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)