JavaScript Foundation Exam Answers

```
/*

1. Password Validator

Write a JavaScript program that checks if the entered
password matches the confirmed password. If the
passwords match, the program should log "Password
Matched. Password validation Successful." to the
console. Otherwise, it should log "Password didn't
match. Password validation unsuccessful" to the
console.



*/

let password = "pw@123";
let confirmPassword = "pw@123";

if (password === confirmPassword) {
  console.log("Password Matched. Password validation Successful.");
} else {
  console.log("Password didn't match. Password validation unsuccessful");
}
```

```
/*
2. Calculator

Create a javascript program that takes in two numbers and a string operator. Make use of a
switch statement to perform the operation on the two numbers.

The calculator function should:

- Take in two numbers, num1 and num2, and a string representing a mathematical operator,
operator.
- Use a switch statement to determine which operation to perform based on the value of the
operator.
- If the operator is one of the four valid operators (+, -, *, /), perform the corresponding
mathematical operation and store the result in a variable called result.
- If the operator is not one of the valid operators, log "Invalid operator" to the console.



*/

let num1 = 10;
let num2 = 12;

let operator = "+";

let result;

switch (operator) {
  case "+":
    result = num1 + num2;
    break;
  case "-":
    result = num1 - num2;
    break;
  case "*":
    result = num1 * num2;
    break;
  case "/":
    result = num1 / num2;
    break;
  default:
    console.log("Invalid operator");
    return;
}

console.log(result);
```

```javascript
/*

3. Color Mixer

Write a JavaScript program that takes in two strings representing colors
and uses a switch statement to determine the resulting color when the two
colors are mixed. The program should print the resulting color based on the
following criteria:

- If color1 is "red" and color2 is "blue" or vice versa, print "purple".
- If color1 is "red" and color2 is "yellow" or vice versa, print "orange".
- If color1 is "blue" and `color2" is "yellow" or vice versa, print
"green".
- If any other combination of colors is input, the program should print
"Invalid color combination".


*/

let color1 = "blue";
let color2 = "yello";

let result;

switch (color1 + color2) {
  case "redblue":
  case "bluered":
    result = "purple";
    break;
  case "redyellow":
  case "yellowred":
    result = "orange";
    break;
  case "blueyellow":
  case "yellowblue":
    result = "green";
    break;
  default:
    result = "Invalid color combination";
    break;
}

console.log(result);
```

```
/*

4. Highest Marks.

A teacher wants to find out the highest marks scored by
a student in a class of five students. The teacher
enters the marks of all five students in an array
called "marks". Write a program that iterates through
the array and finds the highest marks scored by any
student in the class. The highest marks must then be
displayed to the teacher using the console. Make sure
you use the ternary operator to find the student with
the highest marks.


*/

let marks = [1, 2, 3, 4, 5];

let highestMarks = marks[0];

for (let i = 1; i < marks.length; i++) {
  highestMarks = marks[i] > highestMarks ? marks[i] :
highestMarks;
}

console.log(highestMarks);
```

```
/*

5. Capitalize

You are building a form where users can enter their
names. You want to make sure that the first letter of
the name is always capitalized, even if the user
forgets to do so. Write a program that takes in the
user's name as a string and uses the ternary operator
to check if the first letter is lowercase. If it is,
the program capitalizes it and returns the modified
string. Otherwise, it returns the original string
without any changes.

*/

let userName = "mithun";

userName.length > 0 && userName[0] ===
userName[0].toLowerCase()
  ? console.log(userName[0].toUpperCase() +
userName.slice(1))
  : console.log(userName);
```

```javascript
/*

6. Vowel Counter.

We want to count the number of vowels in a person's
name. Given a name as input, the program should iterate
through each character in the name, and check if it is
a vowel or not. If the character is a vowel, it should
be counted.


*/

let name = "Uithun";

const vowels = ["a", "e", "i", "o", "u"];

let count = 0;

for (let i = 0; i < name.length; i++) {
  if (vowels.includes(name[i].toLowerCase())) {
    count++;
  }
}

console.log(count);
```

```
/*

7. Remove Duplicates.

In an online shopping application, customers can add multiple items
to their cart. However, sometimes customers accidentally add the same
item more than once, resulting in duplicate items in their cart. The
duplicate items not only make it difficult for the customer to track
the items they want to purchase but also affect the accuracy of the
purchase order.

To solve this problem, the application needs to remove duplicate
items from the customer's cart. The program should take the
customer's cart with duplicates as input, and return a new cart
without duplicates.

Write a program to solve the problem of duplicate items in cart by
removing duplicates.


*/

let cartWithDuplicates = [
  "Apple",
  "Apple",
  "Chilli",
  "Pine Apple",
  "Strawberry",
];

const cartWithoutDuplicates = [];

for (let i = 0; i < cartWithDuplicates.length; i++) {
  if (!cartWithoutDuplicates.includes(cartWithDuplicates[i])) {
    cartWithoutDuplicates.push(cartWithDuplicates[i]);
  }
}

console.log(cartWithoutDuplicates);
```

```
/*

8. Inverted right-angled triangle pattern with asterisks

Write a program that takes an integer input i and prints an inverted right-angled triangle pattern of asterisks with i rows.


*/

let i = 6;
while (i > 0) {
  let row = "";
  for (let j = 1; j <= i; j++) {
    row += "*";
  }
  console.log(row);
  i--;
}
```

```
/*

9. Check for divisibility.

Write a program that takes an array of numbers and
prints all the numbers that are divisible by 3, but not
by 2. Use a for loop and continue statement.


*/

let arr = [1, 2, 3, 4, 5, 6];

for (let i = 0; i < arr.length; i++) {
  if (arr[i] % 3 === 0 && arr[i] % 2 !== 0) {
    console.log(arr[i]);
    continue;
  }
}
```

```javascript
/*

10. Correct a bug.

You are working on an e-commerce website where
customers can add items to their cart. The cart stores
the quantity of each item that the customer wants to
purchase in an array of numbers. However, the website
is currently experiencing a bug where the quantity of
each item is being recorded incorrectly by reducing it
to half. As a result, you need to write a JavaScript
function that can double the quantity of each item in
the cart array to correct the bug.

*/

const numbers = [1, 2, 3, 4, 5];

const numbersDoubled = [];

function doubleNumber(num) {
  return num * 2;
}

for (let i = 0; i < numbers.length; i++) {
  numbersDoubled.push(doubleNumber(numbers[i]));
}

console.log(numbersDoubled);
```

```
/*

11. Unit convertor.

A local weather station needs to convert temperature
data collected in Celsius to Fahrenheit before
displaying it on its website. They want a function that
can convert Celsius to Fahrenheit accurately and
efficiently. The function should take input in Celsius
and return output in Fahrenheit. This function will
help the weather station to provide temperature
readings that are easily understandable to a wider
audience.



*/

let celsius = 30;

function celsiusToFahrenheit(celsius) {
  let fahrenheit = (celsius * 9) / 5 + 32;
  return fahrenheit;
}

console.log(celsiusToFahrenheit(celsius));
```

```
/*

12. Calculate rental cost

A car rental company needs to calculate the cost of a rental based on the
number of days rented and the type of car. They require a function that
takes in the number of days rented and car type and returns the rental
cost. The total cost would be the rental cost multiplied by the number of
days rented.

The rental costs are

- Economy = Rs. 4000 /- per day.
- Midsize = Rs. 10,000 /- per day.
- Luxury = Rs. 20,000 /- per day.



*/

function calculateRentalCost(numDays, carType) {
  let rentalCost = 0;
  switch (carType) {
    case "economy":
      rentalCost = 4000;
      break;
    case "midSize":
      rentalCost = 10000;
      break;
    case "luxury":
      rentalCost = 20000;
      break;
    default:
      console.log("Invalid car type");
      return;
  }
  return rentalCost * numDays;
}

console.log(calculateRentalCost(5, "midSize"));
```

```
/*

13. Bill splitter

A restaurant wants to calculate the total bill for a
table based on the cost of each dish and the number of
people sharing it. They require a function that takes
in the cost of each dish and the number of people
sharing it and returns an object that contains the
total bill and the bill to be paid by each person in
the group.


*/

function billSplitter(arrayOfCostPerDish,
numberOfPeople) {
  let totalBill = 0;

  for (let i = 0; i < arrayOfCostPerDish.length; i++) {
    totalBill += arrayOfCostPerDish[i];
  }

  return {
    totalBill: totalBill,
    billPerPerson: totalBill / numberOfPeople,
  };
}

console.log(billSplitter([100, 200], 2));
```

```
/*

14. Calculate the final order price.

A retail store needs to calculate the total cost of
items in a customer's cart. A customer cart is an array
of objects with unit price and quantity. Implement an
arrow function to calculate the total cost of items,
based on the unit price and quantity of each item.



*/

let cart = [
  {
    unitPrice: 5000,
    quantity: 2,
  },
];

const calculateTotalCost = (cartArray) => {
  let totalCost = 0;
  for (let item of cartArray) {
    totalCost += item.unitPrice * item.quantity;
  }
  return totalCost;
};

console.log(calculateTotalCost(cart));
```

```
/*

15. Calculate the percentage of the discount.

A retail store is offering a discount on its products
and wants to calculate the percentage of the discount
to show customers how much they can save. Given the
original price and the discounted price of a product,
implement an arrow function to calculate the percentage
of the discount rounded off to two decimal places. This
function could be useful for the store's marketing team
to create promotions and offers that attract customers.



*/

const calculateDiscountPercentage = (originalPrice,
discountedPrice) => {
  const discountAmount = originalPrice -
discountedPrice;
  const discountPercentage = (discountAmount /
originalPrice) * 100;
  return discountPercentage.toFixed(2);
};

console.log(calculateDiscountPercentage(5000, 4500));
```

```
/*

16. Generate a random number.

Create a JavaScript program that generates a random
number between 1 and 100 when the program starts. Use a
self-invoking arrow function to generate the random
number. This program can be used as a component in
various games or applications that require a random
number generator.



*/

const randomNumber = (() => {
  const min = 1;
  const max = 100;
  return Math.floor(Math.random() * (max - min + 1) +
min);
})();

console.log(randomNumber);
```

```
/*

17. Build a banking application.

A banking application needs to manage customer accounts and transactions.
The user detail is stored in an object with a keys name and balance.
Implement an object method to update a customer's account balance based on
a deposit or withdrawal.


*/

let users = [
  { id: "01", balance: 5000 },
  { id: "02", balance: 5000 },
  { id: "03", balance: 5000 },
  { id: "04", balance: 5000 },
];

function deposit(id, ammount) {
  for (let i = 0; i < users.length; i++) {
    if (users[i].id === id) {
      users[i].balance += ammount;
    }
  }
  return users;
}

function withDraw(id, ammount) {
  for (let i = 0; i < users.length; i++) {
    if (users[i].id === id) {
      users[i].balance -= ammount;
    }
  }
  return users;
}

console.log(deposit("01", 1000));
console.log(withDraw("03", 1000));
```

```html
<!-- 18. Change Text on Button click. -->

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Change Text on Button click.</title>
  </head>
  <body>
    <h1 id="headingText">The most affordable learning platform</h1>
    <button id="toggleButton">Change Text</button>
  </body>
  <!-- JS Starts -->
  <script>
    const text = document.getElementById("headingText");
    const button = document.getElementById("toggleButton");

    function changeText() {
      text.innerText == "The most affordable learning platform"
        ? (text.innerText = "PW Skills")
        : (text.innerText = "The most affordable learning platform");
    }

    button.addEventListener("click", changeText);
  </script>
  <!-- JS Ends -->
</html>
```

```
<!-- 19. Validate Password. -->
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Validate Password.</title>
  </head>
  <body>
    <form id="loginForm">
      <label for="email">Email:</label>
      <input type="text" id="email" name="email" required />
      <h1></h1>
      <label for="password">Password:</label>
      <input type="password" id="password" name="password" required />
      <h1></h1>
      <button type="submit" id="loginButton">Login</button>
    </form>
    <p id="errorMsg" style="color: red; display: none">
      Invalid email or password!
    </p>
  </body>
  <!-- JS Starts -->
  <script>
    const form = document.getElementById("loginForm");
    const email = document.getElementById("email");
    const password = document.getElementById("password");
    const errorMsg = document.getElementById("errorMsg");

    form.addEventListener("submit", (event) => {
      event.preventDefault();

      if (!email.value.includes("@") || password.value.length < 8) {
        errorMsg.style.display = "block";
      } else {
        errorMsg.style.display = "block";
        errorMsg.style.color = "green";
        errorMsg.innerText = "Valid email and password!";
      }
    });
  </script>
  <!-- JS Ends -->
</html>
```

```html
<!-- 20. Dynamically Adding List Items to an Ordered List. -->

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Dynamically Adding List Items to an Ordered List.</title>
  </head>
  <body>
    <ol id="listOfLearningObjectives"></ol>
    <button id="addListItem">Add list Item</button>
    <p id="message" style="color: green"></p>
    <!-- JS Starts -->
    <script>
      const arrayOfListItems = [
        "HTML and Semantics",
        "Starting with CSS",
        "Working Template",
        "Mobile responsive webpages",
        "Grid and Flex-box in CSS",
        "Projects using HTML & CSS",
        "Version Control and Git",
        "Getting Started with JavaScript",
        "Advance JavaScript",
        "Working with DOM",
        "Making Projects using HTML, CSS and JavaScript",
        "Understanding Fundamental of Computer Science",
        "Getting Started with Database",
        "Understanding the Database",
        "Starting with NodeJS and Express",
        "Understanding React and its Fundamentals",
        "Understanding Hooks and Routers",
        "Starting and Completing Full Fledge Projects",
      ];

      const list = document.getElementById("listOfLearningObjectives");
      const button = document.getElementById("addListItem");

      const message = document.getElementById("message");

      function addListItem() {
        const indexOfLastElementAdded = list.children.length;
        const newItem = document.createElement("li");

        if (arrayOfListItems[indexOfLastElementAdded + 1]) {
          newItem.textContent = arrayOfListItems[indexOfLastElementAdded + 1];
          list.appendChild(newItem);
        } else {
          message.innerHTML = "All Items Added";
        }
      }

      button.addEventListener("click", addListItem);
    </script>
    <!-- JS Ends -->
  </body>
</html>
```

```html
<!-- 21. TODO App. -->

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>TODO APP</title>
  </head>
  <body>
    <h1>TODO APP</h1>
    <label for="todoItems">Add Todo: </label>
    <input
      type="text"
      name="todoItems"
      id="todoItem"
      placeholder="Enter New TODO Item"
    />
    <button id="addItem">Add Item</button>
    <h3>Your TODO List</h3>
    <ul id="todoList"></ul>
  </body>
  <!-- JS Starts -->
  <script>
    const todoItem = document.getElementById("todoItem");
    const addItem = document.getElementById("addItem");
    const todoList = document.getElementById("todoList");

    function addItemToList() {
      if (todoItem.value != "") {
        const newItem = document.createElement("li");
        newItem.textContent = todoItem.value;
        todoList.appendChild(newItem);
        todoItem.value = "";
      }
    }

    addItem.addEventListener("click", addItemToList);
  </script>
  <!-- JS Ends -->
</html>
```