

# Function

1. Just Print (Function এর মধ্যে টাইপ দিতে হয় না।)

```
<?php
//no need to specify variable type
function myFunction($width,$height){
    echo $width*$height."<br>";
}
myFunction(10,20);
?>
```

localhost:3000/index.php

PGR localhost / 127.0.0.1... Account Par

200

2. value return

```
<?php
//no need to specify variable type
//value return
function areaRectangle($width,$height){
    return $width*$height;
}
echo "Area is ".areaRectangle(10,20);
?>
```

localhost:3000/index.php

PGR localhost / 127.0.0.1... Account Pa

Area is 200

3. Array Return

```
<?php
//no need to specify variable type
//value return
function areaRectangle(){
    return ["name"=>"utsab","age"=>20];
}

$personData=areaRectangle();
print_r($personData);
?>
```

Array ( [name] => utsab [age] => 20 )

#### 4.Built in function

```
<?php
$name="utsab roy";
echo strlen($name);
?>
```

9

#### 5.Default parameter

এখানে দুটি argument pass না করে একটা pass করলে ডিফল্ট argument কাজ করবে। দুটি argument pass ডিফল্ট কাজ করবে না।

```
<?php
function sum($a,$b=10){
    return $a+$b;
}
echo"Sum is ".sum(20,20)."<br>";
echo "Sum is ".sum(15);
?>
```

Sum is 40  
Sum is 25

## 6.Single Type Declaration

(এক্ষেত্রে parameter এ টাইপ দেয়া যায় এবং return type ও দেয়া যায়।)

```
<?php
//we can give data type in method and give return type
function square(int $number): int {
    return $number * $number;
}

echo square(4); // Outputs: 16
?>
```

localhost:3000/index.php

PGR localhost / 127.0.0.1... Accou

16

## 7.Strict Types

(এখানে int এর জায়গায় string পাঠালেও কাজ করতেছে কিন্তু আমরা চাচ্ছি datatype match না হলে এরর দেখাবে এজন্য string type enable করতে হয়।)

```
<?php
//we can give data type in method and give return type
function square(int $number): int {
    return $number * $number;
}

echo square("10"); // Outputs: 16
?>
```

localhost:3000/index.php

PGR localhost / 127.0.0.1... Accou

100

```
<?php
declare(strict_types=1);
//we can give data type in method and give return type
function square(int $number): int {
    return $number * $number;
}

echo square("10"); // Outputs: 16
?>
```

**Fatal error:** Uncaught TypeError: Argument 1 passed to square() must be of the type integer, string given, called in C:\xampp\htdocs\Module 7\index.php(8): square('10') #1 {main} thrown in C:\xampp\htdocs\Module 7\index.php on line 4

## 8. Multiple Type Hint

এক্ষেত্রে একটা argument এ একাধিক টাইপ দিয়ে দিতে পারি অর্থাৎ যে কোন টাইপ match করলে রিসিভ হবে ডাটা।(তবে এটা php 8 এ সাপোর্ট করবে।)

```
<?php

function add(int|float $a, int|float $b): int|float {
    return $a + $b;
}

echo add(5, 3.5);    // Outputs: 8.5
echo add(4, 2);      // Outputs: 6
?>
```

10  
utsab  
20  
babu

## 9. Variadic Function

(এখানে একসাথে একাধিক value recive হয়।)

```
<?php
//we can recive multiple data
function allData(...$values){
    foreach($values as $value){
        echo $value."<br>";
    }
}
allData(10,"utsab",20,"babu")
?>
```

10  
utsab  
20  
babu

```
<?php
function greet($greeting,...$names){
    foreach($names as $name){
        echo "$greeting , $name<br>";
    }
}
greet("Hellow","utsab","babu","sebak");
?>
```

Hellow , utsab  
Hellow , babu  
Hellow , sebak

## 10. anonymous functions or closures.

() Function define

() Function call

এই ফাংশন পেইজ যখন লোড হবে তখন execute হবে।

```
<?php
(function() {
    // Function code
})();
?>
```

```
<?php
(function() {
    echo"Hellow world!";
})();
?>
```

Hellow world!

//value pass করতে পারি anonymous function এ।

```
<?php
    $add=function($a,$b){
        return $a+$b;
    };
    echo $add(10,20);
?>
```

localhost:3000/index.php

PGR localhost / 127.0.0.1... Account Panel

30

# LOOP

## FOR LOOP

```
<?php
    for($i=1;$i<=3;$i++){
        echo"Iteration $i<br>";
    }
?>
```

localhost:3000/index.php

PGR localhost / 127.0.0.1... Account Panel

Iteration 1

Iteration 2

Iteration 3

## WHILE LOOP

```
<?php
    $i=1;
    while($i<=2){
        echo "Iteration $i <br>";
        $i++;
    }
?>
```

← → ↻ ⓘ localhost:3000/index.php

☐ | 🗄️ PGR 🔥 localhost / 127.0.0.1... 🅐 Acco

Iteration 1

Iteration 2

## DO-WHILE LOOP

```
<?php
    $i=1;
    do{
        echo"Iteration $i<br>";
        $i++;
    }while($i<=2);
?>
```

← → ↻ ⓘ localhost:3000/index.p

☐ | 🗄️ PGR 🔥 localhost / 127.0.0.1... 🅐

Iteration 1

Iteration 2

# Array

**Indexed Arrays:** Use numeric indices (either default or custom).

(এখানে ইনডেক্স 0,১,... থাকে)

Indexed array access করা যাবে for loop এবং foreach loop দিয়ে।

```
<?php
//index array access by for loop and foreach loop
$indexed_array=[10,20,30,"utsab","babu"];
//for loop
for($i=0;$i<count($indexed_array);$i++){
    echo "Index=$i value= $indexed_array[$i] <br>";
}

//foreach
foreach($indexed_array as $value){
    echo "value=$value<br>";
}
?>
```

Index=0 value= 10  
Index=1 value= 20  
Index=2 value= 30  
Index=3 value= utsab  
Index=4 value= babu  
value=10  
value=20  
value=30  
value=utsab  
value=babu

**Associative Array:** Use descriptive keys (either strings or integers) to associate with values.

[key value থাকবে এবং এটা foreach লুপ দিয়ে access করতে হবে।]

```
<?php
$person=[
    "name"=>"utsab",
    "id"=>154061,
    "result"=>3.41
];

foreach($person as $key=>$value){
    echo "$key==>$value<br>";
}

?>
```

name==>utsab  
id==>154061  
result==>3.41

## Memory structure

```
$person
└─> Array (hash table)
    └─ Bucket 0
        └─ Key: "name" (String)
            └─ Value: "utsab" (String)
    └─ Bucket 1
        └─ Key: "id" (String)
            └─ Value: 154061 (Integer)
    └─ Bucket 2
        └─ Key: "result" (String)
            └─ Value: 3.41 (Float)
```



Bucket #	Key	Value	Type
0	"name"	"utsab"	String
1	"id"	154061	Integer
2	"result"	3.41	Float

```
<?php
$students = [
    "student1" => [
        "name" => "Utsab",
        "subjects" => [
            "Math" => "A",
            "English" => "B+",
            "Science" => "A-"
        ]
    ],
    "student2" => [
        "name" => "Babu",
        "subjects" => [
            "Math" => "B",
            "English" => "A",
            "Science" => "B+"
        ]
    ],
    "student3" => [
        "name" => "Alice",
        "subjects" => [
            "Math" => "A-",
            "English" => "B",
            "Science" => "A"
        ]
    ]
];

// Displaying data with foreach
foreach ($students as $studentKey => $studentInfo) {
    echo "Student ID: $studentKey<br>";
    echo "Name: " . $studentInfo["name"] . "<br>";
    echo "Subjects and Grades:<br>";

    foreach ($studentInfo["subjects"] as $subject => $grade) {
        echo "- $subject: $grade<br>";
    }

    echo "<br>";
}
?>
```

Student ID: student1

Name: Utsab

Subjects and Grades:

- Math: A
- English: B+
- Science: A-

Student ID: student2

Name: Babu

Subjects and Grades:

- Math: B
- English: A
- Science: B+

Student ID: student3

Name: Alice

Subjects and Grades:

- Math: A-
- English: B
- Science: A

\$students

└> Array (hash table)

└ Bucket 0

|   └ Key: "student1"

|   └ Value: Array

|       └ Bucket 0

|       |   └ Key: "name" -> "Utsab" (String)

|       |

|       └ Bucket 1

|       |   └ Key: "subjects"

|       |   └ Value: Array

|       |       └ Bucket 0

|       |       |   └ Key: "Math" -> "A" (String)

|       |       |

|       |       └ Bucket 1

|       |       |   └ Key: "English" -> "B+" (String)

|       |       |

|       |       └ Bucket 2

|       |       |   └ Key: "Science" -> "A-" (String)

└ Bucket 1

|   └ Key: "student2"

|   └ Value: Array

|       └ Bucket 0

|       |   └ Key: "name" -> "Babu" (String)

|       |

|       └ Bucket 1

|       |   └ Key: "subjects"

|       |   └ Value: Array

|       |       └ Bucket 0

|       |       |   └ Key: "Math" -> "B" (String)

|       |       |

|       |       └ Bucket 1

|       |       |   └ Key: "English" -> "A" (String)

|       |       |

|       |       └ Bucket 2

|       |       |   └ Key: "Science" -> "B+" (String)

```

└─ Bucket 2
  └─ Key: "student3"
    └─ Value: Array
      └─ Bucket 0
        └─ Key: "name" -> "Alice" (String)
      └─ Bucket 1
        └─ Key: "subjects"
          └─ Value: Array
            └─ Bucket 0
              └─ Key: "Math" -> "A-" (String)
            └─ Bucket 1
              └─ Key: "English" -> "B" (String)
            └─ Bucket 2
              └─ Key: "Science" -> "A" (String)

```

### 3 loop

```

<?php
$students=[
    "student1"=>[
        "name"=>"utsab",
        "subjects"=>[
            "math"=>"A+",
            "english"=>"B+",
            "science"=>"A"
        ]
    ],
    "student2"=>[
        "name"=>"babu",
        "subjects"=>[
            "math"=>"A",
            "english"=>"A-",
            "science"=>"Fail"
        ]
    ]
];

```

```

foreach($students as $singleStudent=>$studentInfos){
    echo"Student Id $singleStudent <br>";
    foreach($studentInfos as $studentInfoKey=>$studentInfoValue){
        if($studentInfoKey=='subjects'){
            foreach($studentInfoValue as $singleSubject=>$singleSubjectResult){
                echo"$singleSubject $singleSubjectResult<br>";
            }
        }else{
            echo" $studentInfoKey $studentInfoValue<br>";
        }
    }
    echo"<br><br>";
}
?>

```

```

$students=[
    "student1"=>[
        "name"=>"utsab",
        "subjects"=>[
            "math"=>"A+",
            "english"=>"B+",
            "science"=>"A"
        ]
    ],
    "student2"=>[
        "name"=>"babu",
        "subjects"=>[
            "math"=>"A",
            "english"=>"A-",
            "science"=>"Fail"
        ]
    ]
];

```

Student1

B=0; Key = Student1  
B=1; Key = Student2

value = Array  
value = Array

```

$students=[
    "student1"=>[
    ],
    "student2"=>[
    ]
];

```

foreach (\$students as \$key => \$value)

key: student1, student2

value: Array

name => " " ?  
Subjects => Array

```

$students=[
  "student1"=>[
    "name"=>"utsab",
    "subjects"=>[
      "math"=>"A+",
      "english"=>"B+",
      "science"=>"A"
    ]
  ],
  "student2"=>[
    "name"=>"babu",
    "subjects"=>[
      "math"=>"A",
      "english"=>"A-",
      "science"=>"Fail"
    ]
  ]
];

```

Loop bnto  
students 1 to 2

```

$students=[
  "student1"=>[
    ],
    "student2"=>[
    ]
];

```

```

"student1"=>[
  "name"=>"utsab",
  "subjects"=>[
  ]
]

```

for(\$students as \$key1=>\$value1)  
↓  
student1

[key: value  
key value]

for(\$value1 as \$key2=>\$value2)  
↓

```

$students=[
  0 "student1"=>[
    0 "name"=>"utsab",
    1 "subjects"=>[
      0 "math"=>"A+",
      1 "english"=>"B+",
      2 "science"=>"A"
    ]
  ],
  1 "student2"=>[
    0 "name"=>"babu",
    1 "subjects"=>[
      0 "math"=>"A",
      1 "english"=>"A-",
      2 "science"=>"Fail"
    ]
  ]
];

```

```

$students=[
  "student1"=>[
    ],
    "student2"=>[
    ]
];

```

key=student1

```

"student1"=>[
  "name"=>"utsab",
  "subjects"=>[
  ]
]

```

key=subjects

```

"subjects"=>[
  "math"=>"A+",
  "english"=>"B+",
  "science"=>"A"
]

```

for(\$students as \$key1=>\$value1)

{

0 student1  
1 student2

for(\$value1 as \$key2=>\$value2){

0 name  
1 subject

for(\$value2 as \$key3=>\$value3){

0 math  
1 english  
2 science

}

}