগুলার এক্সপ্রেশন বলে। ইহাকে সংক্ষেপে re বা regex বা regexp ও বলা হয়।

***** ২। মেটা ক্যারেক্টার (Meta Character) কী?** বাকাশিবো- ২০২৩

উত্তরঃRegular Expression এ কিছু Special Character আছে যেগুলো কিছু নির্দিষ্ট কাজে ব্যবহার করা হয়। যাদেরকে মেটা ক্যারেক্টার বলে। যেমন ঃ .(Dot), ^ (Caret), $ (Dollar), *(Asterisk), +(Plus), ? (Question mark) ইত্যাদি।

৩। Regex Special Sequencer কী?

উত্তরঃ একটি Backslash (\ ) এবং সাথে একটি ক্যারেক্টার ব্যবহার করে Regular Expression এর জন্য একটি Special Sequences তৈরি করা

হয়। এদেরকে Regex Special Sequences বলে। যেমন ঃ \A, \b, \d, \D ইত্যাদি।

SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

১। পাইথনে meta character এর তালিকা তৈরি কর।

উত্তরঃ পাইথনে meta character এর তালিকা ঃ

Meta character	কাজ
. (Dot)	যেকোনো character কে match করতে পারে। (newline ব্যতীত)
^ (caret)	String এর শুরুতে pattern match করে।
$ (Dollar)	String এর শেষে pattern match করে।
*(asterisk)	0 সংখ্যক বা যেকোনো সংখ্যক pattern match করতে ব্যবহৃত হয়।
+ (plus)	এক বা একাধিক সংখ্যক pattern match করতে ব্যবহৃত হয়।

? (Question)	0 অথবা 1 সংখ্যক pattern match করতে ব্যবহৃত হয়। কিন্তু একাধিক হলে match করবে না।
{ } (braces)	নির্দিষ্ট রেঞ্জের মধ্যে pattern match করতে ব্যবহৃত হয়।

*** ২। Regex special sequence এর ব্যবহার দেখাও।**

উত্তরঃ Regex special sequence এর ব্যবহার উল্লেখ করা হলো ঃ

Special Sequencer	Meaning
\ A	শুধুমাত্র string এর শুরুতে pattern match করতে।
\Z	String এর শেষে pattern match করতে।
\d	যেকোনো Digit match করে।
\D	যেকোনো Non-Digit Match করে।
\w	Alphanumeric character match করে।

\s	White space character match করে।
\S	Non white space character match করে।
\b	String এর শুরুতে শেষে pattern match করতে ব্যবহৃত হয়।
\B	String এর শুরুতে অথবা শেষে নয় এমন pattern match করতে ব্যবহৃত হয়।

[**Note:** White Space ঃ space, newline, return, tap]

৩। Reg Ex ব্যবহার করে নির্দিষ্ট ওয়ার্ড খোঁজা এবং গণনা করার program লেখ। বাকাশিবো- ২০২৩, ২৪

উত্তরঃ নিচে Reg Ex এর ব্যবহার করে Program টি লেখা হলো ঃ

```
import re
pattern = r"Softmax online School"
a = re.findall("o",pattern)
print(a.count("o"),"o found")
```

SOS রচনামূলক প্রশ্নোত্তর ঃ

**** ১। উদাহরণসহ পাইথন রেগুলার এক্সপ্রেশন এর বিল্ট-ইন মেথডগুলো বর্ণনা কর।** বাকাশিবো- ২০২৩

উত্তরঃ পাইথন রেগুলার এক্সপ্রেশন এর বিল্ট-ইন মেথডগুলো নিম্নরূপ ঃ

Method	Syntax	কাজ
match ( )	re. match (pattern, string)	কোনো string এর শুরুর অংশ match করতে ব্যবহৃত হয়।
search ( )	re. search (pattern, string)	কোনো string এর যেকোন জায়গা থেকে pattern match করার চেষ্টা করে।
findall ( )	re. findall (pattern, string)	substring এর list Return করবে।
split ( )	re. split (pattern, string)	string কে split ভেঙে আলাদা আলাদা করার জন্য ব্যবহৃত হয়।
sub ( )	re. sub (pattern, replacement, string)	এক বা একাধিক pattern কে string এ replace করতে ব্যবহৃত হয়।

উদাহরণ ঃ

match( ):

```
import re
pattern = r"s"
```

```
if(re.match("s","softmax online
school")):
    print("Matched")
else:
    print("Not Matched")
```

search( ):

```
import re
pattern = r"s"
if(re.search("s","softmax online
school")):
    print("Matched")
else:
    print("Not Matched")
```

findall( ):

```
import re
pattern = r"s"
print(re.findall(pattern,"softmax
online school"))
```

split( ):

```
import re
pattern = r"softmax online school"
a = re.split("\s",pattern)
print(a)
```

sub( ):

```
import re
pattern = r"softmax online school"
a = re.sub("s","S",pattern)
print(a)
```

২। RegEx ব্যবহার করে Email Address Verify করার program লেখ।

উত্তরঃ

```
import re
pattern = "[a-zA-Z0-9]+@[a-zA-
Z]+\.(com|net|in)"
def mail(email):
    if re.search(pattern, email):
        print("Valid Email")
    else:
        print("Invalid Email")
mail("rubelduet124019@gmail.com")
mail("sos@gmail.com")
mail("softmax@12.com")
```

অধ্যায়-১১ অ্যাপ্লিকেশন সফটওয়্যার

SOS অতি সংক্ষিপ্ত প্রশ্নোত্তর ঃ

***** ১। অ্যাপ্লিকেশন সফটওয়্যার (Application Software) বলতে কী বুঝায়?** বাকাশিবো- ২০২৪

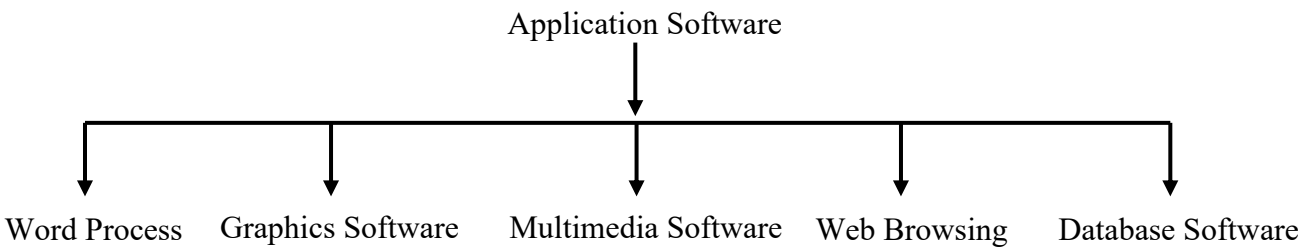
উত্তরঃ Application Software এমন একটি Computer Program যা কোনো নির্দিষ্ট কার্য সম্পাদন করে। অর্থাৎ কোনো Application এর সাহায্যে নির্দিষ্ট কোনো কাজ করা জন্য যে Software ব্যবহার করা হয়, তাকে Application Software বা সংক্ষেপে Apps বলা হয়।

উদাহরণ ঃ Microsoft Word, PowerPoint, Firefox, Google Chrome, Media Player ইত্যাদি।

২। Application Software এর প্রকারভেদ লেখ। বাকাশিবো- ২০২৩, ২৪

উত্তরঃ

Application Software এর প্রকারভেদ ঃ



SOS সংক্ষিপ্ত প্রশ্নোত্তর ঃ

১। Application Software এর ব্যবহার লেখ।

উত্তরঃ নিচে Application Software এর ব্যবহার উল্লেখ করা হলো ঃ

i. **Word Process:** word processing কাজের জন্য Application Software হিসেবে Microsoft Office word ব্যাপক জনপ্রিয়তার সাথে ব্যবহৃত হয়।

ii. **Graphics Software:** বিভিন্ন ধরনের ছবি, ভিত্তি, অ্যানিমেশন কার্টুন, ব্যানার, ভিজিটিং কার্ড, যেকোনো অনুষ্ঠানের কার্ড ইত্যাদি তৈরিতে Application Software হিসেবে Adobe Photoshop, Adobe Illustrator, CorelDRAW, Crello, GIMP, Blender, Paint shop pro, Autodesk Eagle, Sketch ইত্যাদি ব্যাপকভাবে ব্যবহৃত হচ্ছে।

iii. **Multimedia Software:** Multimedia বিভিন্ন কাজের জন্য Audio ও Video player হিসেবে VLC media player, DiVX, GOM media player, SM player, Windows media player ব্যাপকভাবে ব্যবহৃত হচ্ছে।

iv. **Web Brower:** E-mail আদান প্রদান, সামাজিক যোগাযোগ মাধ্যমে তথ্য আদান প্রদানসহ নানাবিধ কাজে বিভিন্ন ধরনের Web Browser ব্যবহৃত হয়। যেমন ঃ Google Chrome, Mozilla

Firefox, Opera, Internet Explorer, Safari, Netscape, Microsoft Edge, Brave, Vivaldi, Chromium.

v. **Database Software:** Database ও Account সম্পর্কিত বিভিন্ন কাজের জন্য Account management system, pay-roll-management system হিসেবে MS Excel, MySQL, Oracle ইত্যাদি ব্যাপকভাবে ব্যবহৃত হয়।

২। Application Software এর কার্যাবলি লেখ। বাকাশিবো- ২০২৩

উত্তরঃ Application Software প্রোগ্রামগুলি তথ্য পরিচালনা, ডেটা ম্যানিপুলেট, ভিজুয়াল নির্মাণসহ বিভিন্ন ধরনের ফাংশন সহজতর করার জন্য তৈরি করা হয়।

i. প্রতিষ্ঠানের তথ্য এবং ডাটা পরিচালনা করা।
ii. প্রতিষ্ঠানের নথিগুলো ম্যানেজ এবং Verify করা।
iii. শিক্ষামূলক উদ্দেশ্য।
iv. উপস্থাপনার উদ্দেশ্য Video Develop করা।
v. Accounting, Finance এবং Salary ব্যবস্থাপনা।
vi. Customer Relation Module (C R M)
vii. ছোট বা বড় আকারের প্রজেক্ট ব্যবস্থাপনা।
viii. Graphics, Animation ও Video product ইত্যাদি।

অধ্যায়-১২

Python Program

১. ফাংশন ব্যবহার করে কোনো সংখ্যা মৌলিক কি না তা নির্ণয়ের প্রোগ্রাম ঃ

```
def prime(num):
    if num> 1:
        for n in range(2,num):
            if (num % n) == 0:
                return False
        return True
    else:
        return False
print(prime(64))
print(prime(5))
```

২. ফাংশন ব্যবহার করে ত্রিভুজ ক্ষেত্রের ক্ষেত্রফল নির্ণয় করার প্রোগ্রাম ঃ

```
def triangle(a,b,c):
    s = (a + b + c) / 2
    area = (s * (s - a) * (s - b) * (s
- c)) ** 0.5
    print('The area of a traingle is
%0.2f' %area)
a = 20
```

```
b = 25
c = 35
triangle(a,b,c)
```

৩. ফাংশন ব্যবহার করে ১ থেকে ১০০ পর্যন্ত বিজোড় সংখ্যাগুলোর যোগফল নির্ণয় কর।

```
num = int(input("Print sum of even
numbers till : "))
sum = 0
for i in range(1, num + 1):
    if((i % 2) == 0):
        sum = sum + i
print("\nSum of even numbers from 1
to", num, "is :", sum)
```

৩. ক্লাস ব্যবহার করে ত্রিভুজের ক্ষেত্রফল নির্ণয়ের প্রোগ্রাম ঃ

```
def _init_(self,a,b,c):
    self.a = float(a)
    self.b = float(b)
    self.c = float(c)
a = input("a = ")
b = input("b = ")
c = input("c = ")
class triangle(a,b,c):
```

```
    def get_area(self):
        s = (a + b + c) / 2
        return (s*(s-a)*(s-b)*(s-c))**
0.5
t = triangle(a,b,c)
print("area : {}".format(t.area()))
```

৪. ইনহেরিটেন্স ব্যবহার করে প্রোগ্রাম লেখ।

```
class Student:
    name = ""
    def result(self):
        print("3.80")
class Name(Student):
    def display(self):
        print("My name is ", self.name)
std = Name()
std.name = "Rubel"
std.result()
std.display()
```

৫. জেনারেটর ব্যবহার করে ১ হতে ১০০ পর্যন্ত সংখ্যার মধ্যে বিজোড় সংখ্যা প্রদর্শন করার প্রোগ্রাম ঃ

```
def odd(n):
    for i in range(n):
        if(i%2!=0):
            yield i
display=odd(100)
for i in display:
    print(i)
```

৬. জেনারেটর ব্যবহার করে ১ হতে ১০০ পর্যন্ত সংখ্যার মধ্যে জোড় সংখ্যা প্রদর্শন করার প্রোগ্রাম ঃ

```
def even(n):
    for i in range(n):
        if(i%2==0):
            yield i
display=even(100)
for i in display:
    print(i)
```

বাংলাদেশ কারিগরি শিক্ষা বোর্ড, ঢাকা

ডিপ্লোমা ইন ইঞ্জিনিয়ারিং

৩য় পর্ব সমাপনী পরীক্ষা- ২০২২

টেকনোলজি ঃ কম্পিউটার সায়েন্স অ্যান্ড টেকনোলজি (২০২২ প্রবিধান)

বিষয় : অ্যাপ্লিকেশন ডেভেলপমেন্ট ইউজিং পাইথন

বিষয় কোড : ২৮৫৩১

সময় ঃ ৩ ঘণ্টা **পূর্ণমান ঃ ৬০**

ক ও খ বিভাগের সকল প্রশ্নের এবং গ-বিভাগ হতে যে-কোনো ৫ (পাঁচটি) টি প্রশ্নের উত্তর দাও।

ক-বিভাগ (মান ঃ ১× ১০ = ১০)

১। Python কোন Level-এর Programming language?

২। Function ও Function arguments কী?

৩। Method কী?

৪। Built in exception বলতে কী বুঝায়?

৫। Decorator কী?

৬। Polymorphism বলতে কী বুঝায়?

৭। Class-এর Syntax লেখ।

৮। Meta character কী?

৯। চারটি File operation-এর নাম লেখ।

১০। Destructor method কী?

খ-বিভাগ (মান ঃ ২× ১০ = ২০)

১১। Passing by reference কী?

১২। File operation-এর সুবিধা লেখ।

১৩। Module ও Package-এর দুটি পার্থক্য লেখ।

১৪। OOP-এর বৈশিষ্ট্য লেখ।

১৫। Four pillar-এর গুরুত্ব লেখ।

১৬। if ... else Syntax-এর গুরুত্ব লেখ।

১৭। error handaling কী?

১৮। Unit testing-এর Structure দেখাও।

১৯। Application software-এর প্রকারভেদ লেখ।

২০। লগিং-এর Level-গুলো উল্লেখ কর।

গ-বিভাগ (মান ঃ ৬× ৫ = ৩০)

২১। Function ব্যবহারপূর্বক List-এর ৫টি সংখ্যাকে যোগ করার Program লেখ।

২২। Log in-এর Configuration বর্ণনা কর।

২৩। Regex ব্যবহারপূর্বক নিম্নোক্ত String হতে Number-গুলোকে Extract করার জন্য Program লেখ ঃ

String = "hello 23, hi 77, lovely 18 and sweet 16"

২৪। OOP-এর Feature-গুলো আলোচনা কর।

২৫। Integrator, Generator ও Decorator সম্পর্কে বিস্তারিত আলোচনা কর।

২৬। Raising Exception ব্যবহারপূর্বক Python-এ একটি Program লেখ।

বাংলাদেশ কারিগরি শিক্ষা বোর্ড, ঢাকা

ডিপ্লোমা ইন ইঞ্জিনিয়ারিং

৩য় পর্ব সমাপনী পরীক্ষা- ২০২৩

টেকনোলজি ঃ কম্পিউটার সায়েন্স অ্যান্ড টেকনোলজি (২০২২ প্রবিধান)

বিষয় ঃ অ্যাপ্লিকেশন ডেভেলপমেন্ট ইউজিং পাইথন

বিষয় কোড : ২৮৫৩১

সময় ঃ ৩ ঘণ্টা পূর্ণমান ঃ ৬০

ক ও খ বিভাগের সকল প্রশ্নের এবং গ-বিভাগ হতে যে-কোনো ৫ (পাঁচটি) টি প্রশ্নের উত্তর দাও।

ক-বিভাগ (মান ঃ ১× ১০ = ১০)

১। Class বলতে কী বুঝায়?

২। Meta character বলতে কী বুঝায়?

৩। Iterator বলতে কী বুঝায়?

৪। Package বলতে কী বুঝায়?

৫। Bug কী?

৬। রেগুলার এক্সপ্রেশন কী?

৭। Unit testing বলতে কী বুঝায়?

৮। Logging-এর Level গুলোর নাম লেখ।

৯। Polymorphism বলতে কী বুঝায়?

১০। Inheritance বলতে কী বুঝায়?

খ-বিভাগ (মান ঃ ২× ১০ = ২০)

১১। Library function ও User function-এর মাঝে পার্থক্য লেখ।

১২। পাইথনে Unit testing-এর গুরুত্ব উল্লেখ কর।

১৩। Class ব্যবহার করে বৃত্তের ক্ষেত্রফল নির্ণয় করার Python program লেখ।

১৪। Try ও Except ব্যবহার করে একটি Python program লেখ।

১৫। Private ও Protected variable-এর মাঝে পার্থক্য লেখ।

১৬। Application software-এর কার্যাবলি লেখ।

১৭। Four pillars-এর গুরুত্ব লেখ।

১৮। Mat plot lib-এর কাজ লেখ।

১৯। Function ব্যবহারের সুবিধাগুলো লেখ।

২০। Abstraction কী? উদাহরণ দাও।

গ-বিভাগ (মান ঃ ৬× ৫ = ৩০)

২১। Class ব্যবহার করে তিনটি সংখ্যার মাঝে বড় সংখ্যা নির্ণয় করার Python program লেখ।

২২। Function ব্যবহার করে ধনাত্মক পূর্ণ সংখ্যা Factorial-এর মান নির্ণয় করার Program লেখ।

২৩। Polymorphism ব্যবহার করে Python program লেখ।

২৪। Super class ও Sub class ব্যবহার করে Python program লেখ।

২৫। Reg Ex ব্যবহার করে নির্দিষ্ট Word খোঁজা ও গণনা করার Python program লেখ।

২৬। উদাহরণসহ Decorators-এর কার্যনীতি বর্ণনা কর।

বাংলাদেশ কারিগরি শিক্ষা বোর্ড, ঢাকা

ডিপ্লোমা ইন ইঞ্জিনিয়ারিং

৩য় পর্ব পরিপূরক পরীক্ষা- ২০২৩

টেকনোলজি ঃ কম্পিউটার সায়েন্স অ্যান্ড টেকনোলজি (২০২২ প্রবিধান)

বিষয় ঃ অ্যাপ্লিকেশন ডেভেলপমেন্ট ইউজিং পাইথন

বিষয় কোড ঃ ২৮৫৩১

সময় ঃ ৩ ঘণ্টা **পূর্ণমান ঃ ৬০**

ক ও খ বিভাগের সকল প্রশ্নের এবং গ-বিভাগ হতে যে-কোনো ৫ (পাঁচটি) টি প্রশ্নের উত্তর দাও।

ক-বিভাগ (মান ঃ ১× ১০ = ১০)

১। ফাংশন বলতে কী বুঝায়?

২। পাইথন প্রোগ্রামে কী কী উপায়ে ইনপুট নেয়া হয়?

৩। ডেকোরেটর বলতে কী বুঝায়?

৪। Bug কী?

৫। ইনহেরিটেন্স বলতে কী বুঝায়?

৬। ক্লাস কী?

৭। Meta character কী?

৮। Built in exception বলতে কী বুঝায়?

৯। Pandas কী?

১০। Open ( ) ফাংশনের কাজ কী?

খ-বিভাগ (মান ঃ ২× ১০ = ২০)

১১। Module ও Package-এর মাঝে দুটি পার্থক্য লেখ।

১২। Four pillar-এর গুরুত্ব লেখ।

১৩। ইউজার ডিফাইন ফাংশন ব্যবহারের সুবিধা লেখ।

১৪। অ্যাবস্ট্রাকশন কী? সংক্ষেপে লেখ।

১৫। Recursive function-এর সুবিধা ও অসুবিধা লেখ।

১৬। লগিং-এর Level গুলো উল্লেখ কর।

১৭। Passing by reference কী?

১৮। লোকাল ভেরিয়েবল ও গ্লোবাল ভেরিয়েবল বলতে কী বুঝায়?

১৯। ইউনিট টেস্টিং-এর প্রয়োজনীয়তা কী?

২০। অ্যাপ্লিকেশন সফটওয়্যার-এর শ্রেণিবিভাগ দেখাও।

গ-বিভাগ (মান ঃ ৬× ৫ = ৩০)

২১। ফাংশন ব্যবহার করে ত্রিভুজের ক্ষেত্রফল নির্ণয়ের জন্য একটি প্রোগ্রাম লেখ।

২২। ক্লাস ব্যবহার করে দ্বিঘাত সমীকরণের মূল নির্ণয়ের জন্য একটি প্রোগ্রাম লেখ।

২৩। উদাহরণসহ পাইথন রেগুলার এক্সপ্রেশনগুলো বর্ণনা কর।

২৪। OOP-এর বৈশিষ্ট্যগুলো আলোচনা কর।

২৫। মডিউল তৈরির জন্য একটি পাইথন প্রোগ্রাম লেখ।

২৬। উদাহরণসহ এনক্যাপসুলেশন পদ্ধতি বর্ণনা কর।

বাংলাদেশ কারিগরি শিক্ষা বোর্ড, ঢাকা

ডিপ্লোমা ইন ইঞ্জিনিয়ারিং

৩য় পর্ব পরিপূরক পরীক্ষা- ২০২৪

টেকনোলজি ঃ কম্পিউটার সায়েন্স অ্যান্ড টেকনোলজি (২০২২ প্রবিধান)

বিষয় ঃ এপ্লিকেশন ডেভেলপমেন্ট ইউজিং জাভা

বিষয় কোড ঃ ২৮৫৩১

সময় ঃ ৩ ঘণ্টা **পূর্ণমান ঃ ৬০**

ক ও খ বিভাগের সকল প্রশ্নের এবং গ-বিভাগ হতে যে-কোনো ৫ (পাঁচটি) টি প্রশ্নের উত্তর দাও।

ক-বিভাগ (মান ঃ ১× ১০ = ১০)

১। User defined function বলতে কী বুঝায়?

২। Passing by reference বলতে কী বুঝায়?

৩। Python-এ Date time module-এর কাজ কী?

৪। Open ( ) function-এর Syntax লেখ।

৫। File operation বলতে কী বুঝায়।

৬। Pandas কী?

৭। Encapsulation-এর কাজ কী?

৮। Unit testing কী?

৯। Application software বলতে কী বুঝায়?

১০। চারটি লাইব্রেরি ফাংশনের নাম লেখ।

খ-বিভাগ (মান ঃ ২× ১০ = ২০)

১১। Function calling বলতে কী বুঝায়?
১২। User defined function ও Built in function-এর মাঝে পার্থক্য লেখ।
১৩। File operation সম্পর্কে সংক্ষেপে লেখ।
১৪। Matplotlib-এর কাজ লেখ।
১৫। Inheritance সম্পর্কে সংক্ষেপে লেখ।
১৬। OOP-এর 4-pillar-এর গুরুত্ব লেখ।
১৭। লগিং-এর বিভিন্ন লেভেলগুলো উল্লেখ কর।
১৮। Python-এর Try ও Except expressions সংক্ষেপে লেখ।
১৯। Application software-এর শ্রেণিবিভাগগুলো লেখ।
২০। ReEx. module methods-এ Optional flag গুলো লেখ।

গ-বিভাগ (মান ঃ ৬× ৫ = ৩০)

২১। Function কী? Function-এর শ্রেণিবিভাগ আলোচনা কর।
২২। একটি File-এ Data store করার Program লেখ।
২৩। উদাহরণসহ ডেকোরেটর-এর প্রসিডিউর কার্যনীতি বর্ণনা কর।
২৪। Application software-এর বৈশিষ্ট্য বর্ণনা কর।
২৫। Class ব্যবহারপূর্বক ত্রিভুজের ক্ষেত্রফল নির্ণয়ের Program লেখ।
২৬। Regular expression ব্যবহারপূর্বক নির্দিষ্ট Word খুঁজে বের করার Program লেখ।

❖ সমাপ্ত ❖