

Welcome to my class

Presented by.....
Shajeda Yeasmin
Instructor(Computer)
Bangladesh Sweden Polytechnic Institute.
Kaptai. Rangmati.



Name of the Subject & code

**Python Programming
Subject Code: 28521**

অধ্যায়-২

পাইথনের প্রোগ্রামিং এর মৌলিক ধারণা (Basics of Python Programming)

প্রোগ্রামিং ল্যাংগুয়েজ হিসেবে পাইথন (Python as a programming language)

কম্পিউটার দিয়ে কাজ করাতে গেলে কম্পিউটার বুঝতে পারে, এমন ভাষায় তাকে নির্দেশ দিতে হয়। এটাই প্রোগ্রামিং ভাষা। প্রোগ্রামিং ভাষা অনেক রয়েছে এবং নিত্যনতুন তৈরি হচ্ছে। তবে অল্প কয়েকটি ভাষাই প্রোগ্রামারদের কাছে জনপ্রিয় হতে পেরেছে। তেমন একটি প্রোগ্রামিং ভাষা হচ্ছে পাইথন।

পাইথন (Python) একটি শক্তিশালী হাই-লেভেল এবং অবজেক্ট-ওরিয়েন্টেড প্রোগ্রামিং ভাষা। গুইডো ভ্যান রাসম এই প্রোগ্রামিং ভাষার স্রষ্টা।

এই প্রোগ্রামিং ভাষার গঠন(Syntax) শৈলী এবং ব্যবহার খুবই সহজ। ফলস্বরূপ, যদি কেউ জীবনে প্রথমবার প্রোগ্রামিং ভাষা শিখতে চাই তাহলে পাইথনই হবে তার জন্য সবচেয়ে পছন্দের ল্যাংগুয়েজ।

পাইথন একটি জনপ্রিয় ল্যাংগুয়েজ (Python as a popular language)

সারা পৃথিবীতে বর্তমানে জনপ্রিয়তার বিচারে পাইথনের স্থান চতুর্থ (শীর্ষ তিনটি হচ্ছে জাভা, সি, সি প্লাস প্লাস), আর যুক্তরাষ্ট্রের শীর্ষ বিশ্ববিদ্যালয়গুলোর ৬৯ শতাংশ শিক্ষার্থীরা প্রোগ্রামিংয়ের সঙ্গে পরিচিত হয় পাইথন ব্যবহার করে। এ ছাড়া বিশ্ববিখ্যাত প্রতিষ্ঠান গুগলের তিনটি আনুষ্ঠানিক প্রোগ্রামিং ভাষার একটি হচ্ছে পাইথন। পাইথনের এত জনপ্রিয় হওয়ার কারণ কী? পাইথন প্রোগ্রামাররা নিচের কারণগুলোকেই প্রধান মনে করেন:

- ☐ পাইথনে সংকেত (কোড) সহজে পড়া যায়।
- ☐ কোড সি বা জাভার চেয়ে তুলনামূলক অনেক ছোট হয়।
- ☐ পাইথনে রয়েছে লিস্ট, ডিকশনারি ও সেটের মতো চমৎকার ডেটা কাঠামো।
- ☐ পাইথন বিভিন্ন অপারেটিং সিস্টেমে ব্যবহার করা যায়।
- ☐ বিশাল ও কার্যকর স্ট্যান্ডার্ড লাইব্রেরি রয়েছে।
- ☐ শক্তিশালী অনলাইন কমিউনিটি। চমৎকার ওয়েব ফ্রেমওয়ার্ক (জ্যাংগো, ফ্লাস্ক ইত্যাদি)।

Uses of Python Language

পাইথন একটি বিস্ময়কর এবং শক্তিশালী ল্যাঙ্গুয়েজ তাই এর বহুমুখী ব্যবহার রয়েছে –

- ❑ ওয়েবভিত্তিক সফটওয়্যার নির্মাণে ।
- ❑ অটোমেশন সফটওয়্যার নির্মাণ।
- ❑ বায়ো ইনফর্মেশন।
- ❑ মেশিং লার্নিং ।
- ❑ ন্যাচারাল ল্যাঙ্গুয়েজ প্রসেসসিং ।
- ❑ ডাটাবেজ আপ্লিকেশনে ।
- ❑ সাইবার সিকিউরিটি ।
- ❑ ইন্টারনেট স্ক্রিপ্টিং।
- ❑ তথ্য বিশ্লেষণ ইত্যাদি ।

History of Python

গুইডো ভ্যান রাসম কর্তৃক সৃষ্ট পাইথন তুলনামূলক একটি পুরাতন ভাষা। ১৯৮০ দশক এর পরে তিনি সর্বপ্রথম পাইথন ডিজাইন শুরু করেছিলেন এবং ১৯৯১ সালের ফেব্রুয়ারিতে এটি প্রথম প্রকাশিত হয়।

- ১৯৮০ দশকের পরে গুইডো ভ্যান রাসম **আমিবা ডিস্ট্রিবিউটেড অপারেটিং সিস্টেম** গ্রুপের সাথে কাজ করছিলেন। তিনি **ABC**(ABC এর সিনট্যাক্স খুবই সহজ এবং বোধগম্য)-এর মত **ইন্টারপ্রেটেড(Interpreted)** ভাষা ব্যবহার করতে চাইলেন যা **আমিবা সিস্টেম কলকে** এক্সেস করতে পারবে। তাই তিনি নতুন একটি ভাষা তৈরির দিকান্ত নিলেন যেটা হবে সমপ্রসারনযোগ্য। এটাকে কেন্দ্র করেই নতুন একটি ভাষার সূত্রপাত হলো এবং পরবর্তীতে এটার নাম রাখা হলো **পাইথন**।
- ভয়ংকর কোন সাপের নামানুসারে পাইথন এর নামকরণ হয়নি। সত্তরের দশকের পরে রাসম একটি কমেডি সিরিজের ভক্ত ছিলেন। "মন্টি পাইথন" এর উড়ন্ত সার্কাস এর সাথে সমঞ্জস্য রেখেই এটার নাম রাখা হয়েছে পাইথন।

পাইথন এর বিভিন্ন ভার্সন এবং প্রকাশকাল

ভার্সন	প্রকাশকাল
পাইথন 1.0 (প্রথম স্টান্ডার্ড প্রকাশনী)	১৯৯৪ এর জানুয়ারি
পাইথন 1.6 (শেষ মাইনর ভার্সন)	২০০০ সালের ৫-এ সেপ্টেম্বর
পাইথন 2.0 (লিস্ট-list এর সূচনা)	২০০০ সালের ১৬ই অক্টোবর
পাইথন 2.7 (শেষ মাইনর ভার্সন)	২০১০ সালের ৩-এ জুলাই
পাইথন 3.0 (ডেপ্লিকেটেড কম্পট্রাক্ট এবং মডিউল রিমুভে জোর প্রদান)	২০০৮ সালের ৩-এ সেপ্টেম্বর
পাইথন 3.5 (নিরাপত্তা সংশোধন)	২০১৫ সালের ১৩-ই সেপ্টেম্বর
পাইথন 3.6 (শেষ আপডেট ভার্সন)	২০১৬ সালের ২৩-এ ডিসেম্বর

পাইথন এর বৈশিষ্ট্যসমূহ

- এটি একটি সাধারণ ভাষা যা শেখা খুবই সহজ। পাইথন-এর গঠন(syntax) শৈলী খুবই সহজ ও পরিচ্ছন্ন। অন্যান্য প্রোগ্রামিং ভাষা যেমন- C++, Java এবং C# ইত্যাদির তুলনায় পাইথন প্রোগ্রাম পড়া এবং লেখা উভয়ই সহজ।
- ফ্রি এবং ওপেনসোর্স- আপনি আপনার ব্যক্তিগত এবং এমনকি ব্যবসায়িক কাজেও মুক্তভাবে পাইথন ব্যবহার এবং বিতরণ করতে পারবেন। আপনি পাইথন দিয়ে লেখা সফটওয়্যার যে শুধু ব্যবহার এবং বিতরণ করতে পারবেন তা নয় বরং পাইথন এর সোর্স কোড-ও পরিবর্তন করতে পারবেন।
- বহনযোগ্যতা(Portability)- আপনি পাইথন প্রোগ্রামকে এক প্লাটফর্ম থেকে অন্য প্লাটফর্ম-এ নিতে পারবেন এবং কোনো ধরনের পরিবর্তন ছাড়াই রান করাতে পারবেন।
- এটি অধিকাংশ প্লাটফর্ম(যেমন- উইন্ডোজ, লিনাক্স এবং ম্যাক ওএস) এ নিজস্ব এপ্লিকেশনের মতই রান করে।

পাইথন এর বৈশিষ্ট্যসমূহ

- **সম্প্রসারণ এবং সংস্থাপনযোগ্য(Extensible and Embeddable)-** এটি আপনার এপ্লিকেশনকে উচ্চ ক্ষমতাসম্পন্ন করে তুলে এবং স্ক্রিপ্টিং সক্ষমতা প্রদান করে। অথচ অন্য প্রোগ্রামিং ভাষা এই সুবিধা প্রদান নাও করতে পারে।
- **হাই-লেভেল এবং ইন্টারপ্রেটেড ভাষা-** পাইথন কোড রান করাবেন তখন এটি স্বয়ংক্রিয়ভাবেই আপনার কোডকে কম্পিউটারের ভাষায় রূপান্তর করবে। এক্ষেত্রে নিম্ন-লেভেল অপারেশন নিয়েও আপনাকে উদ্বিগ্ন হতে হবে না।
- **সাধারণ সমস্যা সমাধানের জন্য অনেক বড় লাইব্রেরী-** পাইথন-এর কিছু নিজস্ব স্ট্যান্ডার্ড লাইব্রেরী আছে যার মাধ্যমে প্রোগ্রামিং জীবন সহজতর হয়ে উঠেছে। কারণ লাইব্রেরী ব্যবহার করলে এপ্লিকেশন এর সমস্ত কোড আপনারকে আর নতুন করে লিখতে হবে না।
- **অবজেক্ট ওরিয়েন্টেড(Object-oriented)-** পাইথনে সবকিছুই অবজেক্ট। অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ব্যবহার করে নিজস্ব জ্ঞান দ্বারাই জটিল জটিল সমস্যার সমাধান করতে পারবেন।

পাইথন প্রোগ্রাম

পাইথন প্রোগ্রামিং শেখা অনেক অনেক সহজ এবং "Hello, World!"
প্রোগ্রাম লেখার জন্য শুধুমাত্র-
`print("Hello, World!")` কোডটুকু লিখতে হয়।

পাইথন প্রোগ্রাম

```
# দুটি সংখ্যার যোগ  
num1 = 5  
num2 = 6  
sum = num1+num2  
print(sum)
```

আমরা উপরে একটি গাণিতিক অপারেশন করলাম, যেটি হচ্ছে যোগ (+) তেমনি আমরা বিয়োগ (-), গুণ (*) ও ভাগের (/) কাজও করতে পারি।

পাইথনের আইডেন্টিফায়ার ও কী-ওয়ার্ড সমূহ

আইডেন্টিফায়ার : পাইথনে ভেরিয়েবল, কন্সট্যান্ট, ক্লাস, মডিউল, , ফাংশন বা অন্যান্য সকল অজেক্ট আইডেন্টিফাই করার জন্য যে নাম বা শব্দ ব্যবহার করা হয় তাই পাইথন আইডেন্টিফায়ার।


আইডেন্টিফায়ার নামকরনের নিয়মঃ

- 1) আইডেন্টিফায়ারের নাম 'A' থেকে 'Z' বা 'a' থেকে 'z' অথবা '_' (Underscore) দিয়ে শুরু হয়। এবং এরপর যে কোনো অক্ষর কিংবা সংখ্যা (০-৯) পর্জন্ত ব্যবহার হয়।
- 2) Python এ Identifier হিসেবে @, \$, % ইত্যাদি বিশেষ চিহ্নগুলি ব্যবহার করা যায়না।
- 3) পাইথন কেইজ সেন্সিটিভ ল্যাঙ্গুয়েজ, অর্থাৎ A এবং a এখানে আলাদা অর্থ বহন করে। যেমন, Python Identifier-এ Computer এবং computer-এই দুটি শব্দ আলাদা অর্থ বহন করে এবং তারা ভিন্ন ভিন্ন আইডেন্টিফায়ার।
- 4) শুধুমাত্র Class Name গুলি বড় হাতের অক্ষর দিয়ে শুরু হয়, অন্য সবকিছুর নামকরণ শুরু হয় ছোট হাতের অক্ষর দিয়ে
- 5) কোন Identifier যদি শুধুমাত্র একটি Underscore (_) দিয়ে শুরু হয় তবে তার অর্থ হচ্ছে এটি একটি "Private Identifier".
- 6) যদি কোন Identifier পর পর দুটি Underscore (_) দিয়ে শুরু হয় তবে সেটি "Strongly Private Identifier".
- 7) যদি কোন Identifier পর পর দুটি Underscore (_) দিয়ে শেষ হয়, তবে তাকে "Language-Defined Special Name" বলে।

পাইথনে কী ওয়ার্ড

- ❑ কী ওয়ার্ড হলো কতগুলো সংরক্ষিত শব্দ, কম্পাইলারের নিকট যাদের বিশেষ অর্থ রয়েছে এবং যাদেরকে কোনো ভেরিয়েবল কনস্ট্যান্ট, ফাংশন মডিউল কিংবা অবজেক্টের নাম হিসেবে ব্যবহার করা যায় না তাদেরকেই কী ওয়ার্ড বলে।
- ❑ পাইথনে মোট ৩৩ টি কী ওয়ার্ড আছে----

কী-ওয়ার্ড	কী-ওয়ার্ড	কী-ওয়ার্ড
False	None	True
and	exec	not
as	finally	or
assert	for	pass
break	from	print
class	global	raise
continue	if	return
def	import	try
del	in	while
elif	is	with
else	lambda	yield
except		



এক সঙ্গে অনেকগুলো **character** বা এক বা একাধিক **word** এর সমষ্টিকে প্রোগ্রামিং এর পরিভাষায় **String** বলা হয়ে থাকে। পাইথনে **String** খুবই গুরুত্বপূর্ণ একটি ডেটা টাইপ। পাইথনে যে কোন **Sentence** কেই **String** হিসেবে ব্যবহার করা যায়।

পাইথনে তিন ভাবে **String** কে রিপ্রেজেন্ট করা যায়। সে গুলো হচ্ছে **Single Quotation** (‘ ‘), **Double Quotation** (” “) এবং **Three Quotation** (“”” “””) এর মাধ্যমে। আপনি যদি পাইথন কনসোলে নিচের মত করে বাক্য লিখে এন্টার প্রেস করেন তাহলে আউটপুটে সেই বাক্যকে দেখতে পাবেন।

পাইথন কোটেশন ও কমেন্টস

➡ পাইথন কোটেশন :

পাইথনে স্ট্রিং কে প্রকাশ করার জন্য তিন ধরনের উদ্ধৃতি চিহ্ন(quotation marks) ব্যবহার করা হয় -

১। সিঙ্গেল কোটেশন (' ') =>> ' Word '

২। ডাবল কোটেশন (" ") =>> "Sentence"

৩। ট্রিপল কোটেশন (" " " ") =>> " paragraph "

পাইথন কোটেশন :

```
>>> 'This is a String with Single Quotation'
```

```
'This is a String with Single Quotation'
```

```
>>> "This is a Another String with Double  
Quotation"
```

```
'This is a Another String with Double Quotation'
```

```
>>> """This is Extra String with Three Quotation"""
```

```
'This is Extra String with Three Quotation'
```

পাইথন কমেন্টস

- String literal এর বাইরে হ্যাশ চিহ্ন (#) টাইপ করে যদি কোন বাক্য লেখা হয়, তবে সেটি একটি 'কমেন্ট' হিসেবে গণ্য হবে। সহজ কথায়, # চিহ্নের পরে ব্যবহৃত একই লাইনের সকল অক্ষর একসাথে একটি কমেন্ট হিসেবে বিবেচিত হবে, এবং Python interpreter সব ধরনের কমেন্টকে ignore করবে। যেমন,

```
#!/usr/bin/python
```

```
# First comment
```

```
print ("Hello, Python!") # second comment
```

Output : Hello, Python!

For multiple comments

```
'''
```

```
DO NOT FORGET TO PROPERLY
```

```
INDENT THE STARTING OF STRING
```

```
LITERALS WITHIN YOUR CODE! '''
```

Command Line আর্গুমেন্ট

আমরা পাইথন প্রোগ্রাম রান করার সময় কিছু অপশন বা আর্গুমেন্ট পাস করতে পারি। এই অপশন বা আর্গুমেন্টগুলো আমরা ফাইল নেইমের পরে একটা একটা করে স্পেস দিয়ে আলাদা করে দিতে পারি। পাইথন এই আর্গুমেন্টগুলোকে sys মডিউলের argv লিস্ট জমা করে। একটি উদাহরন দেখে নেই:

main.py ফাইলের কন্টেন্ট:

```
import sys  
print sys.argv
```

টার্মিনালে রান করলাম এভাবে:

```
$ python main.py 1 2 3
```

এখানে 1, 2 এবং 3 প্রত্যেকে একেকটি কমান্ড লাইন আর্গুমেন্ট।

আউটপুট:

```
['main.py', '1', '2', '3']
```

sys.argv এর প্রথম আইটেমটি হয় ঐ ফাইল নেইম। এরপর কমান্ড লাইন আর্গুমেন্টগুলো থাকে একটার পর একটা।

ধন্যবাদ