WHY PHP?

- PHP is an acronym for "PHP: Hypertext Preprocessor“.
- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.).
- PHP is compatible with almost all servers used today (Apache, IIS, etc.).
- PHP supports a wide range of databases.
- PHP is easy to learn and runs efficiently on the server side.
- PHP is free to download and use.

PHP BASICS

1.PHP Code is written between <?php and ?> tags

```
<?php
echo "Hello world!";
?>
```

2.Every PHP statements ends with semicolon (;)

3. Variables:

- A variable starts with the $ sign, followed by the name of the variable
- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive ($age and $AGE are two different variables)

COMMENTED LINES IN PHP

```
// This is a single-line comment


# This is also a single-line comment


/*
This is a multiple-lines comment block
that spans over multiple
lines
*/
```

IF ELSE ELSEIF CONDITION

SYNTAX:

if (*condition*) {
  code to be executed if this condition is true;
} elseif (*condition*) {
  code to be executed if first condition is false and this condition is true;
} else {
  code to be executed if all conditions are false;
}

EXAMPLE:

```
<?php
$x=2;
if($x<2)
{
   echo “$x is less than 2”;
}
elseif($x>2)
{
   echo “$x is greater than 2”;
}
else
{
   echo “$x is equal to 2”;
 }
?>
```

SWITCH CONDITION

SYNTAX:

switch (*n*)

```
{
  case label1:
    code to be executed if n=label1;
    break;
  case label2:
    code to be executed if n=label2;
    break;
  case label3:
    code to be executed if n=label3;
    break;
    ...
  default:
    code to be executed if n is different from all labels;
}
```

EXAMPLE:

```
<?php
$n = 5;
switch ($n)
{
case 1:
    echo "CASE 1 IS SATISFIED";
    break;
case 2:
    echo "CASE 2 IS SATISFIED";
    break;
default:
    echo "DEFAULT CASE IS SATISFIED";
}
?>
```

LOOPS

Repeat a block of code as per requirement.

WHILE LOOP

SYNTAX:

while (*condition is true*)

{

  code to be executed;

}

EXAMPLE:

```
<?php
$x=1;
While($x<10)
{
echo "The number is: $x <br>";
$x=$x+1;
}
?>
```

DO WHILE LOOP

SYNTAX:

```
do
{
  code to be executed;
}
while (condition is true);
```

EXAMPLE:

```
<?php
$x=1;
Do
{
echo "The number is: $x <br>";
$x=$x+1;
}
While($x<10)
?>
```

FOR LOOP

SYNTAX:

for (*int counter; test counter; increment counter*)

```
 {
  code to be executed for each iteration;
}
```

EXAMPLE:

```
<?php
for ($x = 0; $x < 10; $x++) {
    echo "The number is: $x <br>";
}
?>
```

FOREACH LOOP

SYNTAX:

```
foreach ($array as $value)
{
  code to be executed;
}
```

EXAMPLE:

```
<?php
$colors = array("red", "green", "blue", "yellow");

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

BREAK:

The break statement can be used to jump out of a loop.

CODE:

```
<?php
for ($x = 0; $x < 10; $x++) {

  if ($x == 4) {
    break;
  }
  echo "The number is: $x <br>";
}
?>
```

OUTPUT:

```
The number is: 0
The number is: 1
The number is: 2
The number is: 3
```

CONTINUE:

The continue statement breaks one iteration (in the loop), if a specified condition occurs, and continues with the next iteration in the loop.

CODE:

```
<?php
for ($x = 0; $x < 10; $x++)
 {
  if ($x == 4)
 {
    continue;
  }
  echo "The number is: $x <br>";
}
?>
```

OUTPUT:

```
The number is: 0
The number is: 1
The number is: 2
The number is: 3
The number is: 5
The number is: 6
The number is: 7
The number is: 8
The number is: 9
```

SQL vs NoSQL

Index	SQL	NoSQL
1)	Databases are categorized as Relational Database Management System (RDBMS).	NoSQL databases are categorized as Non-relational or distributed database system.
2)	SQL databases have fixed or static or predefined schema.	NoSQL databases have dynamic schema.
3)	SQL databases display data in form of tables so it is known as table-based database.	NoSQL databases display data as collection of key-value pair, documents, graph databases or wide-column stores.
4)	SQL databases are vertically scalable.	NoSQL databases are horizontally scalable.
5)	SQL databases use a powerful language "Structured Query Language" to define and manipulate the data.	In NoSQL databases, collection of documents are used to query the data. It is also called unstructured query language. It varies from database to database.
6)	SQL databases are best suited for complex queries.	NoSQL databases are not so good for complex queries because these are not as powerful as SQL queries.
7)	SQL databases are not best suited for hierarchical data storage.	NoSQL databases are best suited for hierarchical data storage.
8)	MySQL, Oracle, Sqlite, PostgreSQL and MS-SQL etc. are the example of SQL database.	MongoDB, BigTable, Redis, RavenDB, Cassandra, Hbase, Neo4j, CouchDB etc. are the example of nosql database