

# স্ট্যাটিক

ভেরিয়েবল বানাতে চাইলে কোনো ভেরিয়েবলের সামনে static শব্দটি লিখলেই হবে। ফাংশনে স্ট্যাটিক ভেরিয়েবল ব্যবহার করা হয়। ফাংশনের হেডারে প্যারামিটার থাকলে সেটাও ভেরিয়েবল। যখন ফাংশনটি এক্সিকিউশন শেষ হবে, তখন ভেরিয়েবলগুলো শেষ হয়ে যায় বা ধ্বংস হয়ে যায়। স্ট্যাটিক ভেরিয়েবলের বেলায় এমনটা হয় না। ফাংশনের এক্সিকিউশন শেষ হয়ে গেলেও এ ভেরিয়েবলের মান ধরে রাখা হয় এবং পরে সে ফাংশনকে আবার কল করা হলে ধরে রাখা মানটিকে পাঠিয়ে দেয়। যেমন—

```
<?php
function test_stat() {
static $sekta_variable = 0;
$sekta_variable++;
echo $sekta_variable;
echo "<br />";
}
test_stat();
test_stat();
test_stat();
?>
```

## ব্যাখ্যা

ফাংশনে প্রথমে \$sekta\_variable ভেরিয়েবলের মান ছিল 0 আর \$sekta\_variable++ দিয়ে এই মান 1 বৃদ্ধি পেল। তাই প্রথম ফাংশনটা call করতে আউটপুট দিল 1। এখন \$sekta\_variable-এর মান 0 থেকে হয়ে গেল 1। এরপর দ্বিতীয়বার ফাংশনটিকে call করতে আউটপুট দিল 2। এভাবে বাকিগুলো।

যদি static শব্দটি উঠিয়ে দেয়া হয়, তাহলে আউটপুট আসবে

```
1
1
1
```

আর স্ট্যাটিক ভেরিয়েবল ব্যবহারের কারণে এখন আউটপুট আসবে

```
1
2
3
```

## সুপারগ্লোবাল ভেরিয়েবল

### কনস্ট্যান্ট

কোডে ভেরিয়েবলের মান বিভিন্ন সময় ভিন্ন ভিন্ন হতে পারে বা পরিবর্তন হতে পারে। এ ক্ষেত্রে কনস্ট্যান্টের মান কখনও পরিবর্তন হয় না। সম্পূর্ণ স্ক্রিপ্টে একই মান থাকবে। পিএইচপিতে define () ফাংশন দিয়ে কোনো কনস্ট্যান্টের মান ঠিক করে দেয়া হয়। যেমন—

```
<?php
define ('TUTORIAL','Tutorialpoint is a informa-
tive tutorial site');
echo TUTORIAL;
?>
```

### আউটপুট

Tutorialpoint is a informative tutorial site

একবার মান ঠিক করে দিলে আর কখনও এই মান পরিবর্তন হবে না। যেমন— পাই (π)-এর মান ৩.১৪১৬, এদের মান নির্ধারণ করার ক্ষেত্রে কনস্ট্যান্ট ব্যবহার করা উপকারী।

## পিএইচপি ডাটা টাইপ

সব প্রোগ্রামিং ল্যাঙ্গুয়েজে ডাটা টাইপ থাকে। কোন একটি ভেরিয়েবল কী ধরনের ডাটা ধারণ

করে সেটা বুঝতে হবে যেমন— \$x = 9; অর্থাৎ x-এর মান এখন 9 এবং 9 একটি পূর্ণ সংখ্যা। মানটি পূর্ণসংখ্যা না হয়ে অন্য কিছু হতে পারত। যেমন— দশমিক হতে পারত। যেমন— ৯.৬৭ অথবা এমন কিছু। এর বাইরে হতে পারে একটি অক্ষরের সারি। যেমন— \$x = Tutorialpoint ইত্যাদি। মানে চিনিয়ে বা বুঝিয়ে দিতে হয় যে ভেরিয়েবলের মাঝে রাখা ডাটাটি কি একটি পূর্ণসংখ্যা বা স্ট্রিং বা দশমিক সংখ্যা (বা যতরকম ডাটা টাইপ আছে) বা অন্য কিছু। তাহলে \$x = 9-এর অর্থ হলো \$x এখন integer, \$x = Tutorialpoint-এর অর্থ হলো \$x এখন স্ট্রিং, \$x = 5.36-এর অর্থ হলো \$x এখন floating number ইত্যাদি।

পিএইচপিতে কয়েক ধরনের ডাটা টাইপ বা ধরন আছে। অর্থাৎ পিএইচপি ভেরিয়েবল এ ধরনের বা টাইপের ডাটা নিতে পারে এবং যেকোনো অপারেশন করতে পারে। মূলত পিএইচপির ডাটাকে দুটি ভাগে ভাগ করা হয়েছে।

### আউটপুট

```
int(1011100)
```

গ. Float বা দশমিক সংখ্যা : দশমিক সংখ্যা যেমন— ৫.২৩৬, ৫.২ ইত্যাদি হচ্ছে float number বা doubles বা real number।

### আউটপুট

```
float(5.2054)
```

সূচকীয় ফরম্যাটেও float number লেখা যায়, যেমন— \$x = 2x6e5;

ঘ. String বা অক্ষরের সারি : যেকোনো অক্ষর সিঙ্গেল কোট, ডাবল কোটেশন, heredoc এবং nowdoc সিনট্যাক্সের মধ্যে রাখলে সেটা স্ট্রিং, যেমন—

```
<?php
$x = "Tutorialpoint";
$x = " Tutorialpoint ";
$x = <<<HTML
<p> Tutorialpoint is a tutorial site</p>
HTML;
```

# পিএইচপি টিউটোরিয়াল

আনোয়ার হোসেন

পৃষ্ঠা-৬

## ০১. স্কেলার ডাটা টাইপ

এখানে চার ধরনের ডাটা টাইপ আছে।

ক. boolean বা বুলিয়ান : এ ধরনের ডাটার শুধু দুটি মান থাকতে পারে, সেটা হয় TRUE নতুবা FALSE। পিএইচপিতে অনেক ফাংশন আছে, যেগুলো বুলিয়ান মান ফেরত দেয়। বুলিয়ান ডাটাতে if কন্ডিশন ব্যবহার করা হয়। is\_bool() ফাংশনটি ব্যবহার করে কোনো ভেরিয়েবলে বুলিয়ান ডাটা আছে কি না পরীক্ষা করে দেখা যায়।

```
<?php
$x = TRUE;
$y = 9;
var_dump(is_bool($y)); // output bool(false)
var_dump(is_bool($x)); // output bool(true)
?>
```

দেখুন \$y যখন চেক করেছি, তখন is\_bool()-এর আউটপুট false এসেছে। কেননা \$y-এর মান বুলিয়ান নয় বরং 9 অর্থাৎ পূর্ণসংখ্যা বা integer। এভাবে যেকোনো ভেরিয়েবল কিংবা ফাংশনের রিটার্ন মান চেক করতে পারেন।

খ. integer বা পূর্ণসংখ্যা : দশমিক ছাড়া সংখ্যা হচ্ছে এই ধরনের ডাটা। যেমন— ৯, ৮, -৬৫, -1 ইত্যাদি হচ্ছে integer ডাটা। is\_int() ফাংশন দিয়ে যেকোনো ভেরিয়েবলকে চেক করে দেখতে পারেন, যদি পূর্ণ সংখ্যা (integer) হয়, তাহলে রিটার্ন করবে TRUE আর না হলে করবে FALSE। পূর্ণসংখ্যা ধনাত্মক কিংবা ঋণাত্মক হতে পারে। এমনকি ১০-ভিত্তিক (decimal-10 based) ছাড়াও হেক্সাডেসিমাল (১৬-ভিত্তিক), octal (৮-ভিত্তিক), দ্বিমিক (২-ভিত্তিক) বা বাইনারি সংখ্যাও integer হতে পারে। যেমন—

```
<?php
$x = 1011100; //binary number 92
var_dump($x);
?>
```

?>

অবশ্য সব অক্ষরের (ক্যারেক্টার) সাপোর্ট নেই। সোজা কথায় ইউনিকোড ক্যারেক্টার সাপোর্টেড নয়। পিএইচপি ৭ ভার্সন সামনে আসছে, সেখানে ইউনিকোড ক্যারেক্টারও সাপোর্ট থাকবে।

## ০২. Compound বা মিশ্র ডাটা টাইপ

ক. অ্যারে : পিএইচপি অ্যারে টিউটোরিয়াল (PHP Array)।

খ. অবজেক্ট : অবজেক্ট এবং ক্লাস (Object and Class)।

এ ছাড়া আরও দুটি বিশেষ ডাটা টাইপ আছে resource এবং NULL resource সাধারণত বিভিন্ন ধরনের ডাটা হয়ে থাকে। যেমন— ডাটাবেজে অপারেশন করে যেসব ডাটা রিটার্ন করে সেগুলো resource ডাটা হিসেবে দেখে। আর NULL হলো ফাঁকা ডাটা। কোনো ভেরিয়েবল ডিক্লেয়ার করা হলো কিন্তু কোনো মান দেয়া হলো না, তাহলে সেই ভেরিয়েবলটির মান null হয়ে যায়।

## টাইপ জাগলিং

যেকোনো ল্যাঙ্গুয়েজে কোনো ভেরিয়েবলে কোনো কিছু নেয়া হলে আগেই বলে দিতে হবে সেটা কোন ধরনের ডাটা। একে বলা হয় টাইপ ডিক্লেয়ার করা বা explicit type definition। মজার ব্যাপার হলো, পিএইচপিতে এই কাজটি করার দরকার হয় না। পিএইচপি ইঞ্জিন নিজে নিজে ডাটা দেখে-বুঝে নেয় ডাটার ধরন। এই স্বয়ংক্রিয়ভাবে ডাটা চিনে নেয়াকে বলে 'টাইপ জাগলিং'। এটিকে পিএইচপির শক্তিশালী বৈশিষ্ট্য বলে গণ্য করা হয়।