

```
// Lesson : Lec 1
/*
- Http : the protocol used to communicate or send request and response between
browser and web server
    - the be other devices between them like routers , proxies , gateways
    - those devices can speed up the process if they saved the content locally
and it's called cache

- Web Server : Contains software that allows it to run Server-side programs/scripts
- Sends back their output as responses to web requests

- Php : Hypertext Preprocessor or Personal Home Page
    - make dynamic webpages , interact with database , Authenticate users , Process
form information

- Assignment operators : variable sign= value
    - you can append one string to another using .=

Note : if number is string can auto turn into int or float
Note : $x = "3.5f" + 1;
Note : var_dump($x); // output float but without +1 is a string
*/
```

```
// Lesson : Lec2
/*
- print vs echo
    - print : return value , use it in expressions
    - echo : don't return value , is purely a PHP language construct

- Unary operators : a++ , !
- Binary operators : takes two operands
- The one ternary operator : expr ? : , takes three operands

- Associativity
    - = is a operator's associativity : the process goes from rigth to left

- == تقارن القيمة فقط وتقوم بتحويل تلقائي للنوع

- Multiline Commands : use echo with one string on seperate lines to output
several lines

- explicit casting , type casting
    - (int) value and like that

- Magic constants : two underscores at the start and end

- global: to make the var accessable inside the function.
- static: would also like to keep its value for the next time the function is
called
*/
```

```

- Superglobals : are associative arrays and have useful info about the program
and env
- sanitize : To keep it save like : htmlentities()
*/

// Lesson : Lec3
/*
- print_r(var , true/false [opt] if you want to return the value not just print
it)
- function in PHP is used to print human-readable information about a
variable.

- printf(string, variables) // to format the output
- يمكن تحديد الدقة (%.2f) أو الحشو (%015.2f)
- sprintf(string, variables) function is similar to printf , but instead of
outputting the formatted string, it returns it.

- PHP uses standard Unix timestamps : used number of seconds
- time() : return the now time in seconds
- mktime(hour,minute,second,month,day,year) create timestamp , return in
seconds
- data(format, timesamp) : to format the timestamp
- checkdate(month , day ,year) : cheak if this date it true in the live

- The die and exit functions are used to terminate the execution of a script. and
the two are alias

- file handling
- file_exists("url") : return boll
- fopen("url", char) or die("message")
- $text = <<<_END
    Line 1
    Line 2
    Line 3
    _END; // to write 3 lines
- fwrite(fopen variable , $text) or die("message")
- fread($fopen variable , length) used with binary data
- fgets(%fopen variable , length)
- To update a file, you can open it in read/write mode (r+ or w+), read the
contents, and then write the new contents back to the file.
- fseek($fopen var , offset , SEEK_END)
- fclose($fopen variable)
- flock($fopen var , LOCK_EX/ for unlock : LOCK_UN) to lock file from
another's accesses and it's return bool
- copy("url1","url2") ; return bool
- rename("url1", url2) ; return bool , to moving
- unlink("url"); remove ; reutnr bool
- file_get_contents("url") to wread all the files and fetch file from server

- include "file" to run script from another file or include_once

```

- require "file" if the file not exist he will send erro but include not

```
Note : $a2[] = "Appened!"; # add string to end
*/
```

```
// Lesson : Lec4
```

```
/*
```

* Class: A class is a blueprint for objects, used to hold the objects along with their behaviors (functions) and properties.

* Object: An object in a class is an instance that has its own behaviors and properties. Objects can be related to the entities in real life.

* Encapsulation: Wrapping up of data and functions into a single unit.

* Inheritance: A class can inherit properties and methods from another class.

* Polymorphism: The ability to use a single interface for different data types.

- object = clone object2 : to clone objects : deep copy but the other name is shallow copy

- constructor : function __construct(){} : called when an object is created.

- destructor : function __destruct(){} : called when an object is destroyed.

- when we access property we use ->property without \$ of the property

- Declaring Constants : self::varname or self::fun() for static methods : static function name()

- called on a class rather than an object

- parent::method : it's like super

- final function name(){} : prevent fun from inheritance

- funciton_exists("functionname") : to check on function

- phpversion(): get the php version

```
*/
```

```
// Lesson : Lec5
```

```
/*
```

* Strings إلى MySQL. يتم إرسال الاستعلامات كـ

* Data Structure النتائج ترجع كـ PHP يمكن لـ

* تنسيق النتائج لعرضها في الويب PHP يمكن لـ

- PDO::ATTR_ERRMODE : Determines error reporting behavior

- PDO::ERRMODE_EXCEPTION : Throws PDOExceptions (recommended for development)

- PDO::ERRMODE_WARNING : Issues PHP warnings

- PDO::ERRMODE_SILENT : Only sets error codes (default)

- PDO::ATTR_DEFAULT_FETCH_MODE : Sets default fetch mode

```

- PDO::FETCH_ASSOC : Returns results as associative arrays (column name =>
value)
- PDO::FETCH_OBJ : Returns results as objects
- PDO::FETCH_BOTH : Returns both numeric and associative arrays (default)

- PDO::ATTR_EMULATE_PREPARES : When false, uses real prepared statements
-Setting to false improves security and handles data types properly

- $pdo->query($query)
*/

// Lesson : String functions
/*
* strlen(var)
* strpos(var , value)
* strcmp(var, value); compare to , return boolean
    - ترجع 0 إذا متساويين، رقم موجب إذا الأول أكبر، سالب إذا الثاني أكبر.
    - strcasecmp (case-insensitive) لا يراعي حالة الأحرف.
* strtolower(var), strtoupper(var)
* trim(var , the striped character [opt])
* explode(sep , value), implode(sep , value) = split , join
*/

// Lesson : Array functions
/*
* array_shift(array) : Removes the first element from an array and returns it.
* array_pop(array) : Removes the last element from an array and returns it.
* array_push(array , value) : Adds one or more elements to the end of an array.
* array_reverse(array) : Returns an array with elements in reverse order.
* sort(array) : Sorts an array in ascending order.
* array_slice(array , start , length of the returned array) : Extracts a slice of
the array, returning the selected elements.
* array_merge(array , arrays..) : Merges one or more arrays into one array.
* array_diff(array , another array ) : Computes the difference of arrays : what
in the first array not in the last array
* array_intersect(array , another) : Computes the intersection of arrays.
* array_map(function($e){},array) : Applies a callback function to the elements
of the given arrays.
* array_filter(array, function($e)) : Filters elements of an array using a
callback function.
* array_reduce(array, function($c,$e)) : Iteratively reduces the array to a
single value using a callback function.
* count(array) : length
*/

```