

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-৩

পাইথনের ভেরিয়েবল ও ডাটা টাইপ (Variables & Data Types)

পাইথন ভ্যারিয়েবল ও ভ্যারিয়েবলের মান নির্ধারণ

পাইথন ভ্যারিয়েবল

ভ্যারিয়েল হচ্ছে কম্পিউটার মেমরির মধ্যে অবস্থিত একটি জায়গা/স্থান যা সচরাচর তথ্য(Data)/মান(value) সংরক্ষণের জন্য ব্যবহৃত হয়।

ভ্যারিয়েবলের কাজ হচ্ছে কম্পিউটার মেমরিতে ডেটা স্টোর করে রাখা, যেটা পড়ে প্রয়োজনমত ব্যবহার করা যাবে। ধরুন এক গ্লাস পানি আপনার সামনে রাখা আছে। এই গ্লাসটির কাজ হচ্ছে পানি স্টোর (store) করা। যেন প্রয়োজনের সময় গ্লাসের পানি ব্যবহার করা যায়। এই গ্লাসটিকে আমরা বলছি **ভ্যারিয়েবেল**। আর পানি হচ্ছে এই গ্লাস নামক ভ্যারিয়েবেলটির **ভ্যালু (value)**।

মেমরির মধ্যে অবস্থিত বিভিন্ন স্থানকে আলাদাভাবে বুঝানোর জন্য ভ্যারিয়েবলের নামসমূহ ইউনিক(একটি হতে অন্যটি ভিন্ন) হতে হবে। পাইথনে আইডেন্টিফায়ার যে নিয়মে লিখতে হয় ভ্যারিয়েবলের নামও ঐ একই নিয়মে লিখতে হয়।

যেমন: `number= 10`

`name="Shajeda"`

ভ্যারিয়েবলের মান নির্ধারণ

ভ্যারিয়েবলে ভ্যালু এসাইন করার জন্য আমরা এসাইনমেন্ট অপারেটর (=) ব্যবহার করি। যেকোনো বৈধ(valid) ভ্যারিয়েবলের মধ্যে যেকোনো টাইপের ভ্যালু রাখা যায়।

```
int_num= 10
```

```
float_num= 5.4
```

```
str_var= "SATT"
```

পাইথনে অনেক এসাইনমেন্ট একই স্টেটমেন্টে রাখা যেতে পারে। যেমন-

```
int_num, float_num, str_var= 10, 5.4, "SATT"
```

আমরা যদি একই সাথে বিভিন্ন ভ্যারিয়েবলে একই ভ্যালু এসাইন করতে চাই তাহলে উপরের উদাহরণটি অনুসরণ করতে পারি।

```
x = y = z = "Bangladesh"
```

খানে একই স্ট্রিং "Bangladesh" কে তিনটি ভিন্ন ভ্যারিয়েবলের মধ্যে এসাইন করা হয়েছে।

পাইথনে ভ্যারিয়েবলের নামকরণ এর নিয়মাবলী

- ❑ ভ্যারিয়েবলের নাম লেখার ক্ষেত্রে শুধুমাত্র ইংরেজি বড় হাতের এবং ছোট হাতের অক্ষর (A...Z, a...z), ডিজিটস (0,1,2...9) এবং underscore _ ব্যবহার করা যাবে। (যেমন, students, marks, firstName ইত্যাদি)
- ❑ তবে ভ্যারিয়েবলের নাম কখনোই সংখ্যা বা ডিজিট (যেমন, 2students) দিয়ে শুরু করা যাবে না।
- ❑ ভ্যারিয়েবলের নাম সবসময় এমন কিছু দেয়া উচিত, যেটা দিয়ে বোঝা যাবে যে কী নিয়ে কথা বলা হচ্ছে। যেমন: আমরা যদি স্টুডেন্টদের পরীক্ষার মার্কস নিয়ে কথা বলি, তাহলে শুধু marks না লিখে লেখা উচিত studentMarks
- ❑ আমরা যখন কন্সট্যান্টস নিয়ে কাজ করবো তখন কন্সট্যান্ট ভ্যারিয়েবলের নাম সবগুলোতে ইংরেজী বড় হাতের অক্ষর ব্যবহার করতে হবে। যেমন, PI, SPEED_OF_LIGHT, G ইত্যাদি ।

পাইথন ডাটা টাইপ

পাইথন প্রোগ্রামিং নিয়ে আমরা যখন ভ্যারিয়েবেলে কোন ভ্যালু অ্যাসাইন করে কাজ করছি, তখন এই সব ভ্যালুই কিন্তু একেকটি ডেটা টাইপ। আমাদের আরও মনে রাখতে হবে যে পাইথন অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ল্যাগুয়েজ। এখানে প্রত্যেকটি ভ্যারিয়েবলই হচ্ছে আসলে এক একটি অবজেক্ট (Object) আর ডেটা টাইপ হচ্ছে এর ক্লাস (Class)।

৫ রকমের ডেটা নিয়ে পড়াশোনা করবো।

1. Number
2. List
3. Tuple
4. String
5. Dictionary

পাইথন নাম্বার (Number)

integer (পূর্ণ সংখ্যা), floating point numbers (দশমিক সংখ্যা), complex numbers (জটিল সংখ্যা) এই সবগুলোই Python Numbers এর ক্যাটাগরিতে পড়ে।

- ❑ `x = 1` # এটি একটি পূর্ণ সংখ্যা
- ❑ `y = 2.8` # এটি একটি দশমিক সংখ্যা
- ❑ `z = 1j` # এটি একটি জটিল সংখ্যা

পাইথন লিস্ট (List)

বাজার করার জন্য আমরা অনেক সময় লিস্ট তৈরী করি। যেখানে লিস্ট আমরা যে আইটেমগুলো বাজার থেকে কিনবো সেগুলোর একটি অর্ডার অনুসারে 1, 2, 3... করে সাজানো থাকে।

পাইথনে একইভাবে আমরা সিকোয়েন্স অনুসারে বিভিন্ন আইটেমের লিস্ট সাজিয়ে রাখতে পারি। এটার জন্য আমরা `[]` ব্যবহার করি, যেটার ভেতরে আইটেমগুলো আমরা কমা , দিয়ে আলাদা করে রাখি। নিচে এরকম একটি উদাহরণ দেয়া হলো,

```
shoppingList = ["apple", "banana", "mango"]
```

```
print(shoppingList)
```

```
Output : ['apple', 'banana', 'mango']
```

পাইথন টাপল (Python Tuple)

- ❑ পাইথন টাপল আসলে লিস্টের মতোই। পার্থক্য হচ্ছে টাপলে ইন্ডেক্সের ভ্যালু পরিবর্তন করা যায় না।
- ❑ টাপল ক্রিয়েট করতে আমরা লিস্টের যেখানে আগে `[]` ব্যবহার করতাম, সেখানে `()` ব্যবহার করতে হবে। এরপর আইটেমগুলো কমা , দিয়ে আলাদা করে দিতে হবে।

```
sampleTuple = ("apple", "banana", "mango")
```

```
print(sampleTuple)
```

```
Output:('apple', 'banana', 'mango')
```

পাইথন স্ট্রিং (Python Strings)

- আমরা ইউনিকোড ক্যারেক্টার প্রকাশ করার জন্য পাইথনের স্ট্রিং ব্যবহার করি, এক্ষেত্রে ডবল কোটেশন মার্ক বা সিঙ্গেল কোটেশন মার্ক ব্যবহার করে আমরা স্ট্রিং টাইপ ডেটা প্রকাশ করি। মাল্টি-লাইন স্ট্রিং এর জন্য আমরা ট্রিপল কোটেশন মার্ক ব্যবহার করতে পারি।

```
sample1 = "This is a string"
```

```
print(sample1)
```

```
sample2 = """Here is a multiline  
string"""
```

```
print(sample2)
```

```
output:This is a string
```

```
Here is a multiline
```

```
string
```

পাইথন ডিকশনারি (Python Dictionary)

আমরা যখন প্রচুর পরিমাণ ডেটা নিয়ে কাজ করি তখন ডিকশনারি ব্যবহৃত হয়। ডিকশনারি হচ্ছে আইটেমসমূহের key এবং value এর একটি কালেকশন। প্রত্যেকটি আইটেমের একটি করে key এবং value থাকে। এবং আইটেমগুলো সব একটি সেকেন্ড ব্র্যাকেটের { } মধ্যে রাখা হয়। ধরা যাক আমাদের কাছে 2 টি ফল আছে।

apple আছে 2 টি

mango আছে 4 টি

এখন আমরা চাচ্ছি এটাকে পাইথনের সাহায্যে কম্পিউটার মেমরিতে স্টোর করতে। এটা আমরা ডিকশনারি ব্যবহার করে খুব সহজেই করে ফেলতে পারি।

```
d = {'apple':2,'mango':4}
```

```
print(d)
```

পাইথন ডেটা টাইপ কনভার্সন

- ❑ ডাটাটাইপ কনভার্সন বলতে ভ্যারিয়েবল কে এক টাইপ থেকে অন্য টাইপ এ কনভার্ট করা বুঝায়। একে টাইপ কাস্টিং ও বলা হয়ে থাকে। পাইথনে টাইপ কাস্টিং এর জন্যে কিছু বিল্টইন ফাংশন বানানো আছে। আমরা চাইলে সহজেই সেগুলো ব্যবহার করতে পারি। এখন পর্যন্ত আমরা integers, floats, এবং strings ডাটাটাইপ সম্পর্কে জেনেছি। এই টাইপে কনভার্ট করার জন্য ফাংশন গুলো যথাক্রমে হচ্ছে - int(), float(), str().
- ❑ ইন্টজার এ কনভার্সন
- ❑ ফ্লোট এ কনভার্সন
- ❑ স্ট্রিং এ কনভার্সন

ইন্টজার এ কনভার্সন

- ❑ স্ট্রিং অথবা ফ্লোট থেকে ইন্টজার এ কনভার্ট করার জন্য int() ফাংশন ব্যবহার করা হয়।

String to Integer Conversion

```
>>> int("123")
```

```
123
```

float to Integer Conversion

```
>>> int(12.3)
```

```
12
```

বিঃ দ্রঃ স্ট্রিং থেকে ইন্টজার এ কনভার্ট এর সময় খেয়াল রাখতে হবে স্ট্রিং এ যাতে কোনো নননিউমেরিক ক্যারেকটার না থাকে।

ফ্লোট এ কনভার্সন

- ❑ স্ট্রিং অথবা ইন্টেজার থেকে ফ্লোট এ কনভার্ট করার জন্য float() ফাংশন ব্যবহার করা হয়।

String to float Conversion

```
>>> float("123.456")
```

```
123.456
```

Integer to float Conversion

```
>>> float(123)
```

```
123.0
```

বিঃ দ্রঃ এক্সট্রেণ্ড স্ট্রিং থেকে ফ্লোট এ কনভার্ট এর সময় খেয়াল রাখতে হবে স্ট্রিং এ যাতে কোনো নননিউমেরিক ক্যারেকটার না থাকে এবং একাধিক দশমিক পয়েন্ট না থাকে।

স্ট্রিং এ কনভার্সন

- যে কোন ভ্যারিয়েবল স্ট্রিং -এ কনভার্ট করার জন্য কোনো প্রকার বিধিনিষেধ ছাড়াই `str()` ফাংশন ব্যবহার করবো।

```
>>> str(123)
```

```
Output--'123'
```

ধন্যবাদ