* HTML - PHP - SQL AND BACKEND DATABASE IN WEB PROGRAMMING

II SEM

IMP QUESTIONS

Q.1) What do you mean by client side and server side scripting?

Ans CLIENT SIDE - SCRIPTING :

1) A technique used in web development that involves scripts that run of clients machine browser.

2) Executed in the client side on web browser.

3) It does not provides much security to data

4) Eg; HTML, CSS, JS

SERVER SIDE - SCRIPTING :

1) A technique used in web development that involves using scripts on webserver to produce response which is customized on the basis of clients request to the website.

2) Executed at the backend of the web server.

3) It provides more security and privacy as compared to client side scripting.

4) Eg; PHP, Python
Ruby, Java

Q-2) What are variables? How do you use variables in PHP?

Ans VARIABLE:

→A variable is an abstract storage location paired with a name which contains some quantity of data or object referred to as a value.

→Every variable has a datatype depending upon its declaration and defination.

1) In PHP, variables are case sensitive and expressions are terminated by semicolon.

DENOTION:

2) Each and every variable is denoted with dallar sign.

3) DECLARATION: Variables are declared and defined by assigning value to it.

4) ASSIGNMENT: Values are assigned to the variables by using '=' operator

Eg; `$var = "Value"`

5) VALUE:
Value specifies the datatype of that variable.

6) Variables in PHP do not need declaration before assignment.

7) USAGE:
Variables used before assignment have default values.
The value of the variable is the value of its most recent assignment.

8) Variables are always used and accessed using '$' sign.
Dot operator (.) is used to concatenate the variables in the echo and print statements.

Q.3) What are the different datatypes in PHP?

Ans DATATYPES: Datatypes are determined by the type and nature of value assigned to a variable.

1) Datatype in PHP are streamlined and flexible.

2) Datatypes of variables can be directly ~~is~~ determined during assignment and no need of declaration.

3) Thus, PHP automatically converts datatypes of variables according to the assignment and requirement.

4) PHP has total 8 datatypes as follows:

INTEGERS :
(Whole number without decimal point)
DOUBLES :
(Floating point number)
BOOLEAN :
(True or False)
NULL :
(Only one value : NULL)
STRINGS :
(Sequence of characters)
ARRAYS :
(Named and indexed collection)
OBJECTS :
(Instances of programmer-defined class)
RESOURCES :
(special variables holding references)

5) Out of these, integers, doubles, booleans and null and strings are the simple datatypes.

INTEGERS :
It is the simplest datatype which holds positive and negative whole numbers.

Eg;

```
                                   VALUE AS
$ int var1 = 12340;              (NUMBER)
$ int var2 = -123+456;           (EXPRESSION)
```

DOUBLES :
It is the floating point datatype which holds positive and negative whole numbers

Eg;

```
$ doublevar = 65.367 ;
```

STRINGS :
Sequence of characters specified under single or double quotes.

Eg;

```
$str1 = "String in double quotes";
$str2 = 'String in single quotes';
$str = " "; // Empty string.
```

Q.4) Explain for, while and do while loops in PHP.

Ans **WHILE LOOP :**

1) A while loop in PHP is a looping statetment which is used to check for a condition and iterate over the loop body until the condition is true.

2) SYNTAX;

```
while (condition)
{
    statements ;
}
```

3) It creates a loop that evaluates an expression and executes a block of statements. The loop then repeats as longer as the specified condition is true.

4) Eg;

```
$count =1;
while ($count < = 10)
{
  print ("count is $count <br>");
  $count = $cout+1;
}
```

It will print exactly 10 lines.

DO-WHILE LOOP :

1) Similar to while loop a do while loop also checks for a

condition and iterate over the loop body until the condition is true.

2) SYNTAX ;

```
do
{
   statement 1
   stament n
} while (condition) ;
```

3) But it creates a loop where if statement is false ~~will~~ statement will execute atleast once.

4) Eg;

```
$count = 45;
do
{
  print ("count is $count <br>");
  $count = $count +1;
} while ($count < = 10);
```

It will print a single line "count is 45" because condition is false.

FOR LOOP :

1) A for loop in a javascript is a looping statement which is used to iterate a statement or a part of program several times.

2) SYNTAX;

```
for(        ;        ;        )
{           ↑        ↑        ↑
      INITIALI-  TERMINATION  LOOP-END-
      ZATION     CHECK         -EXPRESSION

    // LOOP BOdy;
}
```

3) It creates a loop that consists of 3 optional expressions enclosed in parenthesis, separated by semicolon and followed by a block of statements.

4) Eg;

```
$n = 10
for($i=0; $i<=$n; $i++)
{
    echo "<br>" . $i;
}
```

The program will print 0 to 10.

Q.5) What are advantages of server side scripting?

Ans 1) Cost efficient
2) Easy to use
3) HTML - embeddedness
4) Cross-platform compatibility
5) Stabilitly
6) Fast feature development
7) Not proprietary.

SERVER TO CLIENT :
Webpages can be assembled from backend - server output.
CLIENT TO SERVER :
Customer - entered information can be acted upon.

Q.6) Explain break and continue statements and exit function in PHP.

Ans BREAK AND CONTINUE :

1) Break and continue are the control keywords in PHP; basically used for manipulating looping statements.
2) These are the standard ways to get exit from the looping structure.

BREAK : It exits from the innermost loop that contains break.
CONTINUE : It skips to the end of the current iteration of the innermost loop.

3) Eg;

```
for{$i=1; $i<10; $i++)
{
    if ($i%2 != 0)
        break;
    else
        print ($i);
  }
```

The program will print nothing because i is odd number and loops terminates completely.

```
Eg; for($i=1 ;$i<10 ; $i++)
  {
   if ($i%2!=0)
   {
     continue;
   }
   else
     print ($i);
  }
 }
```

The program will print 2,4,6,8 and skip odd numbers.

Exit()

1) Exit is a construct and not a function.
2) It takes a single argument and prints that value and terminates the execution of that script.
3) It is used to get terminate the entire PHP script.

4) Exit function can also be used without argument and parenthesis.

Q.7) Explain if-else statement in PHP.

Ans 1) If - Else statements are the decision making statements in PHP.

2) If - Else- If ladder is used to check multiple conditions. Only one condition is true if not then else part is executed.

SYNTAX;

```
3)        if (condition1)
          {                          ) IF TRUE
              statements;  <-
          }
          else if (condition 2)
          {                          ) IF TRUE
              statements;  <-
          }    IF BOTH FALSE
          else  ->
          {      statements;
          }
```

4) If test is false and there is no else part then the next statements after the if-else block will be executed.

5) The result of an if-else block is interpretted as a boolean value

6) The if-else ladder in PHP is also called as cascading sequence of test.

Eg;

```
        $n= $ GET ["num"]  ANY INTEGER
  if ($n >0)
        print (" POSITIVE") ;
else if ($n < 0)
        print (" NEGATIVE") ;
 else
      print ("ZERO") ;
```

7) An if part containing another if-else block as a statement if called as nested-if structure in PHP.

Q.5) Explain Switch-Case statements in PHP.

Ans 1) In PHP, Switch Case statements are used for checking multiple conditions each for a specific type of input.

2) Here case is the value of variable or expression which is checked by using

Switch-case statements.

3) These statements are also used in menu-driven programming or event-based programming of PHP.

4) SYNTAX;

```
      switch ($var or 'expr')
  {
     case value1 :
            statements;
       :      break;
       :
     case value n :
       :      statements;
       :        break;
       :
     default :
             statements;
                break;
          // or ;  exit();
  }
```

5) The statements of the default part will execute if value doesn't matches any of the given case.

6) Here break is used to get exit from the loop infinity.

7) Eg;

```
          switch ($n)
          {
```

```
case 1:
    echo "MONDAY";
    break;
case 2:
    echo "SUNDAY";
    break;
default:
    echo "VALUES SHOULD BE
         BETWEEN 1TO 7";
}
```

Q.9) Explain the different operators in PHP.

Ans Operators in PHP are special symbols responsible for performing specific operation on 1 or more operands.

The following are the different types of operators in PHP.

1> TYPE-CASTING OPERATORS :

(integer) $var or intval ($var)
-> Converts the given variable to integer
(double) $var OR doubleval ($var)
-> Converts the given variable to double

2> ARITHMETIC OPERATORS :

(+) : Performs addition of 2 operands
(-) : Performs subtraction of 2 operands
(*) : Performs multiplication of 2 operands
(/) : Performs division of 2 operands
(%) : Finds remainder of 2 operands.

3> UNARY OPERATORS :
(Operates upon single operand)

Increment operator (++)
++$x -> Pre-increment
$x++ -> Post-increment
Decrement operator (--)
--$x -> Pre-decrement
$x-- -> Post decrement

4> ASSIGNMENT OPERATORS : (=)

Every arithmetic operator has a assignment operator (+=, -=, *=, /=, %=) which assigns the result of an arithmetic operation to a new variable and simultaneously update that variable.

5) COMPARISION OPERATORS :

These operators ~~[illegible]~~ performs simple comparision and evaluate the result in TRUE or FALSE

(<) → Less than
(>) → Greater than
(<=) → Less than or equal
(>=) → Greater than or equal
(==) → Equal to
(!=) → Not equal to
(<>) → Not equal to
(!==) → Not identical to
(===) → Identical to

Q.10) What are the methods and Arguments for accessing HTML information into PHP?

Ans 1) PHP is a server side scripting language whose encoding is not shown on the client side

2) PHP files are just embedded in the HTML code as

a value for action attribute of form tag
Eg; < form action=" file-php" >

3) Similarly PHP retrieves this information using method attribute of form tag.
Eg < form method = " " >

4) The value for this attribute are GET or POST.

5) These 2 are the PHP methods to store this HTML information in the PHP arguments

GET : Provides less security to data
POST : Provides more security to data

6) SYNTAX ;
In PHP

$var= $_GET ["var"]
↓ PHP Argument
↓ HTML variable of form output

$var = $_POST ["var"]
↓ PHP Argument
↓ HTML variable of form output

Q. Consider a table employee in database ym. Write a code to read employee details (name, id, salary) and display the employee having salary greater than 50,000.

Ans:

```
$db = ("localhost", "root", " ", " ym");
$sql = "Select * from emp";
$r = mysqli_query($db, $sql);

echo "<h2> FOLLOWING EMPYOEES HAVE
            SALARY GREATER THAN 50000
                 <h2>";
while ($row = mysqli_fetch_array
                  ($r, MYSQLI_BOTH));
 {
```

02/01 (10) [illegible]

```
    // echo $row['name'];
    // echo $row['salary'];

    if($row['salary'] >=50000)
    {
     echo $row['name'];
     echo $row['salary'];
    }
    else
    {
     echo " NO RECORDS FOUND ";
    }
   }
          ?>
```

Q. Consider a table student in database db with 'rno', 'name' and mark in 5 subjects.
Write a program to insert this details in the student table and then display the name and percentage of those students from the table.

CODE:

```
          <? PHP
$db = ("localhost", "root", " ", "db");
$query1 = " select * from students";

$rn = $_GET ['r'];
$n = $_GET ['t'];          } -> VALUES
$m1 = $_GET ['s1'];              FROM
                                 HTML
                                 FORM
```

```
$m2  = $_GET['S2'];  }
$m3  = $_GET['S3'];  }  ->  VALUES
$m4  = $_GET['S4'];  }      FROM
$m5  = $_GET['S5'];  }      HTML
                            FORM
$query2 = " insert into student
        values ('".rn."','".n."','".m1."','".m2."',
               '".m3."','".m4."','".m5."');
 \\execute ;
 mysqli_query($db,$query2);
$r = mysqli_query ($db, $query1);

while ($raw = mysqli_fetch_array
          ($~~result~~,    MYSQLIBOTH)
{
   echo $raw['name'];
$per =(($raw['sm1'] + $raw['sm2']
       +$raw['sm3'] + $raw['sm4']
      + $raw['sm5]) / 500) * 100;
   echo "PERCEANTAGE : ".$per;
}
            ?>
```

Q. Consider a table "users" having "username" and "password" in the database "yn". Write a program to create a login to check validity of that user and if user is valid then redirect it to the HTML page.

CODE:
```
$db=("localhost", "root", " ", "yn")
$ um = $_GET ["uname"] } -> VALUES FROM
$ pw = $_GET ["pword"] }    HTML FORM
$ sql= "Select * from users";
$ raw = mysqli_query ($db, $sql);

while ($arr = mysqli_fetch_array
                ($raw, MYSQLI_BOTH))
```

```
{
  if($un == $arr["username"] &&
    $pw == $arr["password"])
  {
    header('Location : Home.html');
  }
  else
  {
    print ("<h2> INVALID USER </h2>");
  }
};        ?>
```

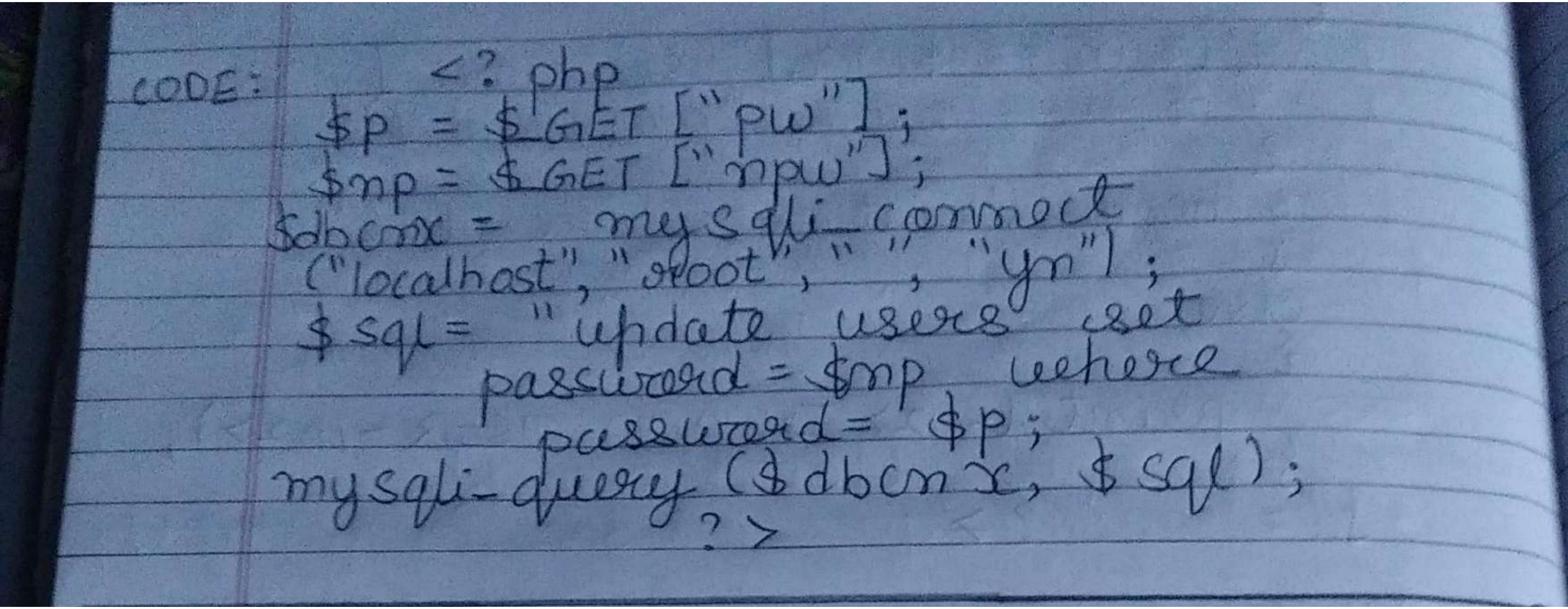
Q. Write a program to delete a user account from a users table by accepting username and password from the user.

PASSWORD ACCEPTED FROM HTML FORM

CODE

```
        <? php
  $p = $_GET ["pw"];
  $db = mysqli_connect
("localhost", "root", " ", "yn");
  $ sql = "delete from users where
              password = $p";
 mysqli_query ($dbcnx, $sql);
//header ('Location : ||) ;
              ?>
```

Q. Write a program to change password of a user by accepting initial password and new password.



```
CODE:   <? php
$p = $GET ["pw"];
$np = $GET ["npw"];
$dbcnx =   mysqli_connect
("localhost", "root", "", "yn");
$sql= "update users set
      password=$np  where
        password= $p;
mysqli_query ($dbcnx, $sql);
        ?>
```