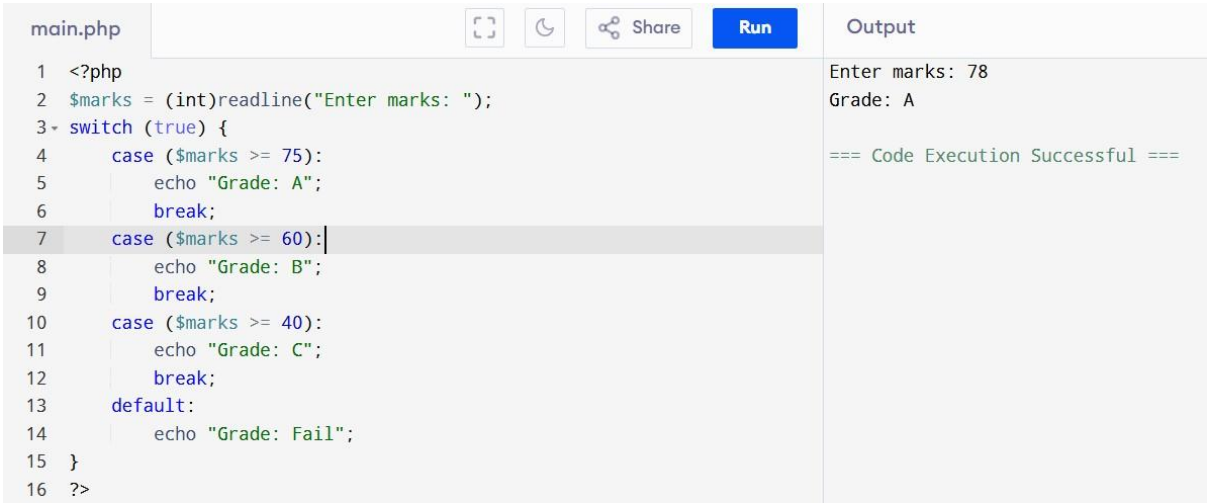
Q1. Create a PHP program using switch to print a grade for marks input (A, B, C, Fail).

Ans:

```
<?php
$marks = (int)readline("Enter marks: ");
switch (true) {
   case ($marks >= 75):
      echo "Grade: A";
      break;
   case ($marks >= 60):
      echo "Grade: B";
      break;
   case ($marks >= 40):
      echo "Grade: C";
      break;
   default:
      echo "Grade: Fail";
}
?>
```

Output:



2. Use while loop to sum the first 10 even numbers.

Ans:

```
<?php
$count = 0;
$num = 2;
```

```
$sum = 0;


while ($count < 10) {
   $sum += $num;
   $count++;
   $num += 2;
}


echo "Sum of first 10 even numbers: " . $sum;
?>
```

Output:

Q3. Write PHP Script to display sum of digit of entered number Ans:

```
<?php
$num = readline("Enter a number: ");
$sum = 0;


while ($num > 0) {
   $sum += $num % 10;
   $num = (int)($num / 10);
}
echo "Sum of digits: $sum";
?>
```

Output:

main.php

Share Run Output

```php
<?php
$num = readline("Enter a number: ");
$sum = 0;

while ($num > 0) {
    $sum += $num % 10;
    $num = (int)($num / 10);
}
echo "Sum of digits: $sum";
?>
```

```
Enter a number: 66
Sum of digits: 12

=== Code Execution Successful ===
```

4. Write a PHP program to display multiplication table of number 'n'.

Ans:

```php
<?php

$n = readline("Enter a number: ");


for ($i = 1; $i <= 10; $i++) {

echo "$n x $i = " . ($n * $i) . "\n";

}

?>
```

Output:

5. Write a PHP program to display multiplication table of 1 to 5]

Ans:

```php
<?php


for ($i= 1; $i<5; $i++) {
   echo "<br>MULTIPLICATION TABLE OF $i<br>";
   for($j=1;$j<=10;$j++){
      echo "$i * $j = ".($i*$j)."<br>";
```

```
    }
}


?>
```

Output:

main.php

```php
<?php
for ($n = 1; $n <= 5; $n++) {
    echo "Table of $n:\n";
    for ($i = 1; $i <= 10; $i++) {
        echo "$n x $i = " . ($n * $i) . "\n";
    }
    echo "\n";
}
?>
```

Output

```
Table of 1:
1 x 1 = 1
1 x 2 = 2
1 x 3 = 3
1 x 4 = 4
1 x 5 = 5
1 x 6 = 6
1 x 7 = 7
1 x 8 = 8
1 x 9 = 9
1 x 10 = 10

Table of 2:
2 x 1 = 2
2 x 2 = 4
2 x 3 = 6
2 x 4 = 8
2 x 5 = 10
2 x 6 = 12
2 x 7 = 14
2 x 8 = 16
2 x 9 = 18
2 x 10 = 20

Table of 3:
3 x 1 = 3
```

6. Write a for each loop to print values of an indexed and associative array.

Ans:

```
<?php
$indexed = array("Apple", "Banana", "Mango");
$associative = array("name" => "John", "age" => 20, "city" => "Mumbai");

echo "Indexed Array values:\n";
foreach ($indexed as $value) {
echo $value . "\n";
}

echo "\nAssosiative Array values:\n";
foreach ($associative as $key => $value) {
echo "$key: $value\n";
}
?>
```

Output:

main.php

```
<?php
$indexed = array("Apple", "Banana", "Mango");
$associative = array("name" => "John", "age" => 20, "city" => "Mumbai"
    );

echo "Indexed Array values:\n";
foreach ($indexed as $value) {
  echo $value . "\n";
}

echo "\nAssosiative Array values:\n";
foreach ($associative as $key => $value) {
  echo "$key: $value\n";
}
?>

```

Output

```
Indexed Array values:
Apple
Banana
Mango

Assosiative Array values:
name: John
age: 20
city: Mumbai


=== Code Execution Successful ===
```

Q

7. Create an associative array for student names and grades, display them using a loop.

Ans:

```
<?php
$students = array("Amit" => "A", "Neha" => "B", "Rahul" => "C");

foreach ($students as $name => $grade) {
echo "Name: $name, Grade: $grade\n";
}
?>
```

Output:

8. Write a program using multidimensional arrays to store student records (Name, Roll No, Marks) and print them.

Ans:

```
<?php
$students = array(
array("Name" => "Amit", "Roll" => 1, "Marks" => 85),
array("Name" => "Neha", "Roll" => 2, "Marks" => 78),
array("Name" => "Rahul", "Roll" => 3, "Marks" => 92)
);

foreach ($students as $student) {
 echo "Name: " . $student["Name"] . ", Roll No: " . $student["Roll"] . ", Marks: " . $student["Marks"] . "\n";
}
?>
```

Output:

main.php

Share Run

Output

```php
<?php
$students = array(
  array("Name" => "Amit", "Roll" => 1, "Marks" => 85),
  array("Name" => "Neha", "Roll" => 2, "Marks" => 78),
  array("Name" => "Rahul", "Roll" => 3, "Marks" => 92)
);

foreach ($students as $student) {
  echo "Name: " . $student["Name"] . ", Roll No: " . $student["Roll"]
      . ", Marks: " . $student["Marks"] . "\n";
}
?>
```

```
Name: Amit, Roll No: 1, Marks: 85
Name: Neha, Roll No: 2, Marks: 78
Name: Rahul, Roll No: 3, Marks: 92


=== Code Execution Successful ===
```

Q9. Write a program using multidimensional arrays to store user records (Name, email id, mobile no and address) and print them.

Ans:

```
<?php

$users = array(

 array("Name" => "Amit", "Email" => "amit@mail.com", "Mobile" => "1234567890", "Address" => "Mumbai"),

 array("Name" => "Neha", "Email" => "neha@mail.com", "Mobile" => "9876543210", "Address" => "Pune")

);


foreach ($users as $user) {   echo "Name: " . $user["Name"] . ", Email: " . $user["Email"] . ", Mobile: " . $user["Mobile"] . ", Address: " . $user["Address"] . "\n";

}

?>
```

Output:

main.php

Share Run

Output

```php
<?php
$users = array(
  array("Name" => "Amit", "Email" => "amit@mail.com", "Mobile" =>
      "1234567890", "Address" => "Mumbai"),
  array("Name" => "Neha", "Email" => "neha@mail.com", "Mobile" =>
      "9876543210", "Address" => "Pune")
);

foreach ($users as $user) {
  echo "Name: " . $user["Name"] . ", Email: " . $user["Email"] . ",
      Mobile: " . $user["Mobile"] . ", Address: " . $user["Address"] .
      "\n";
}
?>
```

```
Name: Amit, Email: amit@mail.com, Mobile: 1234567890, Address: Mumbai
Name: Neha, Email: neha@mail.com, Mobile: 9876543210, Address: Pune


=== Code Execution Successful ===
```

10. Write a PHP program to calculate the length of a string and to count words without using str_word_count().

Q

Ans:

```
<?php
$str="Aditya Makwana";
$l=strlen($str);

$c=str_word_count($str);
echo"String: $str<br>";
echo"Length of string: $l<br>";
echo "Word count: $c<br>";
?>
```

Output:

String: Aditya Makwana
Length of string: 14
Word count: 2

Q11. Write a parameterized function to calculate the sum of two numbers.

Ans:

```
<?php
function sum($a, $b) {
return $a + $b;
}

$num1 = readline("Enter first number: ");
$num2 = readline("Enter second number: ");
echo "Sum: " . sum($num1, $num2);
?>
```

Output:

main.php

```php
<?php
function sum($a, $b) {
    return $a + $b;
}

$num1 = readline("Enter first number: ");
$num2 = readline("Enter second number: ");
echo "Sum: " . sum($num1, $num2);
?>
```

Output

```
Enter first number: 10
Enter second number: 20
Sum: 30

=== Code Execution Successful ===
```

Q12. Implement an anonymous function to print a string.

Ans:

```php
<?php

$print = function($str) {

echo $str;

};


$input = readline("Enter a string: ");

$print($input);

?>
```

Output:

main.php

```php
<?php
$print = function($str) {
    echo $str;
};

$input = readline("Enter a string: ");
$print($input);
?>
```

Output

```
Enter a string: hellow
hellow

=== Code Execution Successful ===
```

Q13. Implement single inheritance in PHP (parent class: student and child class: Test1) and Display details of student with test result

Ans:

```php
<?php

class Student{

   public $name,$rollno;


   function setDetails(){

      $this->name= readline("Enter name:
```

```
        $this->rollno= readline("Enter roll no: ");
    }
    function disp(){
        echo "Name: $this->name, Roll No: $this->rollno";
    }
}

class Test1 extends Student{
    public $marks;
    function setDetails(){
       parent::setDetails();
        $this->marks= readline("Enter marks: ");
    }
    function disp(){
        parent::disp();
        echo " Marks: $this->marks\n";
    }
}

$t1=new Test1();
$t1->setDetails();
$t1->disp();
?>
```

Output:

```
# php hello.php
Enter name: Aditya
Enter roll no: 42
Enter marks: 20
Name: Aditya, Roll No: 42 Marks: 20
```

Q14. Implement multilevel inheritance in PHP (parent class: student → child class: Tests→result) and Display result of student for two class test percentage and average of test1 and test2 subject wise.

Ans:

```
<?php
// Level 1 - Parent class
class Student {
    public $name;
    public $roll;

    function setStudent() {
        $this->name = readline("Enter Student Name: ");
        $this->roll = readline("Enter Roll No: ");
    }

    function getStudent() {
        echo "Name: $this->name\n";
        echo "Roll No: $this->roll\n";
    }
}

// Level 2 - Child class of Student
class Test extends Student {
    public $test1;
    public $test2;
    public function setTests() {
        $this->test1 = readline("Enter Test 1 Marks: ");
        $this->test2 = readline("Enter Test 2 Marks: ");
    }

    public function showTests() {
        echo "Test 1 Marks: $this->test1\n";
        echo "Test 2 Marks: $this->test2\n";
    }
}

// Level 3 - Child class of Test
```

```
class Result extends Test {
    public function showResult() {
        $avg = ($this->test1 + $this->test2) / 2;
        $per =( ($this->test1 + $this->test2) / 200) * 100;
        echo "Average Marks: $avg\n";
        echo "Percentage: $per%\n";
    }
}

// Create object of Result class
$obj = new Result();
$obj->setStudent();
$obj->setTests();
$obj->getStudent();
$obj->showTests();
$obj->showResult();
?>
```

Output:

```
# php hello.php
Enter Student Name: ADITYA
Enter Roll No: 42
Enter Test 1 Marks: 100
Enter Test 2 Marks: 100
Name: ADITYA
Roll No: 42
Test 1 Marks: 100
Test 2 Marks: 100
Average Marks: 100
Percentage: 100%
```

Q15. Write a PHP script to demonstrate any five-class introspection using Ans:

```
<?php class
Demo {
```

```
    public $x;
function test() {}
}
$obj = new Demo();

echo get_class($obj) . "<br>";
echo method_exists($obj, "test") . "<br>";
echo property_exists($obj, "x") . "<br>";
print_r(get_class_methods($obj));
print_r(get_object_vars($obj));
?>
```

Output:

Q16. Create a class product with data members id, name and price. Consider that customer bought multiple products. Then Calculate the total price of products purchased. Use constructor

Ans:

```
    <?php
class Product {
    public $id,$name,$price;

    // Constructor to initialize product
    function __construct($id, $name, $price) {
        $this->id = $id;
        $this->name = $name;
        $this->price = $price;
```

```
    }

    // Function to return price
    function getPrice() {
        return $this->price;
    }

    // Function to show product info (optional)
    function show() {
        echo "ID: $this->id, Name: $this->name, Price: $this->price<br>";
    }
}

// Create multiple product objects
$p1 = new Product(1, "Pen", 10);
$p2 = new Product(2, "Book", 50);
$p3 = new Product(3, "Bag", 300);

// Display products (optional)
echo"<br>Product List:<br>";
$p1->show();
$p2->show();
$p3->show();

echo "<br>Product Purchased:<br>";
$p1->show();
$p3->show();
// Calculate total price
$total = $p1->getPrice() + $p3->getPrice();
echo "<br>Total Price of Products Purchased: ₹$total";
?>
```

Output:

Product List:
ID: 1, Name: Pen, Price: 10
ID: 2, Name: Book, Price: 50
ID: 3, Name: Bag, Price: 300

Product Purchased:
ID: 1, Name: Pen, Price: 10
ID: 3, Name: Bag, Price: 300

Total Price of Products Purchased: ₹310

Q17. Write a program to serialize and unserialize an array/object.

Ans:

```
<?php
$arr = ["name" => "John", "age" => 25];
$serialized = serialize($arr);
echo "Serialized: $serialized\n";

$unserialized = unserialize($serialized);
echo "Unserialized:\n"; print_r($unserialized);
?>
```

Output:

Q18. Design a form for user registration and validate fields like email, password, and phone using PHP.

Ans:

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $email = $_POST["email"];
    $password = $_POST["password"];
    $phone = $_POST["phone"];
```

```
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
echo "Invalid Email<br>";     } elseif
(strlen($password) < 6) {
        echo "Password must be at least 6 characters<br>";
} elseif (!preg_match('/^[0-9]{10}$/', $phone)) {         echo
"Invalid Phone Number<br>";
    } else {
        echo "All inputs are valid!";
    }
}
?>

<form method="post">
    Email: <input type="text" name="email"><br>
    Password: <input type="password" name="password"><br>
    Phone: <input type="text" name="phone"><br>
    <input type="submit">
</form>
```

18. Registration form with validation

php Copy Edit

```
<!-- form.html -->
<form method="post" action="validate.php">
    Email: <input type="text" name="email"><br>
    Password: <input type="password" name="pass"><br>
    Phone: <input type="text" name="phone"><br>
    <input type="submit">
</form>
```

php Copy Edit

```
<!-- validate.php -->
<?php
$email = $_POST['email'];
$pass = $_POST['pass'];
$phone = $_POST['phone'];

if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
    echo "Invalid email<br>";
}
if (strlen($pass) < 6) {
    echo "Password too short<br>";
}
if (!preg_match("/^[0-9]{10}$/", $phone)) {
    echo "Invalid phone<br>";
}
?>
```

Q19. Design a form for user registration and retrieve information on successful submission using PHP.

Ans:

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
   $email = $_POST["email"];
   $password = $_POST["password"];
   $phone = $_POST["phone"];

   if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
      echo "Invalid Email<br>";
   } elseif (strlen($password) < 6) {
      echo "Password must be at least 6 characters<br>";
```

```
    } elseif (!preg_match('/^[0-9]{10}$/', $phone)) {
        echo "Invalid Phone Number<br>";
    } else {
        echo "All inputs are valid, FORM SUBMISTTED SUCCESSFULLY<br>";
    }
}
?>

<form method="post">
    Email: <input type="text" name="email"><br>
    Password: <input type="password" name="password"><br>
    Phone: <input type="text" name="phone"><br>
    <input type="submit">
</form>
```

Q20. Develop a PHP application to enter student data into a MySQL database. Retrieve and display data from the database in tabular form. Update a record in the database. Delete a specific record from the database.

Ans:

```
<?php
// Database connection
$servername = "localhost";
$username = "root";
$password = "";
$database = "student";

$conn = mysqli_connect($servername, $username, $password, $database);

if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// Insert Data
if (isset($_POST['insert'])) {
```

```
    $name = $_POST['name'];
    $email = $_POST['email'];

    $sql = "INSERT INTO user (name, email) VALUES ('$name', '$email')";
    if (mysqli_query($conn, $sql)) {
        echo "Record inserted successfully!";
    } else {
        echo "Error: " . mysqli_error($conn);
    }
}

// Update Data
if (isset($_POST['update'])) {
    $id = $_POST['id'];
    $name = $_POST['name'];

    $sql = "UPDATE user SET name='$name' WHERE id=$id";
    if (mysqli_query($conn, $sql)) {
        echo "Record updated successfully!";
    } else {
        echo "Error: " . mysqli_error($conn);
    }
}

// Delete Data
if (isset($_POST['delete'])) {
    $id = $_POST['id'];

    $sql = "DELETE FROM user WHERE id=$id";
    if (mysqli_query($conn, $sql)) {
        echo "Record deleted successfully!";
    } else {
        echo "Error: " . mysqli_error($conn);
    }
```

```
}

// Fetch Data
$sql = "SELECT * FROM user";
$result = mysqli_query($conn, $sql);
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PHP MySQL Operations</title>
</head>
<body>
    <h2>Insert Data</h2>
    <form method="POST">
        Name: <input type="text" name="name" required>
        Email: <input type="email" name="email" required>
        <button type="submit" name="insert">Insert</button>
    </form>

    <h2>Update Data</h2>
    <form method="POST">
        ID: <input type="number" name="id" required>
        New Name: <input type="text" name="name" required>
        <button type="submit" name="update">Update</button>
    </form>

    <h2>Delete Data</h2>
    <form method="POST">
        ID: <input type="number" name="id" required>
        <button type="submit" name="delete">Delete</button>
    </form>
```

```
    <h2>Users List</h2>
    <table border="1">
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Email</th>
        </tr>
        <?php while ($row = mysqli_fetch_assoc($result)) { ?>
        <tr>
            <td><?php echo $row['id']; ?></td>
            <td><?php echo $row['name']; ?></td>
            <td><?php echo $row['email']; ?></td>
        </tr>
        <?php } ?>
    </table>
</body>
</html>


<?php mysqli_close($conn); ?>
```

Output:-

Insert Data

Name: Email: Insert

Update Data

ID: New Name: Update

Delete Data

ID: Delete

Users List

ID	Name	Email
3	shilpa makwana	shilpamakwana707@gmail.com

21. Develop a PHP application to enter student data into a MySQL database. Retrieve and display data from the database in tabular form. Delete a specific record from the database ?

ANS:

Code:-

```
<?php
$conn = mysqli_connect("localhost", "root", "", "student");

if (!$conn) {
   die("Connection failed: " . mysqli_connect_error());
}

// Insert Data
if (isset($_POST['insert'])) {
   $name = $_POST['name'];
   $email = $_POST['email'];

   $sql = "INSERT INTO user (name, email) VALUES ('$name', '$email')";
   if (mysqli_query($conn, $sql)) {
      echo "Record inserted successfully!";
   } else {
      echo "Error: " . mysqli_error($conn);
   }
}

if (isset($_POST['update'])) {
   $id = $_POST['id'];
   $name = $_POST['name'];
   $email = $_POST['email'];

   $sql = "UPDATE user SET name='$name', email='$email' WHERE id=$id";

   if (mysqli_query($conn, $sql)) {
      echo "Record updated successfully!";
   }
```

```
}
if (isset($_POST['delete'])) {
    $id = $_POST['id'];

    $sql = "DELETE FROM user WHERE id=$id";
    if (mysqli_query($conn, $sql)) {
        echo "Record deleted successfully!";
    }
}
$sql = "SELECT * FROM user";
$result = mysqli_query($conn, $sql);
?>
<html lang="en">

<body>
    <form method="POST">
        <H1>INSERT RECORD </H1>
        Name: <input type="text" name="name" required>
        Email: <input type="email" name="email" required>
        <button type="submit" name="insert">Insert</button>
    </form>
    <form method="POST">
        <H1>UPDATE RECORD </H1>
        ID: <input type="number" name="id"><br>
        Name: <input type="text" name="name"><br>
        Email: <input type="text" name="email"><br>
        <input type="submit" name="update" value="Update"><br><br>
    </form>
    <form method="POST">
        <H1>DELETE RECORD </H1>
        ID: <input type="number" name="id"><br>
        <input type="submit" name="delete" value="Delete"><br><br>
    </form>
```

```
    <H1>RECORDS</H1>
    <table border="1">
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Email</th>
        </tr>
        <?php while ($row = mysqli_fetch_assoc($result)) { ?>
            <tr>
                <td><?php echo $row['id']; ?></td>
                <td><?php echo $row['name']; ?></td>
                <td><?php echo $row['email']; ?></td>
            </tr>
        <?php } ?>
    </table>
</body>


</html>


<?php mysqli_close($conn); ?>
```

INSERT RECORD

Name: Email: Insert

UPDATE RECORD

ID:
Name:
Email:
Update

DELETE RECORD

ID:
Delete

RECORDS

ID	Name	Email
15	Aditya	adityamakwana707@gmail.com