Q1. What is the correct JavaScript syntax to write "Hello World"?

1. System.out.println("Hello World")
2. println ("Hello World")
3. **document.write("Hello World")**
4. response.write("Hello World")

Q2. What will be the output of the following JavaScript code?

function f1() {

var x = '10+10';

alert(eval(x));

}

1. 10
2. **20**
3. Error
4. undefined

Q3. What will be the output of the following JavaScript code?

<script type="text/javascript">

var expression = eval(new String('2 + 2'));

alert(eval(expression));

</script>

1. 4
2. Null
3. undefined
4. **2 + 2**

Q4. What will be the output of the following JavaScript code?

<script type="text/javascript">

var expression = eval(new String('2 + 2'));

alert(eval(expression.toString()));

</script>

1. **4**
2. Null
3. undefined
4. 2 + 2

Q5. What will be the output of the following JavaScript code?

<script type="text/javascript">

var x = 2, y = 4;

alert(eval('x + y'));

</script>

1. **6**
2. x + y
3. Error
4. undefined

Q6. What will be the result of console.log('1' + '1');

1. 1
2. 2
3. **11**
4. Null

Q7. What will be the result of console.log(eval ('1') + eval('1'));

1. 1
2. **2**
3. 11
4. eval ('1') + eval('1')

Q8. What will be the result of console.log(1 + "a1");

1. 1 a1
2. **1a1**
3. 1a2
4. 1a 1

Q9. What will be the result of console.log(1 + "a" + "1");

1. 1A1
2. 1 a 1
3. **1a1**
4. None of the above

Q10. What will be the result of console.log(10 + 10 + "IWAY");

1. 1010IWAY
2. 10 10 IWAY
3. **20IWAY**
4. Error

Q11. What will be the result of console.log(true + true);

1. **2**
2. true + true
3. Error
4. undefined

Q12. What will be the result of console.log(true + false + 5);

1. true + false + 5
2. 1 + 0 + 5
3. **6**
4. Null

Q13. What will be the result of console.log(false - false - 10);

1. 10
2. **-10**
3. false – false – 10
4. Error.

Q14. What will be the result of console.log(false + false - 10);

1. 10
2. **-10**
3. false + false – 10
4. Error.

Q15. What of the following statement will get the value of letters object?

var letters = {a: 'apple', b: 'banana'};

1. console.log(letters.a);
2. console.log(letters[a]);
3. console.log(letters["a"]);
4. **Both A and C**

Q16. Which of the following option you will use to get the values of letters Object.

var letters = { a: 'apple', b: 'banana', c: 'Coconut', d: 'Dates' };

1. **for (var i in letters) {**

**console.log(letters[i]);**

**}**

1. for (var i in letters) {

console.log(letters(i));

}

1. for (var i in letters) {

console.log(letters.i);

}

1. None of the above.

Q17. Which of the following statement you will issue to create a Person Object having details of two employees.

1. var Person = { 'id': '1001', 'ename': 'SMITH', 'salary': 25000 }, { 'id': '1002', 'ename': 'JONES', 'salary': 35000 };
2. var Person = "{ 'id': '1001', 'ename': 'SMITH', 'salary': 25000 }, { 'id': '1002', 'ename': 'JONES', 'salary': 35000 }";
3. **var Person = [{ 'id': '1001', 'ename': 'SMITH', 'salary': 25000 }, { 'id': '1002', 'ename': 'JONES', 'salary': 35000 }];**
4. All of the above.

Q18. Examine the following JavaScript code.

<script type="text/javascript">

var Person = { "age": 28, "name": "Scott", "designation": "developer" };

console.log(Person.name);

delete Person.name;

console.log(Person.name);

</script>

What will the output?

1. Scott and Scott
2. **Scott and undefined**
3. Scott and NULL
4. Scott and Error

Q19. Examine the following JavaScript code.

<script type="text/javascript">

var Person = { "age": 28, "name": "Scott", "designation": "developer" };

console.log(Person.name);

delete Person.ename;

console.log(Person.name);

</script>

What will the output?

1. **Scott and Scott**
2. Scott and undefined
3. Scott and NULL
4. Scott and Error

Q20. The syntax of creating array using array literal is?

1. **var arrayname=[value1,value2.....valueN];**
2. var arrayname=(value1,value2.....valueN);
3. var arrayname=new value1,value2.....valueN;
4. None of the above

Q21. Examine the following JavaScript code?

<script type="text/javascript" language="javascript">

var qpt = "Qualiyt Point Technologies";

var result = qpt.split(" ");

document.write(result);

</script>

1. Quality
2. Q,u,a,l,i,t,y,P,o,i,n,t,T,e,c,h,n,o,l,o,g,i,e,s
3. **Qualiyt,Point,Technologies**
4. QualityPointTechnologies

Q22. Examine the following set of statements

var x = ('a', 'b');

console.log(x);

What will be the output?

1. a
2. **b**
3. a b
4. undefined

Q23. Examine the following set of statements

var x = true

if (x = 1) {

console.log("true");

} else {

console.log('false');

}

What will be the output?

1. 0
2. 1
3. **true**
4. false

Q24. Examine the following set of statements

var x = true

if (x == 1) {

console.log("true");

} else {

console.log('false');

}

What will be the output?

1. 0
2. 1
3. **true**
4. false

Q25. Examine the following set of statements

var x = true

if (x === 1) {

console.log("true");

} else {

console.log('false');

}

What will be the output?

1. 0
2. 1
3. true
4. **false**

Q26. Examine the following set of statements

var x = true

if (x = 0) {

console.log("true");

} else {

console.log('false');

}

What will be the output?

1. 0
2. 1
3. true
4. **false**

Q27. Examine the following set of statements

var x = false

if (x == 0) {

console.log("true");

} else {

console.log('false');

}

What will be the output?

1. 0
2. 1
3. **true**
4. false

Q28. Consider the following array. Select the code that prints the contents of the array.

var years = [1950, 1960, 1970, 1980, 1990, 2000, 2010];

1. years.forEach(function (v, i, ar) {

console.log(v);

});

1. years.forEach(function (v, i, ar) {

console.log(years[i]);

});

1. years.forEach(function (v, i, ar) {

console.log(v[i]);

});

1. **Both A and B**

Q29. Consider the following array. Select the code that prints the contents of the array.

var years = [1950, 1960, 1970, 1980, 1990, 2000, 2010];

1. **for (var i in years) {**

**console.log(years[i]);**

**}**

1. for (var i in years) {

console.log(years.i);

}

1. for (var i in years) {

console.log(i[years]);

}

1. None of the above.

Q30. Consider the following array. Select the code that prints the contents of the array.

var years = [1950, 1960, 1970, 1980, 1990, 2000, 2010];

1. **for (var i = 0; i < years.length; i++) {**

**console.log(years[i]);**

**}**

1. for (var i = 0; i < years.length - 1; i++) {

console.log(years[i]);

}

1. Both A and B
2. None of the above.

Q31. Consider the following array. Select the code that prints the contents of the array in reverse order.

var years = [1950, 1960, 1970, 1980, 1990, 2000, 2010];

1. for (var i = years.length - 1; i >= 0; i--) {
   * 1. console.log(years[i]);
   1. }
2. for (var i = years.length - 1; i >= 0; --i) {
   1. console.log(years[i]);
   2. }
3. **Both A and B**
4. None of the above.

Q32. Consider the following JavaScript code.

var data = ["A", "B", "C", "D"];

data.unshift("X");

data.push("Y");

What does data look like?

1. ["A", "B", "C", "X", "D", "Y"]
2. ["X", "Y", "A", "B", "C", "D"]
3. **["X", "A", "B", "C", "D", "Y"]**
4. ["Y", "A", "B", "C", "D", "X"]

Q33. Consider the following JavaScript code.

var index = [12, 5, 8, 130, 44].lastIndexOf(8);

document.write("index is: " + index);

What will be the last index number of the given value?

1. 0
2. 1
3. **2**
4. 3

Q34. Consider the following JavaScript code.

var index = [5, 12, 5, 8, 130, 44, 5, 16, 5, 18].lastIndexOf(5);

document.write("<br />index is: " + index);

What will be the last index number of the given value?

1. 0
2. 2
3. 6
4. **8**

Q35. Consider the following JavaScript code.

var index = [12, 5, 8, 130, 44].pop();

document.write("index is: " + index);

What value will be present in index?

1. 12
2. 5
3. **44**
4. 130

Q36. Consider the following JavaScript code.

var index = [12, 5, 8, 130, 44].reverse();

console.log(index);

What will be the output?

1. [12, 5, 8, 130, 44]
2. **[44, 130, 8, 5, 12]**
3. [12, 44, 5, 130, 8]
4. None of the above.

Q37. Consider the following JavaScript code.

var index = [12, 5, 8, 130, 44].shift()

console.log("[" + index +"]");

What is the output?

1. **[12]**
2. [5, 8, 130, 44]
3. [130, 44]
4. [44]

Q38. Consider the following JavaScript code.

var index = [12, 5, 8, 130, 44].slice(0,2)

console.log("[" + index + "]");

What is the output?

1. **[12, 5]**
2. [5, 8]
3. [12, 5, 8]
4. [12, 5, 8, 130]

Q39. Consider the following JavaScript code.

var index = ['d', 'a', 'c', 'b'].sort();

console.log("[" + index + "]");

What is the output?

1. **[a,b,c,d]**
2. [c,d,a,b]
3. [d,a,c,b]
4. Invalid syntax for sort method.

Q40. Consider the following, what will be the output of the PHP code?

<?php

$x = "Infoway Technologies, PUNE";

$y = (explode(" " , $x));

print $y[0];

?>

1. I
2. **Infoway**
3. Infoway Technologies, PUNE
4. None of the above.

Q41. Consider the following PHP code.

<?php

$total = "25 students";

$more = 10;

$total = $total + $more;

echo "$total";

?>

What is the output?

1. **35**
2. 25 students
3. 25 students 10
4. 35 students

Q42. Which statement will output $x on the screen?

1. **echo “\$x”;**
2. echo “$$x”;
3. echo “/$x”;
4. echo “$x;”;

Q43. $("div#id1 .cl1"). What does it select?

1. The first element with id="cl1" inside any div element with class="id1"
2. **All elements with class="cl1" inside the first div element with id="id1"**
3. All div elements with id="id1" or class="cl1"
4. None of the above.

Q44. You want to use JQuery to read the value of a textbox. This value contains the users full name which is needed for further processing in Javascript.

Which code segment is NOT a workable solution?

1. **value = $("#txt\_name").text();**
2. value = $("#txt\_name").val();
3. value = $("#txt\_name").attr('value');
4. document.getElementById('txt\_name').value

Q45. Examine the following JavaScript code?

<script type="text/javascript">

var object1 = {same: ‘same‘};

var object2 = {};

object2 = object1;

object1.same = 'not same';

console.log(object2.same);

</script>

What will the output?

1. same
2. **not same**
3. undefined
4. NULL

Q46. Examine the following JavaScript code?

<script type="text/javascript">

var object1 = {same: 'same'};

var object2 = {};

object2 = object1;

object1.same = 'not same';

delete object1.same;

console.log(object2.same);

</script>

What will the output?

1. same
2. not same
3. **undefined**
4. NULL

Q47. Examine the following JavaScript code.

var price1 = 10;

var price2 = 10;

console.log(price1 === price2);

What will be the output?

1. **true**
2. false
3. Invalid comparison operator
4. None of the above.

Q48. Examine the following JavaScript code.

var price1 = 10;

var price2 = new Number(10);

console.log(price1 === price2);

What will be the output?

1. true
2. **false**
3. Invalid comparison operator
4. None of the above.

Q49. Examine the following JavaScript code.

var price1 = new Number(10);

var price2;

price2 = price1;

console.log(price1 === price2);

What will be the output?

1. **true**
2. false
3. Invalid comparison operator
4. None of the above.

Q50. Examine the following code.

var myString = 'Rahul';

var myStringCopy = myString;

var myString = null;

console.log(myString, myStringCopy);

What will be the output?

1. **null "Rahul"**
2. "Rahul" "Rahul"
3. "Rahul" null
4. null null

Q51. Examine the following code.

var myString = 'Rahul';

var myStringCopy = new String();

var myStringCopy = myString;

myString = null;

console.log(myString, myStringCopy);

What will be the output?

1. **null "Rahul"**
2. "Rahul" "Rahul"
3. "Rahul" null
4. null null

Q52. What will the output of the following statements?

var person = null;

console.log(typeof (person));

1. null
2. undefined
3. **object**
4. None of the above.

Q53. Examine the following AngularJS code.

<body style="background-color:{{color}}">

<div id="div1">

<input type="text" ng-model="color" />

</div>

</body>

What will happen, if user type color name in the textbox?

1. The programme will raise an error.
2. **The background-color of the body will be changed according the color given in the textbox**
3. Error, Invalid background-color style property.
4. None of the above.

Q54. Examine the following AngularJS code.

<div ng-app="" ng-init="quantity=7; cost=1200">

</div>

Which of the following statement is used to calculate the total (quantity \* cost).

1. <p>Total in dollar: <span ng-bind=quantity \* cost></span></p>
2. **<p>Total in dollar: <span ng-bind="quantity \* cost"></span></p>**
3. <p>Total in dollar: <span ng\_bind=quantity \* cost></span></p>
4. <p>Total in dollar: <span ngbind=quantity \* cost></span></p>

Q55. Examine the following AngularJS code.

<div ng-app="" ng-init="quantity=7; cost=1200">

</div>

Which of the following statement is used to calculate the total (quantity \* cost).

1. <p>Total in dollar: {quantity \* cost} </p>
2. **<p>Total in dollar: {{quantity \* cost}} </p>**
3. <p>Total in dollar: {{ "quantity \* cost" }} </p>
4. <p>Total in dollar: {{eval('quantity \* cost')}} </p>

Q56. Examine the following AngularJS code.

<div ng-app="" ng-init="firstName='John'; lastName='Doe'">

</div>

Which of the following statement is used to display the firstName and lastName?

1. **<p>The name is {{firstName + " " + lastName}} </p>**
2. <p>The name is {{“firstName + ““+ lastName" }} </p>
3. <p>The name is {{‘firstName || " " || lastName’}} </p>
4. <p>The name is {{$firstName + " " + $lastName}} </p>

Q57. Examine the following AngularJS code.

<div ng-app="" ng-init="firstName='John'; lastName='Doe'">

</div>

Which of the following statement is used to display the firstName and lastName?

1. **<p>The name is <span ng-bind="firstName + ' ' + lastName"> </span> </p>**
2. <p>The name is <span ng-bind= {"firstName + ' ' + lastName"}> </span> </p>
3. <p>The name is <span ng-bind="{firstName + ' ' + lastName}"> </span> </p>
4. <p>The name is <span ng-bind="{{firstName + ' ' + lastName}}"> </span> </p>

Q58. Examine the following AngularJS code.

<div ng-app="" ng-init="person={firstName:'John', lastName:'Doe'}">

</div>

Which of the following statement is used to display the firstName and lastName?

1. <p>The name is <span ng-bind="person.lastName"></span></p>
2. <p>The name is <span ng-bind="person['lastName']"></span></p>
3. <p>The name is <span ng-bind="person('lastName')"></span></p>
4. **Either A or B**

Q59. Examine the following AngularJS code.

<body ng-init="quantity=7; cost=56">

<div id="div1">

<div ng-app="" ng-init="fruits=['Apple','Grapes','Orange','Banana','Cherry']">

<select>

<option> \*\*\* Select fruits \*\*\* </option>

</select>

</div>

</div>

</body>

What of the following option you will use to get all Fruit name in the dropdown control?

1. **<option ng-repeat="x in fruits"> {{x}} </option>**
2. <option ng-repeat="fruits in x" ng-bind="x"></option>
3. <option ng-repeat="fruits in x" ng-bind="fruits"></option>
4. None of the above.

Q60. Examine the following AngularJS code.

<body ng-init="quantity=7; cost=56">

<div id="div1">

<div ng-app="" ng-init="fruits=['Apple','Grapes','Orange','Banana','Cherry']">

<select>

<option> \*\*\* Select fruits \*\*\* </option>

</select>

</div>

</div>

</body>

What of the following option you will use to get all Fruit name in the dropdown control?

1. **<option ng-repeat="x in fruits" ng-bind="x"></option>**
2. <option ng-repeat="fruits in x" ng-bind="x"></option>
3. <option ng-repeat="fruits in x" ng-bind="fruits"></option>
4. None of the above.

Q61. Examine the following AngularJS code.

<body ng-init="quantity=3; cost=5">

<div id="div1" ng-init="fruits=['Apple', 'Grapes', 'Orange', 'Banana', 'Cherry', 'Lemon', 'Mango']">

<input type="range" min="0" max="6" style="width: 100px;" ng-model="x" /> <span> {{x}} </span>

</div>

</body>

Which of the following option you will use to get values from the array?

1. **<p>The third result is <span ng-bind="fruits[x]"></span></p>**
2. <p>The third result is <span ng-bind="fruits[value]"></span></p>
3. <p>The third result is <span ng-bind="fruits[min]"></span></p>
4. <p>The third result is <span ng-bind="fruits[max]"></span></p>

Q62. What will be the output of the following code?

<?php

function track() {

static $count = 0;

$count++;

echo $count;

}

track();

track();

track();

?>

1. **123**
2. 111
3. 000
4. 011

Q63. What will be the output of the following PHP code?

<?php

function calc($price, $tax="")

{

$total = $price + ($price \* $tax);

echo "$total";

}

calc(42);

?>

1. Error
2. 0
3. **42**
4. 84

Q64. Examine the following PHP code.

<?php

$x = '1A';

$y = 2;

echo ($x + $y);

?>

What will be the output?

1. **3**
2. 1A2
3. 1
4. Error.

Q65. Examine the following PHP code.

<?php

$x = "Hello world!";

$x = null + 10;

echo($x);

?>

What will be the output?

1. null
2. undefined
3. Hello world!
4. **10**

Q66. What of the following option you will select to count the array elements.

1. **$c = count($a);**
2. $c = $a.length;
3. $c = $a.length();
4. Either B or C

Q67. How can we make attributes have multiple values?

1. <myElement myAttribute="value1 value2"/>
2. <myElement myAttribute="value1" myAttribute="value2"/>
3. <myElement myAttribute="value1, value2"/>
4. **attributes cannot have multiple values**

Q68. To create a constant in PHP,

1. **use the define() function.**
2. use the const() function.
3. use the constant() function.
4. use the $const() function.

Q69. How will you create constant in PHP

1. define("&GREETING", "Welcome to Infoway, PUNE");
2. **define("GREETING", "Welcome to Infoway, PUNE");**
3. define($"GREETING", "Welcome to Infoway, PUNE");
4. $define("GREETING", "Welcome to Infoway, PUNE");

Q70. What is the output of the following PHP code?

<?php

$txt1="Infoway";

$txt2=", PUNE";

$txt1 .= $txt2;

echo $txt1;

?>

1. Infoway
2. , Pune
3. **Infoway, Pune**
4. Error.

Q71. What will be the output?

<?php

$x = "a";

$y = "a";

echo $x === $y;

?>

1. 0
2. **1**
3. null
4. undefined

Q72. What will be the output?

<?php

$x = array("a" => "red");

$y = array("a" => "red");

echo($x == $y);

?>

1. 0
2. **1**
3. null
4. undefined

Q73. Examine the following PHP code.

<?php

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

foreach ($age as $x => $y) {

echo $x . " ,";

}

?>

What will be the output?

1. 35, 37, 43
2. **Peter, Ben, Joe**
3. Peter 35, Ben 37, Joe 43
4. None of the above.

Q74. Which of the following option will create a two-dimensional array?

1. **$cars = array( array ("Volvo", 22, 18), array ("BMW", 15, 13));**
2. $cars = array(array ["Volvo", 22, 18], array ["BMW", 15, 13]);
3. $cars = array[array ("Volvo", 22, 18), array ("BMW", 15, 13)];
4. $cars = array(array {"Volvo", 22, 18}, array {"BMW", 15, 13});

Q75. . Examine the following code.

<?php

function f1()

{

function f2()

{

echo 'Infoway Technologies';

}

echo ' is in PUNE';

}

f1();

f1();

?>

What will be the output?

1. Infoway Technologies
2. Infoway Technologies is in PUNE
3. Error
4. **is in PUNE Error**

Q76. Examine the following code.

<?php

function f1()

{

function f2()

{

echo 'Infoway Technologies';

}

echo ' is in PUNE';

}

f2();

f1();

?>

What will be the output?

1. Infoway Technologies
2. Infoway Technologies is in PUNE
3. **Error**
4. is in PUNE Error

Q77. Examine the following PHP code.

<?php

$op2 = " World";

function f1 ($op1)

{

echo $op1;

echo $op2;

}

f1 ("Hello");

?>

What will be the output?

1. Hello World
2. **Hello Error**
3. World
4. None of the above.

Q78. Examine the following PHP code.

<?php

$op2 = " World";

function f1 ($op1)

{

global $op2;

echo $op1;

echo $op2;

}

f1 ("Hello");

?>

What will be the output?

1. **Hello World**
2. Hello Error
3. World
4. None of the above.

Q79. What will be the output of the following PHP code?

<?php

$number = array ("10", 2);

echo (array\_sum ($number));

?>

1. 2
2. 10
3. **12**
4. Error

Q80. What will be the output of the following PHP code?

<?php

$number = array ('Hello', 10, 2);

echo (array\_sum ($number));

?>

1. 2
2. 10
3. **12**
4. Error

Q81. What will be the output of the following PHP code?

<?php

$z = 3;

function f1($x, $y) {

return ($x + $y + $z);

}

echo (f1(1,2));

?>

1. 3
2. 6
3. Error: Undefined variable: z
4. **Error: Undefined variable: z and then 3**

Q82. What will be the output of the following PHP code?

<?php

$z = 3;

function f1($x, $y) {

global $z;

return ($x + $y + $z);

}

echo (f1(1,2));

?>

1. 3
2. **6**
3. Error: Undefined variable: z
4. Error: Undefined variable: z and then 3

Q83. . Consider the following code.

<!DOCTYPE html>

<html>

<head>

<title>Infoway Technology, PUNE</title>

<script type="text/javascript" id="script1">

window.onload = function () {

var myHeading = document.querySelector('h2');

myHeading.textContent = 'Hello world!';

}

</script>

</head>

<body>

<div class="container-fluid">

<h2>Header.2 Line1 </h2>

<h2>Header.2 Line2 </h2>

</div>

</body>

</html>

**What will be the output?**

1. Hello world! will be displayed in both <h2> tag.
2. **Hello world! will be displayed in first <h2> tag.**
3. Hello world! will be displayed in second <h2> tag.
4. None of the above.

Q84. What will be the output of the following JavaScript code?

<script type="text/javascript" id="script1">

window.onload = function () {

console.log(x);

var x;

x = 1001;

</script>

1. 0
2. 1001
3. null
4. **undefined**

Q85. var myh2 = document.querySelector ('h2');

myh2.onclick = function () {

console.log ("1");

}

The above code snippet is equivalent to

1. document.querySelector('h2').onclick = function () {

console.log ("Infoway, PUNE");

};

1. **document.querySelector("h2","onclick", function () {**

**console.log ("Infoway, PUNE ");**

**});**

1. document.getElementsByTagName("h2").onclick(function () {

console.log ("Infoway, PUNE ");

});

1. None of the above

Q86. Consider the following array in JavaScript code.

var arr = [];

arr.push ("Apple");

arr.push ("Orange");

arr.push ("Banana");

Which is the proper method to print the value of an array? That is Apple, Orange, and Banana.

1. arr.forEach (function (value, index) {

console.log (value);

});

1. for (var value in arr) {

console.log (arr [value]);

}

1. for (var value in arr) {

console.log (value);

}

1. **Both A and B.**

Q87. What will be the length of Array after executing the following code?

window.onload = function () {

var arr = ["Apple", "Orange", "Banana"];

console.log (arr.length);

arr.length = 5;

arr[4] = "Grapes";

console.log (arr.length);

}

1. 0 and 3
2. 3 and 4
3. **3 and 5**
4. Error

Q88. Which JavaScript code is true, if you want to split array into new arraylist.

1. **window.onload = function () {**

**var array = ["Apple", "Orange", "Banana"];**

**var newArray = array.slice();**

**console.log (newArray [0]);**

**}**

1. window.onload = function () {

var array = ["Apple", "Orange", "Banana"];

var newArray = array.split();

console.log (newArray [0]);

}

1. window.onload = function () {

var array = ["Apple", "Orange", "Banana"];

var newArray = array.break();

console.log (newArray [0]);

}

1. None of the above.

Q89. What will be output of the following JavaScript code?

window.onload = function () {

var array = ["Apple", "Orange", "Banana"];

array.unshift ("Grapes");

console.log (array [0]);

}

1. Orange
2. Banana
3. **Grapes**
4. Error

Q90. Examine the following JavaScript code.

function myFunction (p1, p1) {

console.log (p1 + " " + p1);

}

myFunction ("Infoway", "Pune");

What will be the output?

1. Infoway Pune
2. **Pune Pune**
3. Error: because the parameter names in myFunction() are same.
4. None of the above.

Q91. Examine the following JavaScript code.

<script type="text/javascript" id="script1">

window.onload = function () {

var array = ["Apple", "Orange", "Banana"];

f1 (array);

}

</script>

Which of the following code you will user, so that it prints the data of an array. That is Apple, Orange, and Banana.

1. function f1(p1) {

p1.forEach (function (value, index) {

console.log (value);

});

1. for (var i = 0; i < p1.length; i++) {

console.log (p1 [i]);

}

1. **Both A and B**
2. None of the above

Q92. Examine the following JavaScript code.

var x = (20, 30);

console.log(x);

What will be the value of x?

1. 20
2. **30**
3. (20, 30)
4. Infinity

Q93. What will the result of JavaScript function?

var name1 = "AWP MCQs";

function DisplayName () {

var name2 = " Online";

console.log (name1 + name2);

}

DisplayName();

1. undefined Online
2. **AWP MCQs Online**
3. null Online
4. None of the above option.

Q94. There is JavaScript object called Person.

var Person = {

"firstName": "Saleel",

"lastName": "Bagde",

"display": function () {

/\* Replace code here \*/

}

}

Person.display ();

Which of the following option you will use to display firstName and lastName from the JavaScript object?

1. console.log (Person.FirstName + " " + Person.LastName);
2. console.log (Person["FirstName"] + " " + Person["LastName"]);
3. console.log (Person['FirstName'] + " " + Person['LastName']);
4. **All of the above.**

Q95. Examine the following Javascript code. What will be output of the variables w, x, y, and z.

var firstNumber = 10;

var secondNumber = 10;

var w = '10+10';

var x = 10 + 10;

var y = eval('10+10');

var z = eval('firstNumber + secondNumber')

console.log (w);

console.log (x);

console.log (y);

console.log (z);

1. 20 20 20 20
2. **10+10 20 20 20**
3. 10+10 20 20 'firstNumber + secondNumber'
4. 10+10 20 10+10 'firstNumber + secondNumber'

Q96. What will be the output for the following JavaScript code?

var x = 2;

var y = 5;

var a = eval("x \* y") + "" ;

var b = eval("2 + 2") + "";

var c = eval("x + 8") + "";

var result = a + b + c;

console.log (result);

1. **10410**
2. 104x+8
3. x\*y4x+8
4. x\*y2+2x+8

Q97. What will be the output for the following JavaScript code?

var a = true;

var str = 'if ( a ) { eval("1 + 1"); } else { 1 + 2; }';

var b = eval(str);

console.log (b);

1. **2**
2. 3
3. 1+1
4. 1+2

Q98. Examine the following JavaScript code.

var x = 10;

var y = 10;

var myObject = new Object();

Which of the following option will you use to do addition of variable x and variable y?

1. myObject.f1 = function (a, b) {

console.log (a+b);

}

myObject.f1(x, y);

1. myObject.f2 = (function (a, b) {

console.log (a + b);

}) (x, y);

1. myObject.f3 = function f3(a, b) {

console.log (a + b);

}

myObject.f3(x, y);

1. **Any one of the above.**

Q99. What will the output of the following JavaScript code?

var x = 0;

if (x = 0)

{

console.log("true");

}

else

{

console.log ("false");

}

1. 0
2. true
3. **false**
4. Error: because the comparison operator is invalid

Q100. Examine the following JavaScript code.

if (true) {

let y = 5;

}

console.log (y);

What will be the output?

1. 0
2. 5
3. Undefined
4. **Uncaught ReferenceError: y is not defined**

Q101. Consider the following code snippet:

var scope = "global scope";

function display() {

var scope = "local scope";

function f() {

return scope;

}

return f();

}

console.log (display ());

What will be the output?

1. global scope
2. **local scope**
3. Invalid display function.
4. None of the mentioned

Q102. Consider the following code snippet:

const scope = "global scope";

function display() {

var scope;

function f() {

return scope;

}

return f();

}

console.log (display ());

What will be the output?

1. **undefined**
2. local scope
3. Invalid display function.
4. None of the mentioned

Q103. What will be result of following JavaScript function?

<script type="text/javascript">

var name1 = "Infoway";

function DisplayName () {

var name2 = " The Best";

document.write (name1+name2);

}

</script>

1. InfowayThe Best
2. **Infoway The Best**
3. Object required error
4. Javascript Error

Q104. Consider the following code snippet:

var arr = [];

arr.unshift (1);

arr.unshift (22);

arr.shift();

arr.unshift (3, [4, 5]);

arr.shift();

arr.shift();

for (var i in arr) {

console.log (arr[i]);

}

The final output for the shift() is?

1. **1**
2. [4,5]
3. [3,4,5]
4. Exception is thrown

Q105. Examine the following jQuery code. What will the output?

var array = new Array (4);

array [0] = "Zero";

array [2] = "Two";

array [3] = "Three";

$(function () {

$(":button").click (function () {

console.log (array [1]);

});

});

1. null
2. **undefined**
3. empty
4. nothing

Q106. Which of the following jQuery code will display the name of the Fruits in console.log?

1. $(function () {

$(":button").click(function () {

$.each (['Orange', 'Apple', 'Grapes', 'Banana'], function (index, value) {

console.log (value);

});

});

});

1. $(function () {

$(":button").click(function () {

$.each (fruits, function (index, value) {

console.log (value);

});

});

});

1. fruits.forEach (function (value, index) {

console.log (value);

});

1. **All of the above.**

Q107. Examine the following code.

var Person = '[{"name":"saleel"}, {"name":"sharmin"}, {"name":"vrushali"}]';

// Replace code here

$.each (p, function (index, value) {

console.log (value.name);

});

Which of the following option you will use to parse the object?

1. var p = $.Person.parseJSON();
2. **var p = $.parseJSON(Person);**
3. var p = JSON.Parse(Person);
4. var p = Person.Parse(Person);

Q108. Consider the following code.

<!DOCTYPE html>

<html>

<head>

<title>Infoway Technology Pvt. Ltd, PUNE</title>

<meta charset="utf-8" />

</head>

<body>

<div class="container-fluid">

<ol>

<li>Apple</li>

<li>Orange</li>

<li>Banana</li>

</ol>

<button id="btn2">Append item</button>

</div>

</body>

</html>

Which of the following option you will use to add a new fruit?

1. **$(document).ready (function () {**

**var cnt = 1;**

**$("#btn2").click (function () {**

**$("ol").append ("<li><b >Newly added fruit [" +cnt++ +"] </b></li>");**

**});**

**});**

1. $(document).ready (function () {

var cnt = 1;

$("#btn2").click (function () {

$("ol").after ("<li><b >Newly added fruit [" +cnt++ +"] </b></li>");

});

});

1. $(document).ready (function () {

var cnt = 1;

$("#btn2").click (function () {

$("ol").appendAt ("<li><b >Newly added fruit [" +cnt++ +"] </b></li>");

});

});

1. None of the above.

Q109. Consider the following code.

<body>

<div class="container-fluid">

<ol>

<li class="c1">Apple</li>

<li class="c1">Orange</li>

<li class="c2">Banana</li>

</ol>

</div>

</body>

How do you change the background color of the first <li> tag in <ol> tag?

1. $("li.c1:first").css("background-color", "yellow");
2. $("li:nth-child(1)").css("background-color", "yellow");
3. $("li:firstChild").css("background-color", "yellow");
4. **Both A and B.**

Q110.

Q111.

Q112.

Q113.

Q114.

Q115.

Q116.

Q117.

Q118.

Q119.

Q120.

Q121.

Q122.

Q123.

Q124.

Q125.

Q126.

Q127.

Q128.

Q129.

Q130.

Q131.

Q132.

Q133.

Q134.

Q135.

Q136.

Q137.

Q138.

Q139.

Q140.

Q141.

Q142.

Q143.

Q144.

Q145.

Q146.

Q147.

Q148.

Q149.

Q150.