

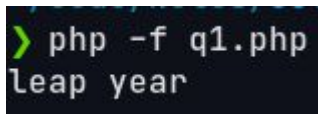
# Software Tools Lab 3

**SAHIL BONDRE: U18CO021**

**1. Write a PHP Program to check if a given year is leap year or not.**

```
<?php
$year = 2100;

if($year % 4 == 0) {
    if($year % 100 == 0 && $year % 400 != 0) {
        echo "not a leap year\n";
    } else {
        echo "leap year\n";
    }
} else {
    echo "not a leap year\n";
}
?>
```



```
> php -f q1.php
leap year
```

**2. Write a PHP Program to demonstrate use of Indexed arrays**

```
<?php
$a = array(45, 46, 47, 48);
echo "Accessing Index:\n";
echo $a[0]."\n";
echo $a[count($a) - 1]."\n";
echo "For Loop:\n";
for($i = 0; $i < count($a); ++$i) {
    echo $i.": ".$a[$i]."\n";
}
echo "For Each Loop\n";
foreach($a as $i) {
    echo $i."\n";
}
?>
```

```
> php -f q2.php
Accessing Index:
45
48
For Loop:
0: 45
1: 46
2: 47
3: 48
For Each Loop:
45
46
47
48
```

3. Write a PHP program to compute the sum of the digits of a number using function.

```
<?php
function countDigits($num) {
    $num = (string)$num;
    $sum = 0;
    for ($i = 0; $i < strlen($num); $i++){
        $sum += $num[$i];
    }
    return $sum;
}

echo countDigits(1234)."\n";
?>
```

```
> php -f q3.php
10
```

4. Write a PHP program to print out the following table

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50

```
<?php
for ($i=1; $i < 6; $i++) {
    for ($j=1; $j < 11; $j++) {
        echo $i * $j;
        echo " ";
    }
    echo "\n";
}
?>
```

```
> php -f q4.php
1 2 3 4 5 6 7 8 9 10
2 4 6 8 10 12 14 16 18 20
3 6 9 12 15 18 21 24 27 30
4 8 12 16 20 24 28 32 36 40
5 10 15 20 25 30 35 40 45 50
```

5. Write a PHP program to find the factorial of a number using a recursive function.

```
<?php
function factorial($n) {
    if($n <= 1) return 1;
    return $n * factorial($n - 1);
}

echo factorial(5)."\n";
echo factorial(4)."\n";
?>
```

```
> php -f q5.php
120
24
```

6. You need to write a PHP program to calculate electricity bills using if-else conditions.

Conditions:

For first 50 units – Rs. 3.50/unit

For next 100 units – Rs. 4.00/unit

For next 100 units – Rs. 5.20/unit

For units above 250 – Rs. 6.50/unit

You can use conditional statements.

```
<?php

function getBill($i) {
    $bill = 0;
    if ($i <= 50) {
        $bill = 3.50 * $i;
    } else if($i == 51 && $i <= 150) {
        $bill = 4.00 * $i;
    } else if($i = 151 && $i <= 250) {
        $bill = 5.20 * $i;
    } else {
        $bill = 6.50 * $i;
    }
    return $bill;
}

echo "Bill: ".getBill(32)."\n";
echo "Bill: ".getBill(76)."\n";
echo "Bill: ".getBill(150)."\n";
echo "Bill: ".getBill(300)."\n";
?>
```

```
> php -f q6.php  
Bill: 112  
Bill: 5.2  
Bill: 5.2  
Bill: 0
```

## 7. Write a simple calculator program in PHP using a switch case

```
<?php  
  
$a = 71;  
$b = 12;  
$op = "*";  
  
echo $a." ".$op." ".$b." = ";  
switch($op) {  
    case "+":  
        echo ($a + $b);  
        break;  
    case "-":  
        echo ($a - $b);  
        break;  
    case "*":  
        echo ($a * $b);  
        break;  
    case "/":  
        echo ($a/$b);  
}  
echo "\n";  
?>
```

```
> php -f q7.php  
71 * 12 = 852
```

## 8. Write a PHP Program to perform following Operations of a Matrix

a) Transpose of matrix

b) Multiplication of two matrix

c) Addition of two matrix

```
<?php
```

```
function printMatrix($m) {  
    foreach ($m as $a) {  
        foreach ($a as $i) {  
            echo $i." ";  
        }  
        echo "\n";  
    }  
    echo "\n";  
}
```

```
$m1 = array(  
    array(1, 2, 3),  
    array(4, 5, 6),  
    array(7, 8, 9)  
);  
echo "m1: \n";  
printMatrix($m1);
```

```
$m2 = array(  
    array(2, 5, 8),  
    array(6, 4, 7),  
    array(1, 7, 5)  
);  
echo "m2: \n";  
printMatrix($m2);  
$m3 = array(array());  
$m4 = array(array());
```

```
$row = 3;  
$col = 3;
```

```
// a
```

```
for($i = 0; $i < $row; $i++) {  
    for($j = 0; $j < $i; $j++) {
```

```
    $temp = $m1[$i][$j];
    $m1[$i][$j] = $m1[$j][$i];
    $m1[$j][$i] = $temp;
}
}
echo "transpose: \n";
printMatrix($m1);
```

// b

```
for ($i = 0; $i < $row; $i++) {
    for ($j = 0; $j < $col; $j++) {
        $m3[$i][$j] = 0;
        for ($k = 0; $k < $col; $k++)
            $m3[$i][$j] += $m1[$i][$k] * $m2[$k][$j];
    }
}
echo "multiplication: \n";
printMatrix($m3);
```

// c

```
for($i = 0; $i < $row; $i++) {
    for($j = 0; $j < $col; $j++) {
        $m4[$i][$j] = $m1[$i][$j] + $m2[$i][$j];
    }
}
```

```
echo "addition: \n";
printMatrix($m4);
?>
```

```
> php -f q8.php
```

```
m1:
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

```
m2:
```

```
2 5 8
```

```
6 4 7
```

```
1 7 5
```

```
transpose:
```

```
1 4 7
```

```
2 5 8
```

```
3 6 9
```

```
multiplication:
```

```
33 70 71
```

```
42 86 91
```

```
51 102 111
```

```
addition:
```

```
3 9 15
```

```
8 9 15
```

```
4 13 14
```

## 9. Write a PHP Program to demonstrate

- Functions with Default Argument Value
- Passing Arguments by Reference

```
<?php
function defaultVal($i = 0) {
    echo "You passed ".$i."\n";
}

function addOne(&$i) {
    ++$i;
}

defaultVal();
$x = 1;
```



```
echo "initial value: ".$x."\n";
addOne($x);
echo "final value: ".$x."\n";
?>
```

```
> php -f q9.php
You passed 0
initial value: 1
final value: 2
```

## 10. Write a PHP Program to demonstrate

- String Functions
- Math Functions
- Date and Time Functions

```
<?php
echo "String Functions:\n";
echo "1. Gets a character from a specified ASCII value: ";
echo chr(52)."\n";

echo "2. Split the string after each character and add a "." after each
split: ";
$str = "Hello world!";
echo chunk_split($str,1,"")."\n";

echo "3. Add a backslash in front of each double quote: ";
$str = addslashes('What does "yolo" mean?');
echo $str."\n";

echo "4. Add a backslash in front of the character W: ";
$str = addslashes("Hello World!", "W");
echo $str."\n";

echo "5. Gets the ASCII value of character: ";
echo ord("hello")."\n";

echo "6. Repeat the string Wow 3 times: ";
echo str_repeat("Wow ", 3)."\n";

echo "\nMath Functions:\n";
echo "1. Gets the absolute (positive) value of a number: ";
echo abs(6.7)."\n";
```

```
echo "2. Rounds a number up to the nearest integer: ";
echo ceil(5.1)."\\n";

echo "3. Gets the cosine of a number: ";
echo cos(3)."\\n";

echo "4. Gets hyperbolic cosine of a number: ";
echo cosh(3)."\\n";

echo "5. Gets the arc cosine of a number: ";
echo acos(0.64)."\\n";

echo "6. Round numbers down to the nearest integer: ";
echo floor(0.60)."\\n";

echo "\\nDate Functions:\\n";
echo "1. Check if several dates are valid Gregorian dates: ";
var_dump(checkdate(12,31,-400));

echo "2. Gets the default timezone: ";
echo date_default_timezone_get()."\\n";

echo "3. Add 40 days to the 15th of March, 2020: ";
$date=date_create("2020-03-15");
date_add($date,date_interval_create_from_date_string("40 days"));
echo date_format($date,"Y-m-d")."\\n";

echo "4. Gets a new DateTime object, and then format the date: ";
$date=date_create("2020-06-10");
echo date_format($date,"Y/m/d H:i:s")."\\n";

echo "5. Gets date/time information of the current local date/time: ";
$mydate=getdate(date("U"));
echo "$mydate[weekday], $mydate[month] $mydate[mday],
$mydate[year]"."\\n";

echo "6. Gets current time: ";
$mytime=gettimeofday();
echo "$mytime[sec].$mytime[usec]"."\\n";
?>
```

```
> php -f q10.php
String Functions:
1. Gets a character from a specified ASCII value: 4
2. Split the string after each character and add a space after each split: H.e.l.l.o. .w.o.r.l.d.!.
3. Add a backslash in front of each double quote: What does \"yolo\" mean?
4. Add a backslash in front of the character W: Hello \\World!
5. Gets the ASCII value of character: 104
6. Repeat the string Wow 3 times: Wow Wow Wow

Math Functions:
1. Gets the absolute (positive) value of a number: 6.7
2. Rounds a number up to the nearest integer: 6
3. Gets the cosine of a number: -0.98999249660045
4. Gets hyperbolic cosine of a number: 10.067661995778
5. Gets the arc cosine of a number: 0.87629806116834
6. Round numbers down to the nearest integer: 0

Date Functions:
1. Check if several dates are valid Gregorian dates: bool(false)
2. Gets the default timezone: UTC
3. Add 40 days to the 15th of March, 2020: 2020-04-24
4. Gets a new DateTime object, and then format the date: 2020/06/10 00:00:00
5. Gets date/time information of the current local date/time: Thursday, August 27, 2020
6. Gets current time: 1598530837.820093
```