

৩.১ ভেরিয়েবল ও মান নির্ধারণ

- ভেরিয়েবল হচ্ছে মেমোরি তে নির্ধারিত জায়গা (স্পেস) যেখানে বিভিন্ন জিনিসের মান (value) স্টোর করে রাখা হয়। ভেরিয়েবল তৈরি করা মানেই কম্পিউটারের মেমোরিতে একটা নির্দিষ্ট স্পেস সঞ্চয় করে রাখা। যেকোনো ভেরিয়েবল কি ধরনের ডাটা রেকর্ড/ স্টোর করবে সেটা নির্ভর করে ভেরিয়েবল উপর।
- Python এ সমতা চিহ্নের (=) মাধ্যমে যেকোনো ধরনের মান (value) ভেরিয়েবল হিসেবে স্টোর করা যায়, এক্ষেত্রে আলাদা করে কোন ডিক্লারেশনের প্রয়োজন নেই। সমতা চিহ্নের বাম পাশে ব্যবহৃত শব্দটি ভেরিয়েবলের নাম এবং ডান পাশের সংখ্যাটি ভেরিয়েবলের মান নির্দেশ করে। যেমন,

number = 100



ভেরিয়েবলের নাম



ভেরিয়েবলের মান

৩.১ ভেরিয়েবল ও মান নির্ধারণ

□ পাইথনে ভ্যারিয়েবল লেখার সময় কিছু নিয়ম মেনে ভ্যারিয়েবল ডিফাইন করা হয়।

✓ ভ্যারিয়েবলের নাম অবশ্যই এক শব্দের হতে হবে। অর্থাৎ এরকম ভাবে ভ্যারিয়েবল লেখা যাবে না।

যেমন- `my variable = 10`

✓ প্রথম অক্ষর অবশ্যই একটি alphabetic letter(uppercase or lowercase) অথবা underscore (`_`) হতে হবে । যেমন: `nafis`, `a`, `b`, `_variable` লেখা যাবে ভ্যারিয়েবল হিসেবে কিন্তু `1name`, `@nafis`, `7a`, `%b` এই ভাবে লেখা যাবে না । প্রথম অক্ষর ছাড়া পরে letter, underscore, number ব্যবহার করা যাবে। যেমন: `variable1`, `my_variable`, `not_Very_Good_Name10` যদিও ভ্যারিয়েবলের শুরুতে underscore ব্যবহার করা যায়, কিন্তু পাইথনের কনভেনশন হচ্ছে ভ্যারিয়েবলের নাম সবসময় lowercase letter দিয়ে শুরু করা।

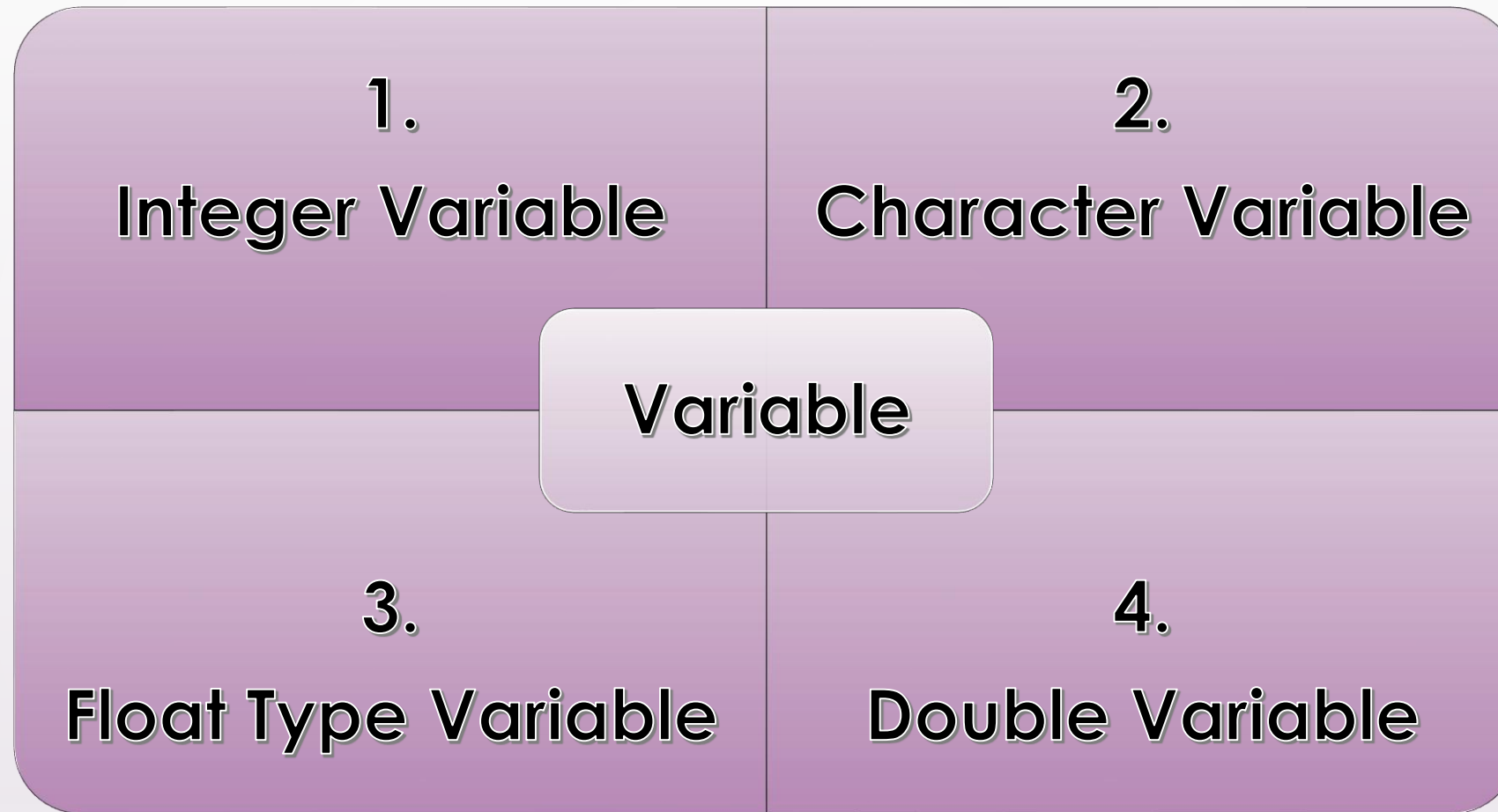
✓ পাইথন Case Sensitive অর্থাৎ `a = 4` এবং `A = 4` একই ভ্যারিয়েবল না।

✓ পাইথনের কিছু reserved কী-ওয়ার্ড আছে, এগুলো ব্যবহার করা যাবে না । যেমন: `if`, `else`, `elif`, `for`, `while`, `break`, `continue`, `del`, ইত্যাদি ।

৩.১ ভেরিয়েবল ও মান নির্ধারণ

উদাহরণ,

- `>>> this_is_a_normal_name = 7`
- `>>> 123abc = 7`
- `SyntaxError: invalid syntax`
- `>>> spaces are not allowed`
- `SyntaxError: invalid syntax`



৩.১ ভেরিয়েবল ও মান নির্ধারণ

লোকাল ভেরিয়েবল

গ্লোবাল ভেরিয়েবল