WEB PROGRAMMING

SERVER SIDE PROGRAMMING

PHP SCRIPTING LANGUAGE

MYSQL DATABASE

APACHE SERVER

SERVER-SIDE SCRIPTING

- Server-side scripting is a web server technology in which a user's request is fulfilled by running a script directly on the Web server to generate dynamic HTML pages.
- Server-side scripting enables the ability to highly customize the response based on the user's requirements, access rights, or queries into data stores.

SERVER SIDE WEB TECHNOLOGIES

- PHP
- ASP (Active Server Pages) & ASP.NET
- JSP (Java Server Pages)
- Java Servlets
- C++/Java
- PERL
- XML

INTRODUCTION TO PHP

- Originally stood for Personal home Page, now Hypertext Preprocessor
- Html embedded scripting language, Which makes developing dynamic websites more accessible
- A server side scripting language (every thing does in the server)
- Cross platform technology / Platform Independent (runs on most OS)

INTRODUCTION TO PHP

- Better faster and easier to learn, powerful, popular
- Design to something only after and event occurs
- Ex. When user submit a form goes to URL
- Supports many DB's (not only MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC)

INTRODUCTION TO PHP

- Web Server Independent
- Free and Open Source
  - no warranty, no limits on usage
  - Source code is available for any modifications
  - Freely Extended (must share source, represent original works and owners)
  - License is not specific to a product or restrict other software and also technology neutral.

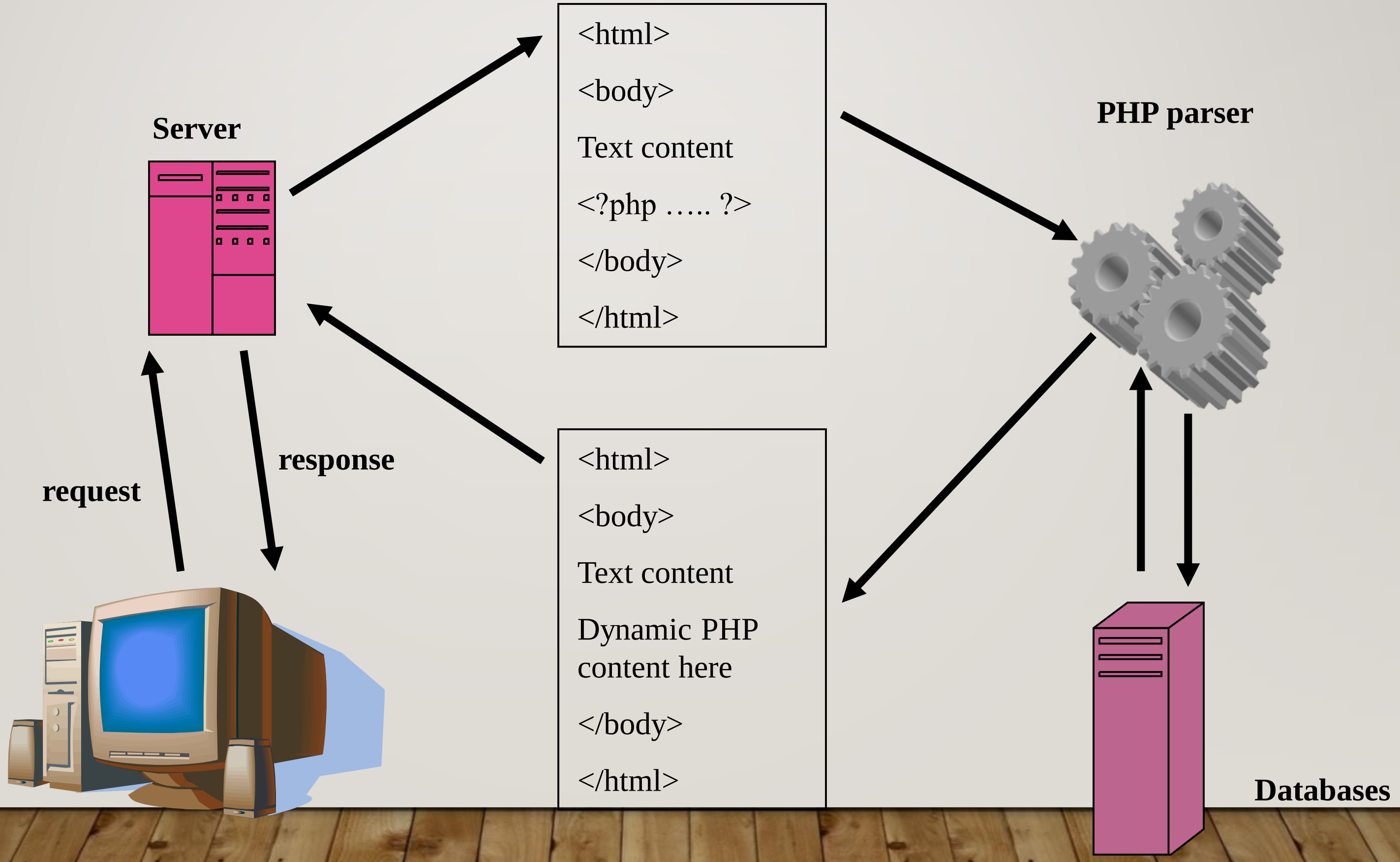
WHAT'S SO GREAT ABOUT PHP?

- **PHP runs on different platforms** (or cross platform) such as Windows, Linux, Unix, etc.
- **PHP is compatible with almost all servers used today** (Apache, IIS and any other web server that support the CGI standard)
- **PHP is FREE** to download from the official PHP resource: www.php.net
- PHP is easy to learn and runs efficiently on the server side

HOW DOES PHP WORKS?

- A typical PHP page will contain number of PHP elements along with HTML markup elements and other textual content
- When a web browser request a PHP page from a web server that is PHP enabled the server will call up the PHP parser to process all the PHP elements on that page
- The PHP parser executes the PHP script instructions on the page ,generating a HTML document that is then sent to the web browser as a response to the original request
- The PHP parser may also be asked to retrieve information from a database so the entire process appears like the illustration bellow

HOW DOES PHP WORKS?



Server

request

response

```
<html>
<body>
Text content
<?php ….. ?>
</body>
</html>
```

PHP parser

```
<html>
<body>
Text content
Dynamic PHP
content here
</body>
</html>
```

Databases

CREATING A PHP ENVIRONMENT?

In order to develop and run PHP web pages following three components need to be installed on the computer system

- Web server
  - Internet Information Server (IIS)
  - Apache server
- Database
  - Oracle
  - MySQL
- PHP parser

WHAT DO YOU NEED?

- Download PHP for free here:
  http://www.php.net/downloads.php
- Download MySQL for free here:
  http://www.mysql.com/downloads/index.html
- Download Apache for free here:
  http://httpd.apache.org/download.cgi

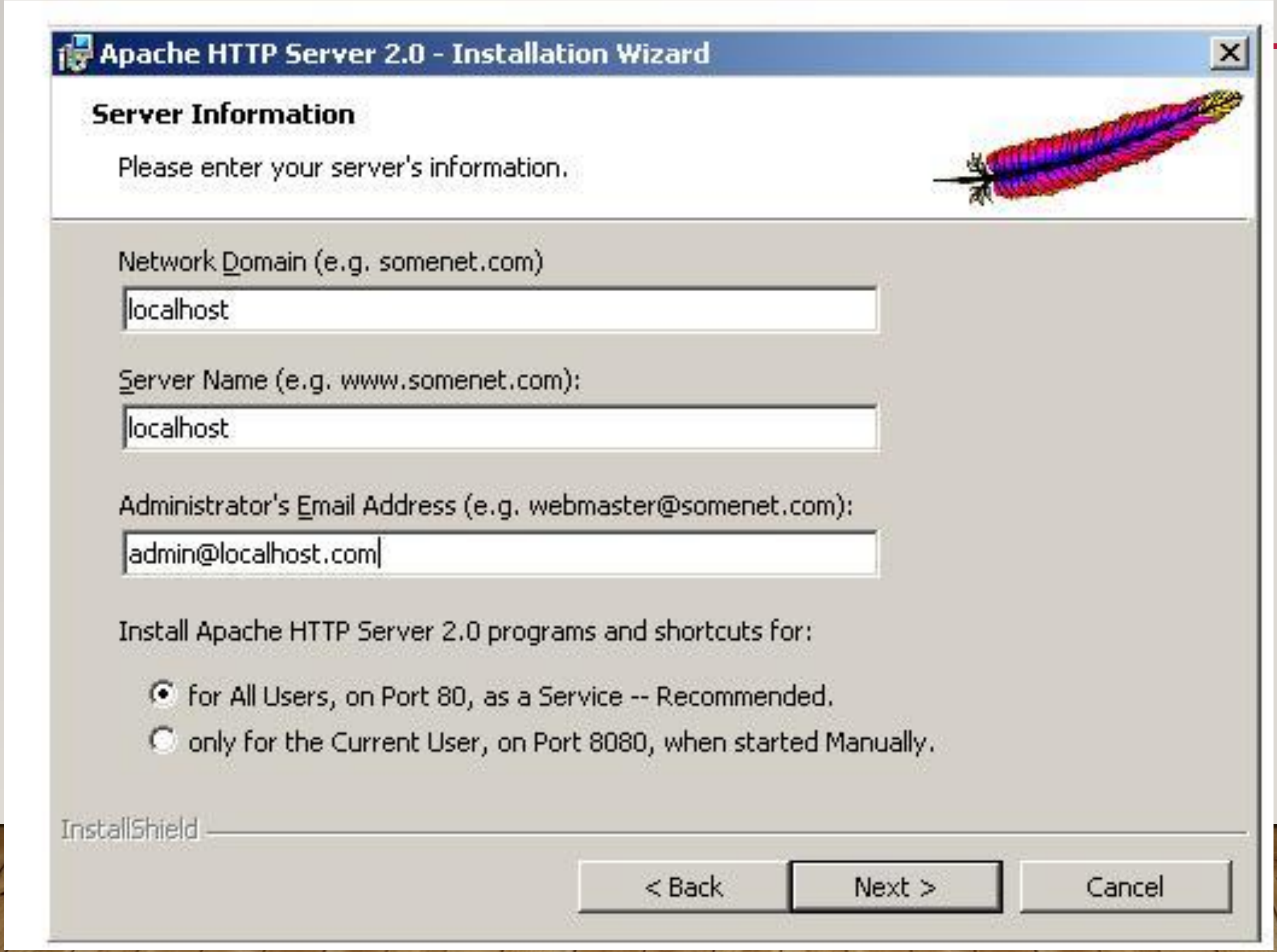
INSTALLING APACHE

- At first remove IIS , Start → settings → control panel → add or remove programs → click add/remove windows components → if the IIS tick on , disable and stop it.
- Double click on apache setup and install it your computer.

Give any domain name e.g.. anyname.com

Give any server name e.g. www.anyname.com

INSTALLING APACHE



Apache HTTP Server 2.0 - Installation Wizard

Server Information

Please enter your server's information.

Network Domain (e.g. somenet.com)

localhost

Server Name (e.g. www.somenet.com):

localhost

Administrator's Email Address (e.g. webmaster@somenet.com):

admin@localhost.com

Install Apache HTTP Server 2.0 programs and shortcuts for:

● for All Users, on Port 80, as a Service -- Recommended.

○ only for the Current User, on Port 8080, when started Manually.

INSTALLING PHP

- ❑PHP zip(php 5.x) file must be extracted ,using any zip file tool like winzip, to convenient location , like C:\PHP or copy the extracted file and paste it to the C:\ root directory , then rename as PHP
- ❑In this PHP folder there is a text file called "install.txt" which has all the details about how to install php and other php related softwares .
- ❑Also there are two files called
  - php.ini-dist
  - php.ini-recommended

  Copy one of these file to C:\Apache2 and rename it as "php.ini"

INSTALLING PHP CONT.

- Go to conf folder in apache2 folder and open httpd.txt file and paste following lines in the top of the file .

  ```
  LoadModule php5_module C:/php/php5apache2.dll
  AddType application/x-httpd-php .php
  ```

- Stop the web server and restart it.

TESTING PHP

- Take notepad and write following code

<?php phpinfo(); ?> save this as follows in the web server

Save as **c:\ windows\ apache\ hotdocs\ info.php**

- Access Internet Explorer and type **http:\\localhost\info.php** (display information about php.ini)

WHAT IS MYSQL?

- MySQL is a database server
- MySQL is ideal for both small and large applications
- MySQL supports standard SQL
- MySQL compiles on a number of platforms
- MySQL is free to download and use

PHP + MYSQL

- PHP combined with MySQL are cross-platform (you can develop in Windows and serve on a Unix platform)

CONFIGURING OF MYSQL IN PHP

- Double click on php.ini file and see windows extensions uncomment php-mysql.dill line.
- Copy php-mysql.dill file in ext folder in php folder and paste in php folder.
- Copy libmysql.dill and php-mysql.dill and paste it in windows\system32
- If mysql not working restart the web server.

APACHE BENEFIT

- **Apache is well supported** - Apache is free and available 24 hours a day via Internet
- **Apache is multi-platform** - Apache can run on virtually any hardware platform (from PCs to mainframes), and almost any operating system, such as Linux, Windows etc.
- **Apache is secure** - security holes are rare but when they exist they are discovered and fixed quickly
- **Apache is extensible** - anyone can write modules that easily plug in to Apache. anyone with programming skills can write the modules you need.
- **Apache is database-friendly** - you can interface Apache with virtually any commercial database, such as Oracle, Sybase, DB2, and Informix, as well as free databases such as MySQL and Postgres.
- **Apache is hardware-friendly** - Apache generally consumes far fewer hardware resources that commercial web servers.
- **No Microsoft Viruses** - Apache is immune viruses that target at Microsoft Web servers.

MYSQL BENEFIT

- ❑One of the most used open source database in the world.
- ❑Capacity to handle 50,000,000+ records.
- ❑Very fast command execution, perhaps the fastest to be found on the market
- ❑Flexible and secure password system to protect your data - powerful security system
- ❑Fast, reliable, easy to use,
- ❑On-line help facility
- ❑Comes with a source code
- ❑Multi-User and works on Several Platforms

THREE IN TOGETHER…

- Easy to install
- No Manual configuration
- E.g
  - WAMP server
  - XAMPP

WHAT IS A PHP FILE?

- PHP files can contain text, HTML tags and scripts
- PHP files are returned to the browser as plain HTML
- PHP files have a file extension of ".php", ".php3", or ".phtml"

BASIC PHP SYNTAX

- A P~~HP scripting block always starts with **<?php** and ends with **?>**.A~~ PHP scripting block can be placed anywhere in the document.
- E.g.

  <?php

  some cording

  ?>

  → Start Tag
- On servers with shorthand support enabled you can start a scripting block with **<?** and end with **?>**.

  → End Tag

SAY “HELLO WORLD!” USING PHP

- A PHP file normally contains HTML tags, just like an HTML file, and some PHP scripting code.

```
<html>
<body>
    <?php
        echo “Hello World!”;
     ?>
</body>
</html>
```

Each code line in PHP must be ended with a semicolon

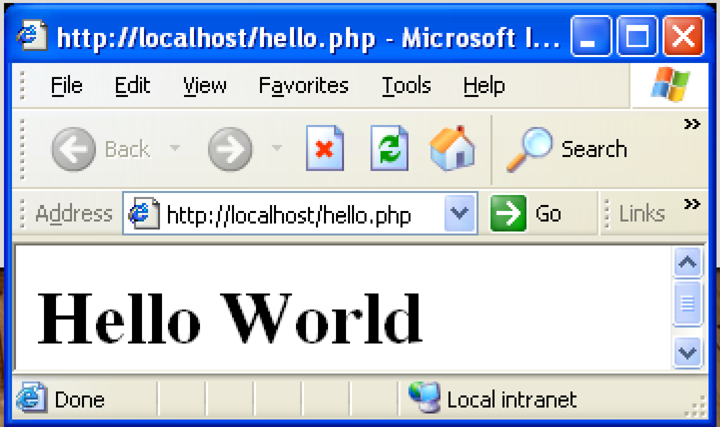
SIMPLE PHP PROGRAM

- Type this ~~code in any text editor and save it as hello.php in Apache’s htdocs~~ folder

```
<html>
<?php  echo ("<h1>Hello World</h1>");  ?>
</html>
```

- Open internet Explorer and write http:\\localhost\hello.php in address bar

- OUTPUT



http://localhost/hello.php - Microsoft I...

Address: http://localhost/hello.php

```
Hello World
```

OUTPUT A TEXT IN PHP

- There are two basic statements to output text with PHP
- **echo**
- **print**
- In the example above we have used the **echo** statement to output the text "Hello World".
- Also you can use **print** as bellow

```
<?php     print “Hello World!”;     ?>
```

COMMENTS IN PHP

- In PHP there are two types of comments
- Single-line
  - **// or** # to make a single line comment
- Large comment block
  - **/* and */** to make a large comment block

```
<?php
//This is a comment
# This is a Comment

/*
This is
a comment
block
*/
?>
```

VARIABLES IN PHP

- Variables are creating by adding “**$**“ to the front of a name.

  $variable name

  e.g. $result , $text, $num

- Variable name is **case sensitive**
- Semicolon is used to end the php statement

  ```
  $a=5;
  $n=7.7;
  ```

- php will determined the data type at the time data is assign to a each variable.

STRING VARIABLES IN PHP

- String variables are used for values that contains characters.
- A string can be used directly in a function or it can be stored in a variable.

```
<?php

$txt = "Hello World";

echo $txt;

?>
```

VARIABLES IN PHP

- Variable names can be of any length
- can include letters, numbers and underscores
- cannot start with a digit
- case-sensitive
- can have the same name as a function.

VARIABLE CONT.

- A variable can represent another variable's name

```
        $a = "Mark";
        $$a = "Jones";
        $a = "Mickey";
        $$a = "Brown";

echo $Mark; // Jones
echo $Mickey; // Brown
echo $$a; // Brown
$a = "Mark";
echo $$a; // Jones
```

SIMPLE PROGRAMS

(1)

```
<?php
$a=7;
echo “$a”;
/>
```

(2)

```
<?php
$name=“saman”
echo $name;
?>
```

EXERCISE

- Write a php program to display your name , address, birthday and school.

ANSWER

```
<?php
$name="Sarath";
$address="kegalle";
$birth="1981/10/30";
echo $name;
echo "<br/>";
echo $address;
echo "<br/>";
echo $birth;

?>
```

DATA TYPES

- Eight standard data types available in php

Type	Example	Keyword
Boolean	True/false	bool
Integer	5	integer
Float or Double	5.767	double
string	“saman”	string
object	an instance of class	
Array	Array of values	
Resource	database	
Null	Un initialized variable	

DATA TYPES CONT.

- To assign values to variables:

```
$foo = ‘bar’;     Data Type: String
$foo = 1;    Data Type: integer
$foo = 5.34;     Data Type: Double
$foo = array(“bar”,”united”);    Data Type: Array
```

- Data Types are automatically assigned though you can force a data type by type casting. For example:

```
$foo = ‘Hello’;
$bar = (int)$foo;
$bar now equals 0
```

- Almost all variables are local. Global include $_POST
- Array in php

```
$names[0] = 'Helen';
$names[1] = 'Susan';
$names[2] = 'Marc';
```

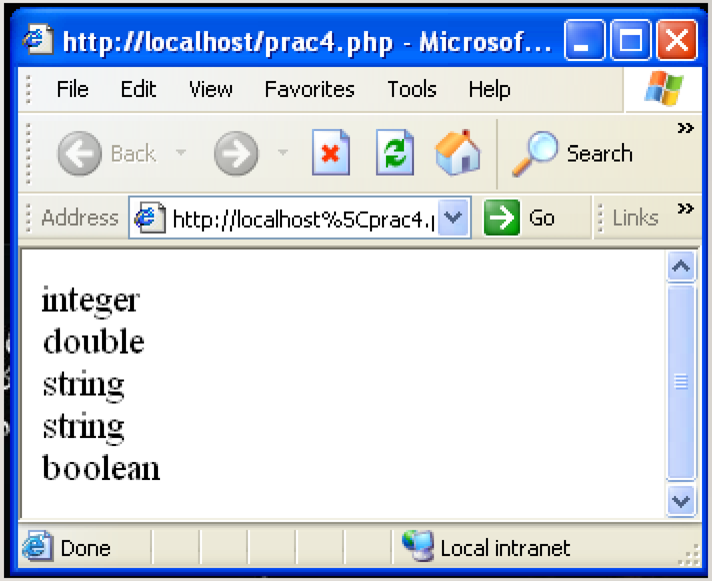
GETTYPE()

- You can used the building php function called gettype() to test the type of variable.

- If you place a variable between the parenthesis of the function called gettype() returns a string representing the relevant type.

GETTYPE()

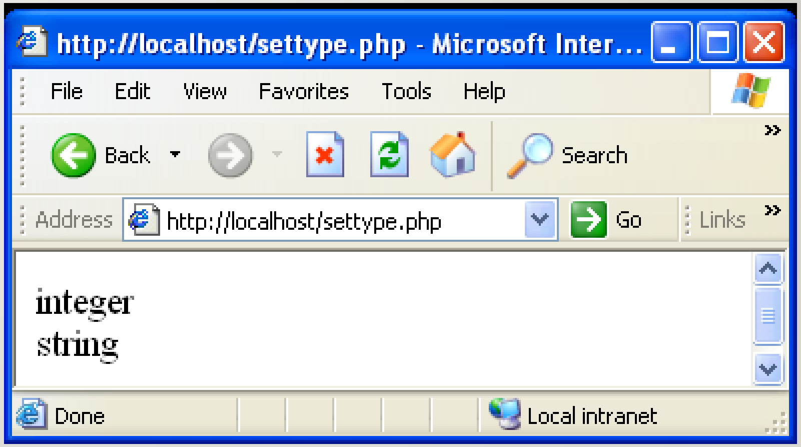
```
<html>
<body>
<?PHP
$varint=12;
$vardouble=12.12;
$varstring1="test";
$varstring2='test';
$varbool=true;
echo gettype($varint),"<br>";
echo gettype($vardouble),"<br>";
echo gettype($varstring1),"<br>";
echo gettype($varstring2),"<br>";
echo gettype($varbool),"<br>";
?>
</body>
</html>
```



```
integer
double
string
string
boolean
```

SETTYPE()

```
<?php
   $ccc=7;
   echo gettype($ccc);
   settype($ccc,'string');
   echo "<br/>";
   echo gettype($ccc);
?>
```


http://localhost/settype.php - Microsoft Inter...
File Edit View Favorites Tools Help
Back
Search
Address http://localhost/settype.php
Go
Links
Done
Local intranet

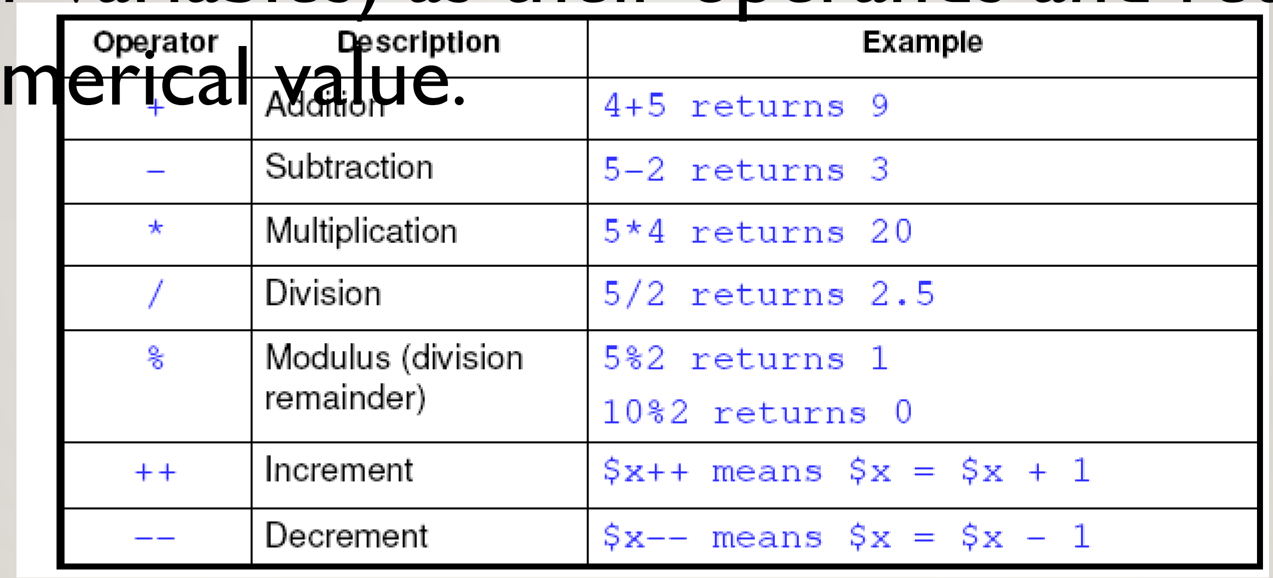
```
integer
string
```

PHP OPERATORS

- There are lists of different operators used in PHP
- **Arithmetic Operators (e.g. =, +=, *=)**
- **Assignment Operators (e.g. +, -, *)**
- **Comparison Operators (e.g. <, >, >=, ==)**
- **Logical Operators (e.g. !, &&, ||)**

ARITHMETIC OPERATORS

Arithmetic operators take numerical values (either literals or variables) as their operands and return a single numerical value.



Operator	Description	Example
+	Addition	4+5 returns 9
–	Subtraction	5–2 returns 3
*	Multiplication	5*4 returns 20
/	Division	5/2 returns 2.5
%	Modulus (division remainder)	5%2 returns 1 10%2 returns 0
++	Increment	$x++ means $x = $x + 1
––	Decrement	$x–– means $x = $x – 1

EXERCISE

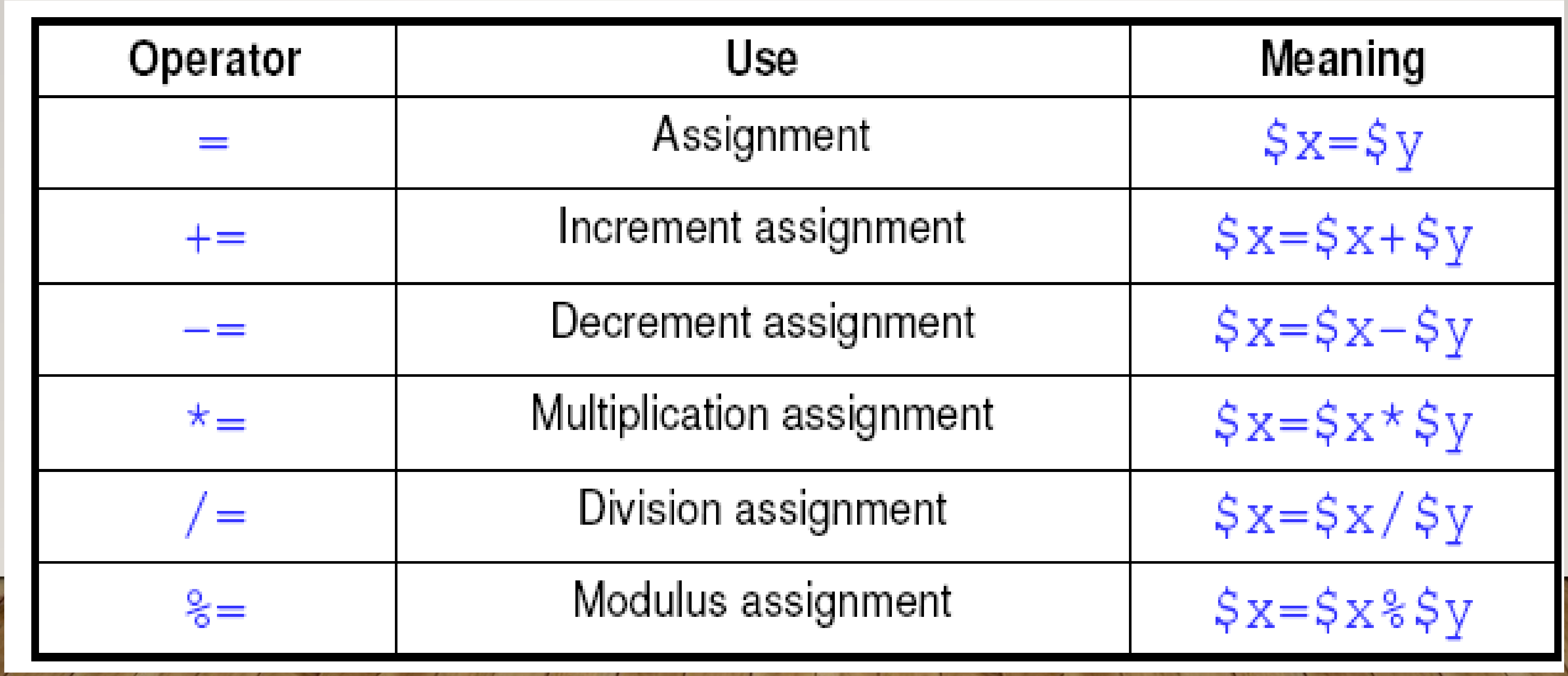
- Write a php program to add two values and print the final result.

ANSWER

```
<?php
$a=7;
$b=8;
$c=$a+$b;
echo “$c”;
/>
```

ASSIGNMENT OPERATORS

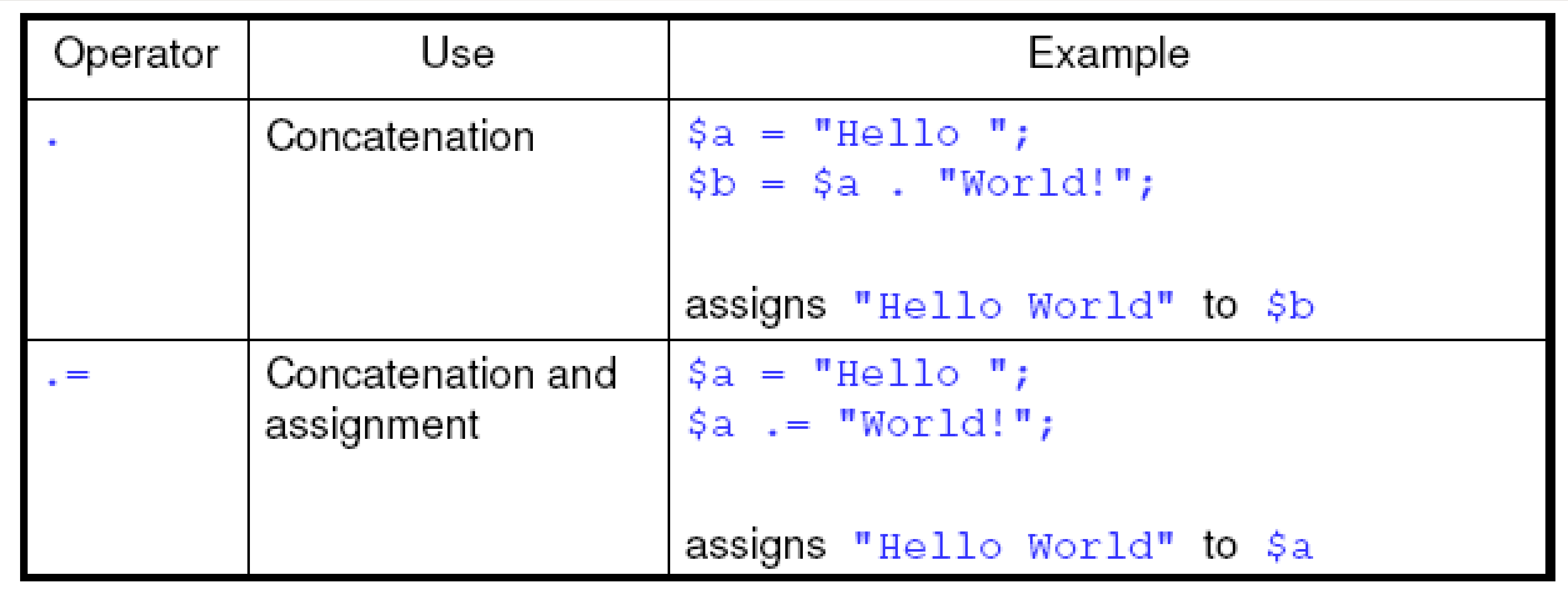
An assignment operator assigns a value to its left operand based on the value of its right operand.



Operator	Use	Meaning
=	Assignment	$x=$y
+=	Increment assignment	$x=$x+$y
−=	Decrement assignment	$x=$x−$y
*=	Multiplication assignment	$x=$x*$y
/=	Division assignment	$x=$x/$y
%=	Modulus assignment	$x=$x%$y

STRING OPERATORS

- PHP has two string operators:



Operator	Use	Example

`.`	Concatenation	`$a = "Hello ";`
		`$b = $a . "World!";`
		
		assigns `"Hello World"` to `$b`

`.=`	Concatenation and assignment	`$a = "Hello ";`
		`$a .= "World!";`
		
		assigns `"Hello World"` to `$a`

EXERCISE

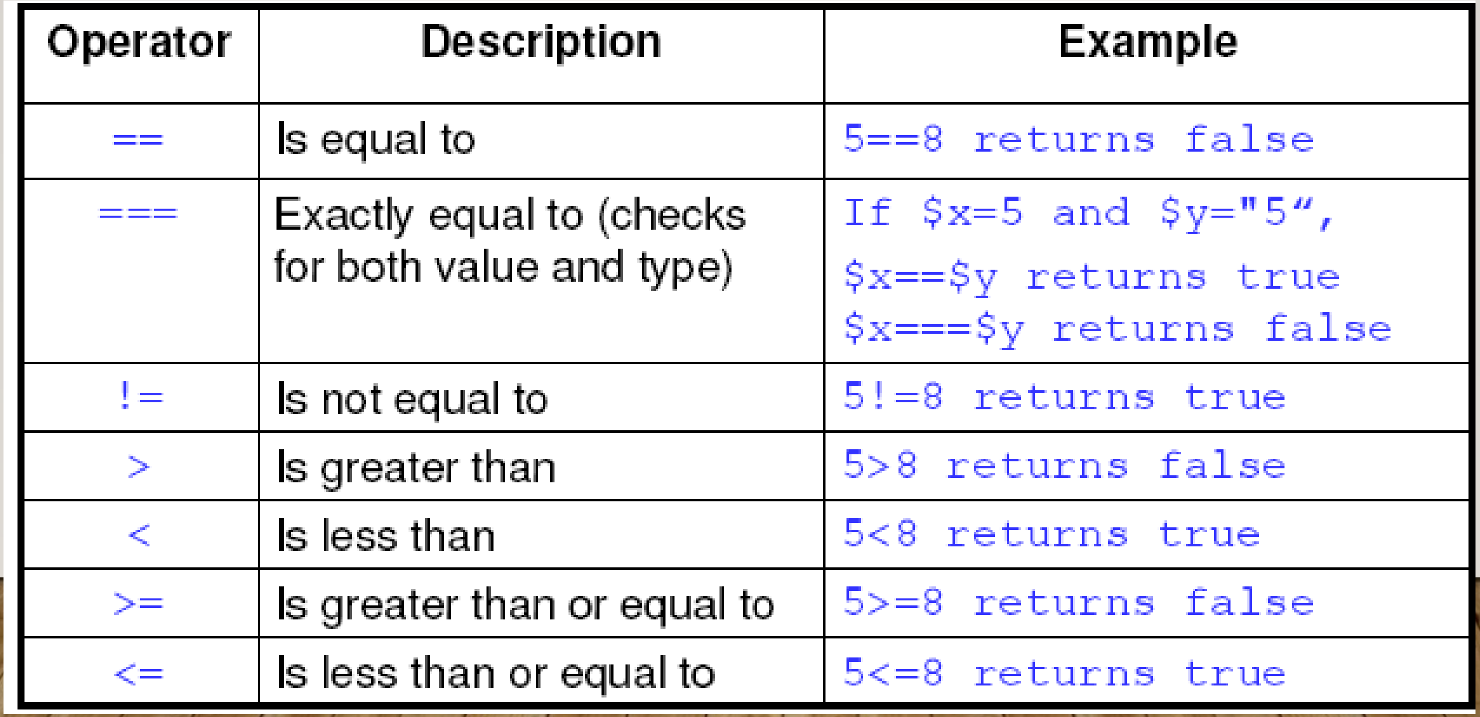
e.g.

```
<?php
$name=“sarath” ;
$lname=“perera”;
$full=$fname .= $lname;
echo $full;
?>
```

Answer is **sarath perera**

COMPARISON OPERATORS

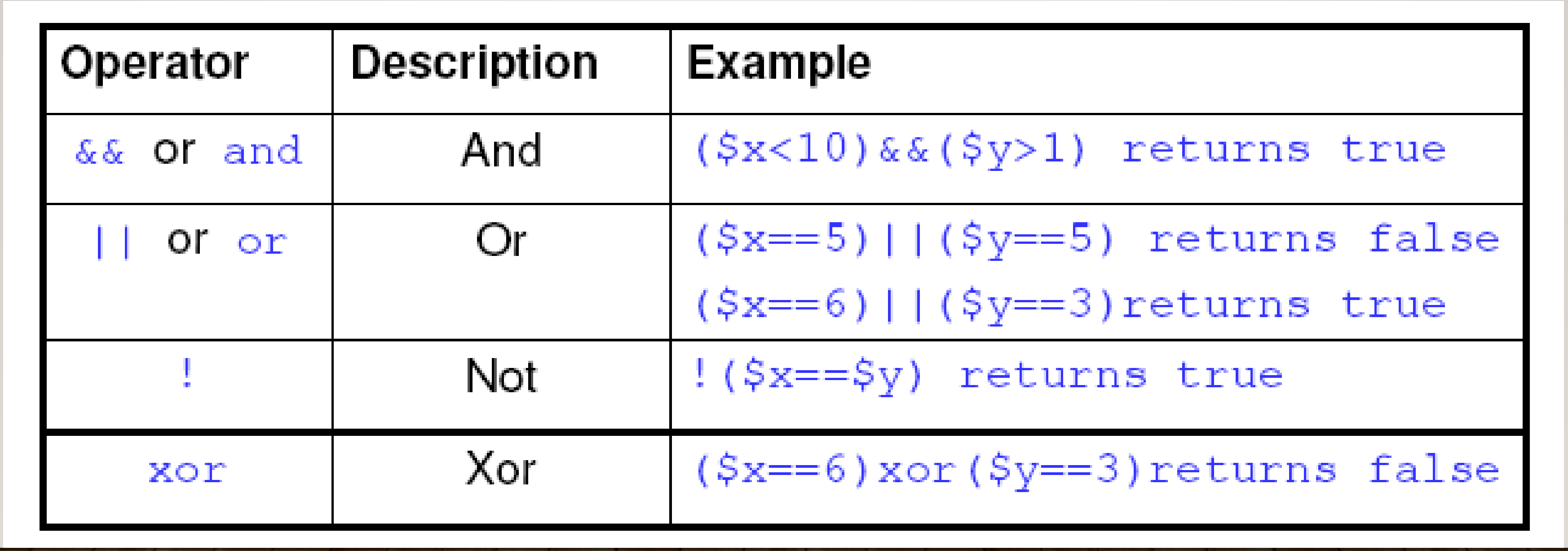
- A comparison operator compares its operands and returns a logical value based on whether the comparison is true or not



Operator	Description	Example
==	Is equal to	5==8 returns false
===	Exactly equal to (checks for both value and type)	If $x=5 and $y="5“, $x==$y returns true $x===$y returns false
!=	Is not equal to	5!=8 returns true
>	Is greater than	5>8 returns false
<	Is less than	5<8 returns true
>=	Is greater than or equal to	5>=8 returns false
<=	Is less than or equal to	5<=8 returns true

LOGICAL OPERATORS

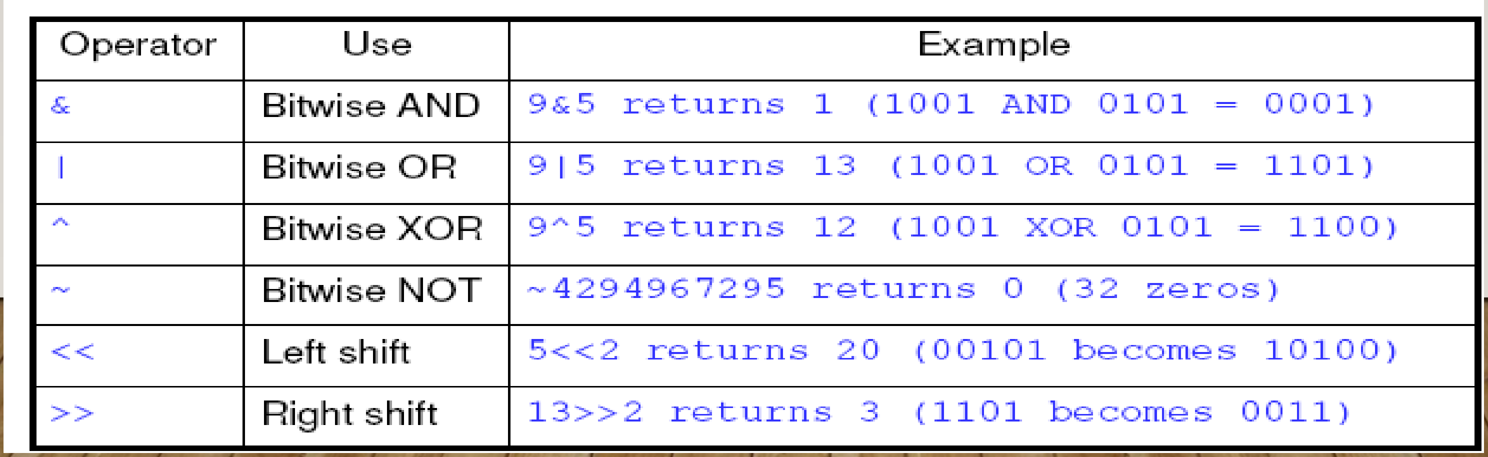
- Logical ~~operators take Boolean (logical) values as operands and return~~ a Boolean value.



Operator	Description	Example
&& or and	And	($x<10)&&($y>1) returns true
|| or or	Or	($x==5)\|\|($y==5) returns false ($x==6)\|\|($y==3)returns true
!	Not	!($x==$y) returns true
xor	Xor	($x==6)xor($y==3)returns false

BITWISE OPERATORS CONTINUE

Examples	Name	Result
$a & $b	AND	Bits that are set both $a and $b are set.
$a | $b	OR	Bits that are set in either $a or $b are set.
$a ^ $b	XOR	Bits that are set in $a or $b
~ $a	NOT	Bits that are set in $a are not set, and vice versa.
$a << $b	Shift Left	Shift the bits of $a, $b steps to the left. (each step means "multiply by two")
$a >> $b	Shift Right	Shift the bits of $a, $b steps to the right. (each step means "divide by two")



Operator	Use	Example
&	Bitwise AND	9&5 returns 1 (1001 AND 0101 = 0001)
|	Bitwise OR	9|5 returns 13 (1001 OR 0101 = 1101)
^	Bitwise XOR	9^5 returns 12 (1001 XOR 0101 = 1100)
~	Bitwise NOT	~4294967295 returns 0 (32 zeros)
<<	Left shift	5<<2 returns 20 (00101 becomes 10100)
>>	Right shift	13>>2 returns 3 (1101 becomes 0011)

CONSTANTS

- You must use the PHP's define() functions to create a constant which is subsequently cannot be changed unless use separately define it again.
- Syntax

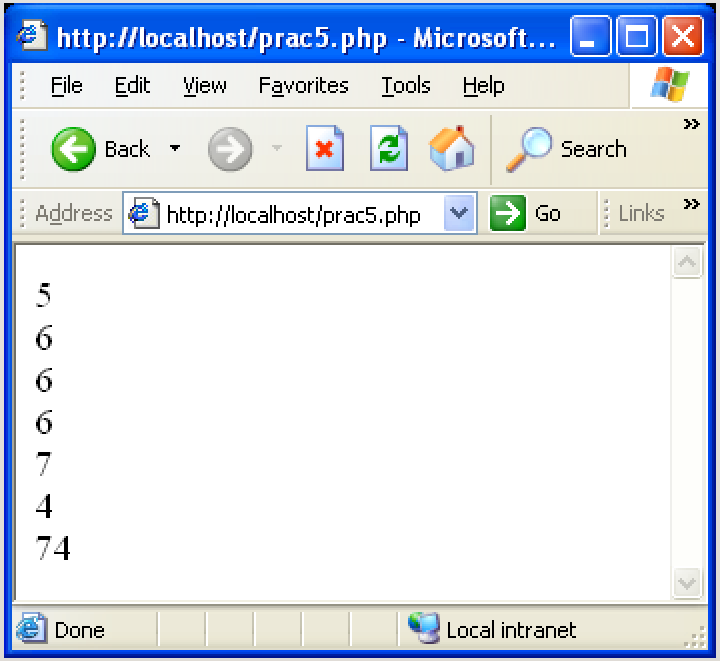
define(" your -constant-name" ,value);

E.g. define("age",25);

EXERCISE

```
<?PHP
$a = 5;
 echo $a++,"<br>";
echo $a--,"<br>";
$b = 5;
echo ++$b,"<br>";
 echo $b,"<br>";
 $a += 2;
 echo $a,"<br>";
$b -= 2;
 echo $b,"<br>";

$sentence_c = $a . $b;
 echo $sentence_c;
?>
```

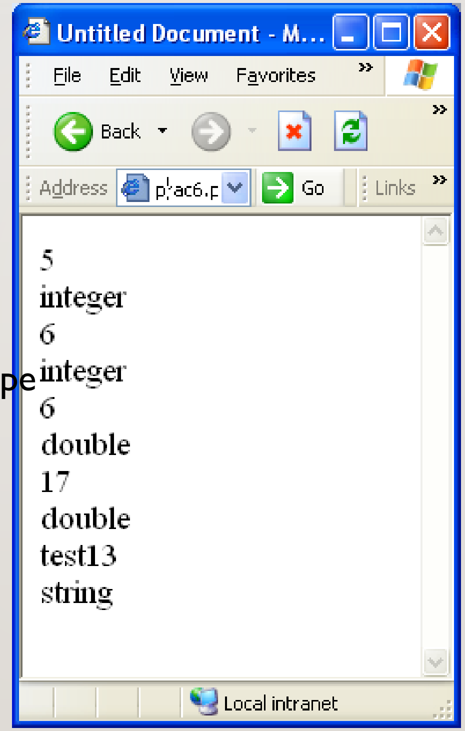


http://localhost/prac5.php - Microsoft...
File Edit View Favorites Tools Help
Back
Search
Address
http://localhost/prac5.php
Go
Links
Done
Local intranet

```
5
6
6
6
7
4
74
```

EXERCISE

```
<?php
$b = $a = 5;
$c = $a++;
echo $c,"<br>";
echo gettype($c),"<br>";
$e = $d = ++$b;
echo $e,"<br>";
echo gettype($e),"<br>";
$f = (double)($d++);//convert to another data type
echo $f,"<br>";
echo gettype($f),"<br>";
$g = (double)(++$e);
$h = $g += 10;
echo $h,"<br>";
echo gettype($h),"<br>";
$s="test";
$m = $s."13";
echo $m ,"<br>";
echo gettype($m),"<br>";
?>
```



```
5
integer
6
integer
6
double
17
double
test13
string
```