Question 1: Answer the following questions (10 points)

1. Difference between `var` and `let` keyword in JavaScript.

- Answer:

- `var` is function-scoped and can be re-declared within the same scope. It is also hoisted to the top of its scope.

- `let` is block-scoped and cannot be re-declared within the same scope. It is also hoisted, but not initialized until its declaration is evaluated.

2. Difference between `==` and `===` operators.

- Answer:

- `==` is the equality operator that performs type coercion, meaning it converts the operands to the same type before comparing.

- `===` is the strict equality operator that does not perform type coercion and compares both the value and type.

3. Difference between `while` and `for` loop.

- Answer:

- `while` loop continues to execute as long as the specified condition evaluates to true.

- `for` loop is typically used when the number of iterations is known. It has three parts: initialization, condition, and increment/decrement.

4. Difference between `if` and `switch`.

- Answer:

- `if` is used to execute a block of code if a specified condition is true. It is suitable for checking a few

conditions.

- `switch` is used to execute one block of code among many, based on the value of a variable. It is suitable for checking many conditions.

Question 2: What is the output (5 points)

a.

```
function test () {
  return {
    x: 1
  }
}
console.log(typeof(test()));
```

Answer:

- The output will be `"object"`, because `test()` returns an object.

b.

```
function sum(a, b) {
  return a + b;
}
console.log(sum(1, '2'));
```

Answer:

- The output will be `"12"`, because `'2'` is a string, so `1` is concatenated to `'2'`.

c.

```
let x = 10;

let y = x;

x = 20;

console.log(y);
```

Answer:

- The output will be `10`, because `y` is assigned the value of `x` before `x` is updated to `20`.

d.

```
let arr = ['foo', 'bar'];

for (let x in arr) {

   console.log(arr[x]);

   console.log(arr['x']);

}
```

Answer:

- The output will be:

foo

undefined

bar

undefined

`arr[x]` prints the elements of the array, whereas `arr['x']` prints `undefined` because `'x'` is not a valid key in the array.

e.

for (var i = 1; i < 10; i *= 2) {

}

console.log(i);

Answer:

- The output will be `16`, because `i` is incremented as `1, 2, 4, 8, 16` inside the loop, and then it prints the final value after the loop.

Question 3: Write JS Code to solve these problems (50 points)

1. Check if a string is a palindrome or not.

   - Answer:

   function isPalindrome(str) {

       const cleanedStr = str.replace(/[^A-Za-z0-9]/g, '').toLowerCase();

```
    return cleanedStr === cleanedStr.split('').reverse().join('');
}

console.log(isPalindrome("eye")); // true

console.log(isPalindrome("Welcome")); // false
```

2. Create a function to check if the given number is even.

- Answer:

- Using `if` statement without `else`:

```
function isEven(num) {
    if (num % 2 === 0) {
        return true;
    }
    return false;
}

console.log(isEven(4)); // true

console.log(isEven(5)); // false
```

- Using ternary operator:

```
const isEven = (num) => num % 2 === 0 ? true : false;
```

```
console.log(isEven(4)); // true

console.log(isEven(5)); // false
```

3. Find Largest Number in Array.

- Answer:

```
function findLargestNumber(arr) {
    return Math.max(...arr);
}

console.log(findLargestNumber([10, 100, 50, 4])); // 100
```

4. Write a function that takes an integer hours and converts it to seconds.

- Answer:

```
function hoursToSeconds(hours) {
    return hours * 3600;
}

console.log(hoursToSeconds(1)); // 3600

console.log(hoursToSeconds(2.5)); // 9000
```

5. Write a JS code to find the power of a number using `for` loop.

- Answer:

```
function power(base, exponent) {
    let result = 1;
    for (let i = 0; i < exponent; i++) {
        result *= base;
    }
    return result;
}

console.log(power(2, 3)); // 8

console.log(power(5, 4)); // 625
```

Question 4: Using HTML, CSS, and JS make this possible (15 points)

- The answer for this question can be found at the following link: [GitHub Link] (https://github.com/mahitab77/javascript-training/tree/main/js-change%20bg%20gd%20colors-dark%20light%20theme)

Question 5: (20 points)

1. Refactor this code by doing it in another two ways.

Original Code:

```
const KeyNum = Number(prompt('enter key num:'));

if (key === 1) {
```

```
    console.log("the key is on");
} else {
    console.log("the key is off");
}
```

Refactor 1: Using Ternary Operator

```
const KeyNum = Number(prompt('enter key num:'));
console.log(key === 1 ? "the key is on" : "the key is off");
```

Refactor 2: Using Switch Statement

```
const KeyNum = Number(prompt('enter key num:'));
switch (key) {
    case 1:
        console.log("the key is on");
        break;
    default:
        console.log("the key is off");
}
```

2. Refactor this code using `for` loop

Original Code:

```
let i = 5;

while (i >= 1) {
   console.log(i);
   i--;
}
```

Refactored Code:

```
for (let i = 5; i >= 1; i--) {
   console.log(i);
}
```

3. True or false.

1. JavaScript is async, blocking and single-threaded language.
   - Answer:False. JavaScript is async, non-blocking, and single-threaded.
2. Const arr = [1, 2, 3, 4]. The output of `console.log(typeof(arr))` will be array.
   - Answer:False. The output will be `object`.