

a.s. 2012-13	ISTITUTO TECNICO INDUSTRIALE STATALE “E. Fermi” – Frascati (RM) Indirizzo Informatica “Abacus” Laboratorio di Informatica	Codice documento
		LABINF.01

PHP Quick Reference

Commenti	
singola riga	<i>// riga di commento</i>
	<i># riga di commento</i>
più righe	<i>/* commento su piu' righe */</i>

Tipi di dato	
integer	array
float	object
boolean	resource
string	NULL

Operatori aritmetici	
+	Addizione
-	sottrazione
*	Moltiplicazione
/	Divisione
%	Modulo

Operatori logici	
&&	Prodotto logico
	Somma logica
!	Negazione
xor	Somma logica esclusiva

Operatori relazionali	
<	Minore
>	Maggiore
<=	Minore o uguale
>=	Maggiore o uguale
==	Uguale
===	Uguale con comparazione di tipo
!=	Diverso
<>	Diverso
!==	Diverso con comparazione di tipo

Operatori aritmetici abbreviati			
=	Assegnazione	&=	And
+=	Somma	=	Or
-=	Concatenazione	^=	Or esclusivo
*=	Prodotto	<<=	Shift sinistro
/=	Quoziente	>>=	Shift estro
%=	Modulo		

Strutture di controllo		
Selezione semplice	if (<i>condizione</i>) { <i>istruzione</i> ; }	
Selezione doppia	if (<i>condizione</i>) { <i>istruzione</i> ; else <i>istruzione</i> }	Op. ternario <i>condizione</i> ? <i>istr1</i> : <i>istr2</i> ;
Selezione multipla	if (<i>condizione</i>) { <i>istruzione</i> ; } elseif (<i>condizione</i>) { <i>istruzione</i> ; } else { <i>istruzione</i> ; }	
	switch (<i>espressione</i>) { case <i>valore</i> : <i>istruzione</i> ;[break]; case <i>valore</i> : <i>istruzione</i> ;[break]; case <i>valore</i> : <i>istruzione</i> ;[break]; ... default : <i>istruzione</i> ; }	

Schema di una funzione
function *nome* (*lista_parametri*)
{ *istruzioni*;
[**return** *espressione*;]
}

Gestione Eccezioni
try
{
istruzioni; //genera eventuali eccezioni
}
catch (*ExceptionClass \$nomeeccezione*)
{
istruzioni; //trattamento eccezioni
}

a.s. 2012-13	ISTITUTO TECNICO INDUSTRIALE STATALE “E. Fermi” – Frascati (RM) Indirizzo Informatica “Abacus” Laboratorio di Informatica	Codice documento
		LABINF.01

Strutture di controllo	
Iterazione predefinita	for (<i>contatore</i> = <i>valore_iniziale</i> ; <i>condizione</i> ; <i>incremento/decremento</i>) { <i>istruzione</i> ; } foreach (<i>array</i> as [<i>valore</i> <i>indice</i> => <i>valore</i>]) { <i>istruzioni</i> ; [break]; [continue]; }
Iterazione precondizionata	while (<i>condizione</i>) { <i>istruzione</i> ; }
Iterazione .. .	do { <i>istruzione</i> ; }

Variabili	
<i>\$nome</i> = <i>valore</i> ;	Inizializza la variabile <i>nome</i> con <i>valore</i>
<i>\$nome</i> = & <i>\$nome</i> ;	Assegnazione del riferimento
Array	
<i>\$nome</i> = array ();	Dichiarazione array <i>nome</i>
<i>\$nome</i> = array (<i>val1</i> , <i>val2</i> , <i>val3</i>);	Inizializza l'array <i>nome</i>
<i>\$nome</i> = array (<i>i1</i> = <i>val1</i> , <i>i2</i> = <i>val2</i> ...);	Inizializza l'array <i>nome</i> ponendo <i>val1</i> in posizione <i>i1</i> , <i>val2</i> in posizione <i>i2</i> ,

Matrici	
<i>\$nome</i> = array (<i>ind</i> => array (<i>val1</i> , <i>val2</i>));	Dichiara un array multidimensionale
<i>\$nome</i> = array (<i>ind</i> => <i>val</i> , <i>ind1</i> => array (<i>val</i> , <i>val</i>));	Crea una matrice con specificati range di righe e colonne.

Funzioni matematiche	
abs (<i>numero</i>)	Restituisce il valore assoluto di un <i>numero</i> (valore senza segno)
acos (<i>numero</i>)	Restituisce l'arcoseno dell'angolo
asin (<i>numero</i>)	Restituisce l'arcoseno dell'angolo
atan (<i>numero</i>)	Restituisce la funzione inversa della tangente
ceil (<i>numero</i>)	Restituisce l'intero per eccesso del valore razionale <i>numero</i>
cos (<i>numero</i>)	Restituisce il coseno di un angolo.
dechex (<i>numero</i>)	Converte <i>numero</i> da decimale a esadecimale
decibin (<i>numero</i>)	Converte <i>numero</i> da decimale a binario
decoct (<i>numero</i>)	Converte <i>numero</i> da decimale a ottale
exp (<i>numero</i>)	Restituisce e (base dei logaritmi naturali) elevato a <i>numero</i> .
hexdec (<i>numero</i>)	Converte <i>numero</i> da esadecimale a decimale
int (<i>numero</i>)	Restituisce la parte intera di <i>numero</i> .
log (<i>numero</i>)	Restituisce il logaritmo naturale di <i>numero</i> .
log10 (<i>numero</i>)	Restituisce il logaritmo decimale di <i>numero</i> .
<i>A</i> % <i>b</i>	Dà il resto della divisione tra <i>a</i> e <i>b</i>
pow (<i>numero</i> , <i>esponente</i>)	Restituisce la potenza di <i>numero</i> elevato a <i>esponente</i> ..
rand ([<i>min</i> , <i>max</i>])	Restituisce un valore casuale, eventualmente indicando l'intervallo tra <i>min</i> e <i>max</i>
round (<i>numero</i>)	Arrotonda <i>numero</i> all'intero più vicino
sin (<i>numero</i>)	Restituisce il seno di un angolo.
sqrt (<i>numero</i>)	Restituisce la radice quadrata di <i>numero</i> .
tan (<i>numero</i>)	Restituisce la tangente di un angolo

a.s. 2012-13	ISTITUTO TECNICO INDUSTRIALE STATALE “E. Fermi” – Frascati (RM) Indirizzo Informatica “Abacus” Laboratorio di Informatica	Codice documento
		LABINF.01

Funzioni per variabili e costanti	
define (<i>nome</i> , <i>valore</i> [, <i>case-sens</i>])	Definisce la costante <i>nome</i> e <i>valore</i> .
defined (<i>x</i>)	Controlla se esiste la costante. Restituisce true o false .
max (<i>x</i> , <i>y</i> , <i>z</i> ,)	Confronta più variabili e restituisce quella di valore massimo.
min (<i>x</i> , <i>y</i> , <i>z</i> ,)	Confronta più variabili e restituisce quella di valore minimo.
echo (<i>variabile</i>);	Stampa <i>variabile</i>
empty (<i>variabile</i>)	Controlla se <i>variabile</i> è nulla (stringa vuota, valore zero, variabile non definita o di valore NULL). Risponde con true o false .
isset (<i>variabile</i>)	Controlla se la variabile è definita (se è inizializzata o se diversa dal valore NULL). Risponde con true o false .
is_null (<i>variabile</i>)	Controlla se la variabile è NULL (errore 'notice' se la variabile non è definita.). Risponde con true o false .
is_int (<i>variabile</i>) is_integer (<i>variabile</i>)	Controlla se la variabile è di tipo intero. Risponde con true o false .
is_long (<i>variabile</i>)	Controlla se <i>variabile</i> è intero lungo. Risponde con true o false .
is_float (<i>variabile</i>)	Controlla se <i>variabile</i> è numerica. Risponde con true o false .
is_double (<i>variabile</i>)	Controlla se <i>variabile</i> è numerica. Risponde con true o false .
is_real (<i>variabile</i>)	Controlla se <i>variabile</i> è numerica. Risponde con true o false .
is_string (<i>variabile</i>)	Controlla se <i>variabile</i> è una stringa. Risponde con true o false .
is_array (<i>variabile</i>)	Controlla se <i>variabile</i> è un array. Risponde con true o false .
is_numeric (<i>variabile</i>)	Controlla se <i>variabile</i> è alfanumerica. Risponde con true o false .
gettype (<i>variabile</i>)	A seconda di <i>variabile</i> , dà <i>boolean</i> , <i>integer</i> , <i>double</i> , <i>string</i> , <i>array</i> .
settype (<i>variabile</i> , <i>tipo</i>)	Imposta <i>variabile</i> a <i>tipo</i>
var_dump (<i>variabile</i>)	Restituisce al browser informazioni su <i>variabile</i> (nome e contenuto)
print_r (<i>variabile</i>)	Stampa <i>variabile</i> a video. Risponde con true o false .
unset (<i>variabile</i>)	Elimina la variabile. Non restituisce valori.

Funzioni stringa	
. (<i>punto</i>)	Concatenazione di stringhe
bin2hex (<i>stringa</i>)	Restituisce il codice esadecimale dei caratteri corrispondenti alle lettere della <i>stringa</i> .
chr (<i>codicecar</i>)	Restituisce il carattere associato al <i>codicecar</i>
explode (<i>str1</i> , <i>str2</i> [<i>numero</i>])	Converte la stringa in un array, i cui elementi sono le sottostringhe di <i>str2</i> separate da <i>str1</i> (che può essere uno spazio). <i>numero</i> (opzionale) indica quanti elementi può contenere l'array; se inferiore al numero di elementi, l'ultimo conterrà il resto della stringa. Restituisce un array.
floatval (<i>stringa</i>)	Converte una stringa in numero in doppia precisione
intval (<i>stringa</i>)	Converte <i>stringa</i> in numero
is_numeric (<i>stringa</i>)	Restituisce true se <i>stringa</i> rappresenta un numero
ltrim (<i>stringa</i>)	Restituisce la stringa eliminando eventuali spazi a sinistra.
rtrim (<i>stringa</i>)	Restituisce la stringa eliminando eventuali spazi a destra.
strcmp (<i>str1</i> , <i>str2</i>)	Restituisce un valore che indica il risultato di un confronto di stringhe
strlen (<i>stringa</i>)	Restituisce il numero di caratteri che compongono una stringa.
strpos (<i>stringa1</i> , <i>stringa2</i>)	Restituisce la posizione della prima occorrenza di <i>stringa2</i> all'interno di <i>stringa1</i> , oppure false .
strstr (<i>stringa1</i> , <i>stringa2</i>)	Controlla se <i>stringa2</i> è contenuta in <i>stringa1</i> . Se presente, restituisce la sottostringa che inizia da <i>stringa2</i> . Se <i>stringa2</i> non è presente restituisce false .
strtolower (<i>stringa</i>)	Converte <i>stringa</i> in minuscolo.
strtoupper (<i>stringa</i>)	Converte <i>stringa</i> in maiuscole
strval (<i>numero</i>)	Restituisce una rappresentazione <i>numero</i> in forma di stringa
substr (<i>stringa</i> , <i>inizio</i> [, <i>num</i>])	Restituisce <i>num</i> caratteri di <i>stringa</i> , partendo dalla posizione <i>inizio</i> (se <i>inizio</i> è -1 ci troviamo sull'ultimo carattere)
trim (<i>stringa</i>)	Restituisce la stringa eliminando eventuali spazi prima e dopo.

a.s. 2012-13	ISTITUTO TECNICO INDUSTRIALE STATALE “E. Fermi” – Frascati (RM) Indirizzo Informatica “Abacus” Laboratorio di Informatica	Codice documento
		LABINF.01

Funzioni su array	
array_key_exists (<i>key</i> , <i>array</i>)	Controlla se <i>key</i> è presente tra le chiavi di <i>array</i> . Restituisce true o false .
array_pop (<i>array</i>)	Toglie da <i>array</i> l'ultimo elemento e lo restituisce.
array_push (<i>array</i> , <i>val1</i> , <i>val2</i> ...)	Unisce più valori ad un <i>array</i> . Restituisce il numero degli elementi che compone il nuovo <i>array</i> .
array_shift (<i>array</i>)	Toglie da <i>array</i> il primo elemento e lo restituisce.
array_search (<i>val</i> , <i>array</i>)	Se <i>val</i> è presente in <i>array</i> , restituisce la chiave, oppure false .
array_reverse (<i>array</i> , true false)	Inverte l'ordine degli elementi di <i>array</i> . Se impostiamo true confermiamo le chiavi dell' <i>array</i> originarie. Restituisce un <i>array</i> .
array_unshift (<i>array</i> , <i>val1</i> , <i>val2</i> ,...)	Immette i <i>val1</i> , <i>val2</i> , ...all'inizio di <i>array</i> . Restituisce il numero degli elementi che compone il nuovo <i>array</i> .
arsort (<i>array</i>)	Ordina le chiavi degli elementi in ordine decrescente (non le modifica).
asort (<i>array</i>)	Ordina le chiavi degli elementi in ordine crescente (non le modifica).
count (<i>array</i>)	Conta il numero di elementi di <i>array</i> . Restituisce un numero.
current (<i>array</i>)	Restituisce il primo valore dell' <i>array</i> . Restituisce una stringa.
in_array (<i>val</i> , <i>array</i>)	Controlla se <i>val</i> è presente in <i>array</i> . Restituisce true o false .
is_array (<i>array</i>)	Controlla se <i>array</i> è una variabile <i>array</i> . Restituisce una stringa.
rsort (<i>array</i>)	Modifica l' <i>array</i> originario ordinando gli elementi con chiavi numeriche in ordine decrescente partendo da zero (resetta gli indici)
sizeof (<i>array</i>)	Restituisce il numero degli elementi nell' <i>array</i> .
sort (<i>array</i>)	Modifica l' <i>array</i> originario ordinando gli elementi con chiavi numeriche in ordine crescente partendo da zero (resetta gli indici)
implode (<i>stringa</i> , <i>array</i>)	Funzione opposta di explode() , e serve a riunire in un'unica stringa i valori dell' <i>array</i> . La <i>stringa</i> indicata come primo parametro è considerata come <i>separatore</i> . Restituisce la stringa risultato dell'aggregazione

Funzioni data e ora	
date ("j/n/Y")	Restituisce la data corrente di sistema (Y - anno su 4 cifre, y - anno su 2 cifre)
date ("j")	Restituisce un numero intero compreso tra 1 e 31 inclusi che rappresenta il giorno del mese della <i>data</i> (d - giorno del mese su due cifre, j - giorno del mese senza lo zero)
date ("F")	Restituisce un numero intero tra 1 e 12 inclusi, che rappresenta il mese dell'anno di <i>data</i> (n - mese numerico senza lo zero, m - mese numerico su 2 cifre, F - mese testuale in lingua inglese M - mese testuale su 3 lettere in inglese)
date ("H : i : s")	Restituisce l'ora di sistema corrente (H - ora su due cifre , G – ora senza zero iniziale, i - minuti su due cifre, s - secondi su due cifre)
date ("w")	Restituisce un numero intero che rappresenta il giorno della settimanam, (w - giorno della settimana, numerico (0=dom, 6=sab), I - giorno della settimana testuale in inglese, D - giorno della settimana su 3 lettere in inglese, d – numero del giorno del mese)
I separatori (: - / .)	Vanno utilizzati a piacimento, in quanto influenzano solo l'estetica. Provare ("H<j/Y-i#D").
mktime (<i>h</i> , <i>m</i> , <i>s</i> , <i>mm</i> , <i>gg</i> , <i>aa</i>)	Fornisce il <i>timestamp</i> di una data ben definita (inserire solo numeri interi). Per fare calcoli possiamo inserire numeri superiori a quelli usuali ad esempio se inseriamo 20 nei mesi verrà interpretato come 12+8, fornendo il <i>timestamp</i> dell'agosto dell'anno successivo. Restituisce un numero.
time ()	Restituisce il <i>timestamp</i> attuale (il numero di secondi dal 01/01/1970)

a.s. 2012-13	ISTITUTO TECNICO INDUSTRIALE STATALE “E. Fermi” – Frascati (RM) Indirizzo Informatica “Abacus” Laboratorio di Informatica	Codice documento
		LABINF.01

Variabili superglobali	
\$GLOBALS	Accesso a tutte le variabili globali nello script
\$ _SERVER	Accesso alle variabili del web server
\$ _GET	Valori passati allo script tramite URL
\$ _POST	Valori passati allo script tramite post (HTTP)
\$ _COOKIE	Valori passati dal cookie dell'utente
\$ _FILES	Valori passati tramite HTTP Post File Uploads
\$ _ENV	Valori passati allo script tramite l'ambiente
\$ _REQUEST	Valori passati attraverso URL, HTTP Post, Cookies dell'utente
\$ _SESSION	Valori passati tramite la sessione utente

Server	
\$ _SERVER['SERVER_SOFTWARE']	Il server
\$ _SERVER['SERVER_NAME']	Nome del server
\$ _SERVER['SCRIPT_FILENAME']	Percorso e nome dello script
\$ _SERVER['SCRIPT_NAME']	Nome dello script
Informazioni del client	
\$ _SERVER['REQUEST_METHOD']	Riporta se si era scelto GET o POST
\$ _SERVER['REMOTE_ADDR']	Indirizzo browser richiedente (è l'isp con si è connessi)
Variabili_server_php	
HTTP_COOKIE_VAR['PHPSESSID']	Contiene il valore dell'ID di sessione
HTTP_SERVER_VAR['HTTP_COOKIE']	Contiene il valore di tutti i cookie
HTTP_SERVER_VAR['HTTP_HOST']	Contiene il nome dell'host su quale risiede il server web
HTTP_SERVER_VAR['REMOTE_ADDR']	Contiene l'indirizzo remoto del browser (sicuramente dell'isp)
HTTP_SERVER_VAR['SCRIPT_FILENAME']	Contiene il nome e il percorso completo dello script corrente
HTTP_SERVER_VAR['SERVER_ADMIN']	Dà l'indirizzo e-mail inserito nel file di configurazione
HTTP_SERVER_VAR['SERVER_NAME']	Contiene il nome del server web
HTTP_SERVER_VAR['SERVER_SOFTWARE']	Contiene il nome del server web

Cookies	
bool setcookie (<i>name, value[, expire][,path]</i> <i>[, domain]</i>);	Crea il cookie <i>nome</i> , con <i>valore</i> e visibile per una certa <i>durata</i> (sec). Facoltativi sono il path e il dominio in cui deve risultare visibile
\$ _COOKIE['nome'];	Restituisce il valore del cookie avente <i>nome</i>
print_r (\$ _COOKIE);	visualizza tutti i cookie

Sessioni	
session_start();	Crea una sessione
\$ _SESSION["variabile"] = valore;	Imposta la variabile di sessione
\$ _SESSION['sessione'];	Recupera il valore dalla variabile di sessione
session_destroy();	Chiude la sessione
session_unset()	Recupera tutte le variabili di sessione
session_name()	Restituisce o imposta il nome della sessione

a.s. 2012-13	ISTITUTO TECNICO INDUSTRIALE STATALE “E. Fermi” – Frascati (RM) Indirizzo Informatica “Abacus” Laboratorio di Informatica	Codice documento
		LABINF.01

```
[var | <modificatori1>] [<classe membro variabile>];
[<modificatori*>] function <nomefunzione>([<parametri>])
{
<azioni>;
}
}
```

Dichiarazione e utilizzo di una Classe:

```
$variable = new nomeclasse();
$variable->nomefunzione();
nomeclasse::nomefunzione(); (Chiamata statica)
```

Utilizzo di PHP-MySQL (nuova libreria **mysqli**)

Salvare come “funzioni_mysql.php”

```
<?php
class MysqlClass
{
    protected $mysqli=null; // attributo
    private $attiva = false; // controllo connessioni attive

    public function __construct(mysqli $mysqli)
    { $this->mysqli=$mysqli; }

    public function connetti() // connessione a MySQL
    {
        if (mysqli_connect_errno()) // verifica connessione
        { // notifica in caso di errore
            echo "Errore in connessione al DBMS: ".mysqli_connect_error();
            exit(); // interruzione per errore
        }
        else
        { // notifica in caso di connessione attiva
            echo "Connessione avvenuta con successo";
        }
    }

    public function disconnetti()// chiusura della connessione
    {
        if($this->attiva)
        {
            if($this->mysqli->close())
            {
                $this->attiva = false;
                return true;
            }
            else { return false; }
        }
    }

    public function provaquery($sql)// query
    {
        $result = $this->mysqli->query($sql) or // esecuzione query
```

¹ I modificatori <public | private | static> sono implementati in PHP5
M.Malatesta – 29/11/2015 - PHP_Quick_Reference-06.doc

a.s. 2012-13	ISTITUTO TECNICO INDUSTRIALE STATALE “E. Fermi” – Frascati (RM) Indirizzo Informatica “Abacus” Laboratorio di Informatica	Codice documento
		LABINF.01

```

        die ($this->mysqli->error);
    if($result->num_rows >0) // numero record presenti
    {
        // generazione di un array numerico
        while($row = $result->fetch_array(MYSQLI_NUM))
        {
            echo "<BR>".$row[0]." ".$row[1];
        }
    }
    // liberazione delle risorse occupate dal risultato
    $result->close();
}

public function createTable ($sql)// query
{
    $result = $this->mysqli->query($sql) or // esecuzione query
    die ($this->mysqli->error);
}
} // end class
?>

File connessione.php
<?php
include "funzioni_mysql.php";
// istanza della classe
$nomehost = "localhost";
$nomeuser = "root";
$password = "";
$db = "agenti";
$mysqli = new mysqli($nomehost, $nomeuser, $password, $db);
$data = new MysqlClass($mysqli);
$data->connetti(); // funzione di connessione
/* esempi di query
$query = "SELECT .....";
$query="CREATE TABLE test (id INT UNSIGNED AUTO_INCREMENT NOT NULL,
title VARCHAR(32) NOT NULL,content TEXT NOT NULL,PRIMARY KEY(id));
";
*/
/*
esecuzione della query
$data->provaquery($query);
$data->createTable($query);
*/
$data->disconnetti(); // funzione di disconnessione
echo "<BR>disconnessione avvenuta!";
?>

```