DUMPSWAY

JS Institute

JSE-40-01

JSE – Certified Entry-Level JavaScript Programmer

QUESTION & ANSWERS

Demo Version

QUESTION: 1

Review the following code:

```
let x = 10;

function test(x) {

    console.log(x);

}

test(20);
```

What will be displayed in the console as a result of its execution?

Option A :

"x"

Option B :

20

Option C :

10

Option D :

Nothing will show up.

Correct Answer: B

Explanation/Reference:

Topics: function declaration parameter

Try it yourself:

```
let x = 10;
```

```
function test(x) {
   console.log(x); // 20
}
test(20);
```

Explanation:

The value 20 will be passed to the function test()

and assigned to the parameter x

and then displayed.

The function test() has its own scope and therefore it does not matter

that there is also a global variable named x

The function declaration (function statement) defines a function.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/function

A parameter is a named variable passed into a function.

https://developer.mozilla.org/en-US/docs/Glossary/Parameter

QUESTION: 2

If the interpreter recognizes from the syntax of code that a word represents a function,

but is unable to find such a function (e.g. through a typo), it is:

Option A :

a syntax error

Option B :

a semantic error

Option C :

a logical error

Option D :

an arithmetic error

Correct Answer: B

Explanation/Reference:

Topics: semantic error

Try it yourself:

```
function test1() {
   console.log("Test 1");
}
test2(); // Uncaught ReferenceError: test2 is not defined
```

Explanation:

This ReferenceError is a semantic error.

Semantic error

A programming error that arises from a misunderstanding

of the meaning or effect of some construct in a programming language.

https://www.encyclopedia.com/computing/dictionaries-thesauruses-pictures-and-press-releases/semantic-error

QUESTION: 3

What does shadowing mean?

Option A :

Declaring a local variable with the same name

as the previously declared global variable.

Option B :

Declaring a global variable with the same name

as a previously declared global variable.

Option C :

Deleting and rewriting a selected piece of program code.

Option D :

Changing the value of a variable.

Correct Answer: A

Explanation/Reference:

Try it yourself:

```
let x = 23;
{
   let x = 42;
   console.log(x);  // 42
}
console.log(x);  // 23
```

Explanation:

Shadowing

When a variable is declared in a certain scope

having the same name defined on its outer scope

and when we call the variable from the inner scope,

the value assigned to the variable in the inner scope

is the value that will be stored in the variable in the memory space.

https://www.geeksforgeeks.org/variable-shadowing-in-javascript

QUESTION: 4

Placing a debugger; statement in the program code will:

Option A :

pause the program with the ability to continue,

as long as the execution environment supports "debugging functionality".

Option B :

stop the program without the ability to continue,

as long as the execution environment supports "debugging functionality".

Option C :

put the interpreter into report mode,

which will cause the console to print out all sequentially executed commands.

Option D :

cause the console to display the completion status

of the statement preceding the debugger.

Correct Answer: A

Explanation/Reference:

Topic: debugger statement

Try it yourself:

console.log("Before debugger");

debugger;

console.log("After debugger");

Explanation:

The debugger statement stops the execution of JavaScript,

and calls the debugger.

https://www.w3schools.com/jsref/jsref_debugger.asp

QUESTION: 5

The TypeScript language is:

Option A :

a new language based on JavaScript which,

among other things, introduces static typing.

Option B :

an alternative name for JavaScript.

Option C :

the original name for JavaScript, which has been changed over time.

Option D :

a variant of JavaScript that is compiled rather than interpreted.

Correct Answer: A

Explanation/Reference:

Topic: TypeScript

Explanation:

TypeScript is a strongly typed programming language that builds on JavaScript.

https://www.typescriptlang.org

TypeScript provides static typing

through type annotations to enable type checking at compile time.

https://en.wikipedia.org/wiki/TypeScript#Type_annotations

QUESTION: 6

Performing the operation:

let x = 100 / 0;

will result in:

Option A :

the value NaN being stored in the variable x

Option B :

an Infinity value being stored in the variable x

Option C :

the value 0 being stored in the variable x

Option D :

an undefined value being stored in the variable x

Correct Answer: B

Explanation/Reference:

Topics: Infinity NaN undefined

Try it yourself:

let x = 100 / 0;

console.log(x); // Infinity

Explanation:

Most programming languages will raise an exception if you divide by zero.

JavaScript returns the value Infinity

Infinity is a number that represents positive infinity.

https://www.w3schools.com/jsref/jsref_infinity.asp

NaN is short for "Not-a-Number" and is a number that is not a legal number.

https://www.w3schools.com/jsref/jsref_nan.asp

The undefined property indicates that a variable

has not been assigned a value, or not declared at all.

https://www.w3schools.com/jsref/jsref_undefined.asp

QUESTION: 7

Analyze the following code:

let height;

console.log(height);

Executing it will:

Option A :

display the null value in the console.

Option B :

display the undefined value in the console.

Option C :

cause the program to terminate with an error

because we are referring to a variable that has no value.

Option D :

display the message "height" in the console.

Correct Answer: B

Explanation/Reference:

Try it yourself:

let height;

console.log(height); // undefined

Explanation:

The undefined property indicates that a variable has not been assigned a value,

or not declared at all.

https://www.w3schools.com/jsref/jsref_undefined.as

QUESTION: 8

The result of the operation 3 * 4 > 20 - 15 will be:

Option A :

NaN

Option B :

False

Option C :

-14

Option D :

true

Correct Answer: D

Explanation/Reference:

Topics: multiplication operator greater than operator

subtraction operator

Try it yourself:

```
console.log(3 * 4 > 20 - 15);  // true
console.log(12 > 20 - 15);     // true
console.log(12 > 5);           // true
```

Explanation:

12 is greater than 5

The multiplication operator produces the product of the operands.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Multiplication

The greater than operator returns true if the left operand is

greater than the right operand, and false otherwise.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Greater_than

The subtraction operator subtracts the two operands,

producing their difference.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Subtraction

QUESTION: 9

Analyze the following code:

```
let n = 100;
switch(a) {
    case 0: break;
    case 1: n++; break;
    case 2: n--; break;
    default: n = 0;
}
```

What does the a variable contain if after executing

the code above the n variable contains 101?

Option A :

1

Option B :

true

Option C :

100

Option D :

2

Correct Answer: A

Explanation/Reference:

Topics: switch postfix increment operator

postfix decrement operator

Try it yourself:

```
let a = 1;

let n = 100;

switch(a) {

   case 0: break;

   case 1: n++; break;

   case 2: n--; break;

   default: n = 0;

}

console.log(n)  // 101
```

Explanation:

a is 1 and case 1 will be executed.

n will be incremented by one and will be 101 after the switch

The switch statement executes a block of code depending on different cases.

https://www.w3schools.com/jsref/jsref_switch.asp

The increment operator increments its operand and returns a value.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Increment

The decrement operator decrements its operand and returns a value.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Decrement

QUESTION: 10

We want to declare a protocol constant and initialize it with the value "http"

What should such a declaration look like?

Option A :

let const protocol = "http";

Option B :

let protocol; const protocol = "http";

Option C :

const protocol; protocol = "http";

Option D :

const protocol = "http";

Correct Answer: D

Explanation/Reference:

Topic: constant

Try it yourself:

const protocol1 = "http";

// let const protocol2 = "http";

// Uncaught SyntaxError: unexpected token: keyword 'const'

// const protocol3; protocol3 = "http";

// Uncaught SyntaxError: missing = in const declaration

// let protocol4; const protocol4 = "http";

// Uncaught SyntaxError: redeclaration of let protocol4

Explanation:

The value of a constant can't be changed through reassignment

and it can't be redeclared.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/const

QUESTION: 11

In the animals variable, we store an array of animal names.

We want to enlarge this array by appending elements

of the anotherAnimals array to it.

To do this, we can call the command:

Option A :

animals.merge(anotherAnimals);

Option B :

animals = animals.concat(anotherAnimals);

Option C :

anotherAnimals.concat(animals);

Option D :

```
animals.concat(anotherAnimals);
```

Correct Answer: B

Explanation/Reference:

Topic: array.concat()

Try it yourself:

```
let animals1 = ["dog", "cat", "mouse"];

let anotherAnimals1 = ["tiger", "lion"];

animals1 = animals1.concat(anotherAnimals1);

console.log(animals1);

// [ "dog", "cat", "mouse", "tiger", "lion" ]

// animals2.merge(anotherAnimals2);

// Uncaught TypeError: animals2.merge is not a function

let animals3 = ["dog", "cat", "mouse"];

let anotherAnimals3 = ["tiger", "lion"];

animals3.concat(anotherAnimals3);

console.log(animals3);  // [ "dog", "cat", "mouse" ]

let animals4 = ["dog", "cat", "mouse"];

let anotherAnimals4 = ["tiger", "lion"];

anotherAnimals4.concat(animals4);

console.log(animals4);  // [ "dog", "cat", "mouse" ]
```

Explanation:

Because the concat() method does not change the existing arrays

you have to assign it to the original variable.

The array.concat() method concatenates (joins) two or more arrays.

This method does not change the existing arrays,

but instead returns a new array.

https://www.w3schools.com/jsref/jsref_concat_array.asp

QUESTION: 12

Analyze the following code:

```
let numbers = [10, 20, 30];
let x = 0;
for(let n of numbers)
   x = x + n;
console.log(x);
```

What will appear on the console as a result of its execution?

Option A :

60

Option B :

3

Option C :

0

Option D :

10

20

30

Correct Answer: A

Explanation/Reference:

Topic: for of over array

Try it yourself:

```
let numbers = [10, 20, 30];

let x = 0;

for(let n of numbers)

   x = x + n;

console.log(x);  // 60
```

Explanation:

The loop determines the sum of the array.

In each iteration one element of the array is added to x

and in the end x is the sum of all elements of the array.

The for of statements combo iterates over the values of any iterable.

The code block inside the loop is executed once for each value.

https://www.w3schools.com/jsref/jsref_forof.asp

QUESTION: 13

Examine the following code:

```
for (let a = 5; a > 1; a--) {

   console.log(a);

}
```

Which statement can replace the for from the example above?

Option A :

```
let a = 6;
while (a >= 1) console.log(a--);
```

Option B :

```
let a = 5;
while (a > 1) console.log(a--);
```

Option C :

```
let a = 1;
while (a < 5) console.log(a++);
```

Option D :

```
let a = 5;
while (a > 1) console.log(a++);
```

Correct Answer: B

Explanation/Reference:

Try it yourself:

```
for (let a = 5; a > 1; a--) {
   console.log(a);  // 5 4 3 2
}
console.log("a1");
let a1 = 5;
while (a1 > 1) console.log(a1--);     // 5 4 3 2
console.log("a2");
```

```
let a2 = 1;

while (a2 < 5) console.log(a2++);     // 1 2 3 4

console.log("a3");

let a3 = 6;

while (a3 >= 1) console.log(a3--);    // 6 5 4 3 2 1

console.log("a4");

let a4 = 5;

// while (a4 > 1) console.log(a4++);  // Infinite loop
```

Explanation:

You need the same three parts in the while loop:

a = 5

a > 1

a--

The for statement defines a code block that is executed

as long as a condition is true

https://www.w3schools.com/jsref/jsref_for.asp

The while statement creates a loop (around a code block)

that is executed while a condition is true

https://www.w3schools.com/jsref/jsref_while.asp

QUESTION: 14

Entering about:blank in the address bar of your browser will:

Option A :

reset the browser to its default settings.

Option B :

generate and load a minimal blank HTML page into the current tab.

Option C :

open a tab with information about your browser.

Option D :

generate a page with information about the browser's status

and send it to the developer.

Correct Answer: B

Explanation/Reference:

Explanation:

about:blank references a blank HTML document.

This is widely used to load blank pages into browsing contexts.

https://en.wikipedia.org/wiki/About_URI_scheme

https://www.howtogeek.com/656466/what-is-aboutblank-and-how-do-you-remove-it

QUESTION: 15

Which method will display a dialog box

that allows the user to type in any string?

Option A :

fill

Option B :

alert

Option C :

confirm

Option D :

prompt

Correct Answer: D

Explanation/Reference:

Topic: prompt()

Try it yourself:

let name = prompt("Please enter your name!");

console.log(name);

Explanation:

The prompt() method displays a dialog box that prompts the user for input.

The prompt() method returns the input value if the user clicks "OK",

otherwise it returns null

https://www.w3schools.com/jsref/met_win_prompt.asp

QUESTION: 16

Which of the following declarations using the String type is incorrect?

Option A :

let msg = "The vessel \"Mars\" called at the port";

Option B :

let msg = "The vessel 'Mars' called at the port";

Option C :

let msg = 'The vessel "Mars" called at the port';

Option D :

let msg = "The vessel "Mars" called at the port";

Correct Answer: D

Explanation/Reference:

Topics: string quotes

Try it yourself:

// let msg1 = "The vessel "Mars" called at the port";

// Uncaught SyntaxError: unexpected token: identifier

let msg2 = "The vessel 'Mars' called at the port";

console.log(msg2); // The vessel 'Mars' called at the port

let msg3 = 'The vessel "Mars" called at the port';

console.log(msg3); // The vessel "Mars" called at the port

let msg4 = "The vessel \"Mars\" called at the port";

console.log(msg4); // The vessel "Mars" called at the port

Explanation:

You can use quotes inside a string,

as long as they don't match the quotes surrounding the string.

The sequence \" inserts a double quote in a string.

https://www.w3schools.com/js/js_strings.asp

QUESTION: 17

Analyze the following code:

```
try {
    ocnsole.log("start");
} catch (error) {
    console.log("error");
}
console.log("end");
```

What will happen as a result of its execution?

Option A :

In the console, there will appear in successive

lines the words "error", "end"

Option B :

The words "start", "error", "end" will appear

in the console on successive lines.

Option C :

The operation of the program will be interrupted

and the console will display the default message

"Uncaught ReferenceError: ocnsole is not defined"

Option D :

The following words will appear in the console: "start", "end"

Correct Answer: A

Explanation/Reference:

Try it yourself:

```
try {
    ocnsole.log("start");
} catch (error) {
    console.log("error");  // error
}
console.log("end");  // end
// ocnsole.log("start");
// Uncaught ReferenceError: ocnsole is not defined
```

Explanation:

There is a typo in console (it is written ocnsole)

therefore an exception is raised, the catch block is executed

and "error" is printed to the console.

After the try catch statements "end" is printed to the console.

The try catch statements combo handles errors without stopping JavaScript.

https://www.w3schools.com/jsref/jsref_try_catch.asp

QUESTION: 18

Which of the following loop instructions is intended only to loop

through all the keys of the indicated object?

Option A :

for ... of

Option B :

do ... while

Option C :

if ... else

Option D :

for ... in

Correct Answer: D

Explanation/Reference:

Topic: for in over object

Try it yourself:

let animal = {name: "Snoop", age: "3"};

for (let n in animal) console.log(n); // name age

Explanation:

The for in statements combo iterates over the properties of an object.

The code block inside the loop is executed once for each property.

https://www.w3schools.com/jsref/jsref_forin.asp

QUESTION: 19

We want to measure how long a certain piece of code executes.

In order to do so, it is enough to:

Option A :

precede the fragment with the command timeStart()

and end with timeEnd()

Option B :

precede the fragment with the command console.time("counter")

and end with console.timeEnd("counter")

Option C :

precede the fragment with the command console.timeStart("counter")

and end with console.timeEnd("counter")

Option D :

precede the fragment with the command console.time("start")

and end with console.timeEnd("end")

Correct Answer: B

Explanation/Reference:

Topics: console.time() console.timeEnd()

Try it yourself:

```
console.time("counter")

let num;
```

```
    do {
    num = Math.floor(Math.random() * 10000);
    console.log(num);
} while (num != 7777);
console.timeEnd("counter")  // e.g. counter: 187ms - timer ended
```

Explanation:

The time() method starts a timer in the console view.

https://www.w3schools.com/jsref/met_console_time.asp

The timeEnd() method ends a timer, and writes the result to the console.

https://www.w3schools.com/jsref/met_console_timeend.asp

QUESTION: 20

Analyze the following code snippet:

let n = 5 > 2 ? 10 : 20;

What is the result stored in the n variable?

Option A :

5

Option B :

2

Option C :

10

Option D :

20

Correct Answer: C

Explanation/Reference:

Topics: conditional (ternary) operator

Try it yourself:

```
let n = 5 > 2 ? 10 : 20;

console.log(n);               // 10

console.log(5 > 2 ? 10 : 20);  // 10

console.log(true ? 10 : 20);   // 10

console.log(10);              // 10
```

Explanation:

5 > 2 evaluates to true

and therefore 10 is assigned to n

The conditional (ternary) operator is the only JavaScript operator

that takes three operands: a condition followed by a question mark (?),

then an expression to execute if the condition is truthy followed by a colon (:),

and finally the expression to execute if the condition is falsy.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Conditional_Operator

QUESTION: 21

We can replace the declaration

let x = 0x21;

with:

Option A :

let x = 17;

Option B :

let x = "0x21";

Option C :

let x = 21;

Option D :

let x = 33;

Correct Answer: D

Explanation/Reference:

Topic: hexadecimal number

Try it yourself:

let x = 0x21;

console.log(x); // 33

console.log(2 * 16 + 1); // 33

Explanation:

Hexadecimal number syntax uses a leading zero

followed by a lowercase or uppercase Latin letter "X" (0x or 0X).

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Numbers_and_dates#hexadecimal_numbers

QUESTION: 22

Analyze the following code:

let x = 10 / 100;

console.log(typeof (x));

As a result of its execution:

Option A :

an error will appear because JavaScript does not

allow operations on fractional numbers.

Option B :

it will display "fraction" in the console.

Option C :

it will display "number" in the console.

Option D :

it will display 0.1 in the console.

Correct Answer: C

Explanation/Reference:

Topics: typeof operator division operator

Try it yourself:

let x = 10 / 100;

console.log(x); // 0.1

console.log(typeof (x)); // number

Explanation:

The typeof operator returns a string indicating the type

of the unevaluated operand.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/typeof

The division operator (/) produces the quotient of its operands where the left

operand is the dividend and the right operand is the divisor.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Division

Q520 (Please refer to this number, if you want to write me about this question.)

QUESTION: 23

Using the debugger, we insert a breakpoint in the code at which,

after running the program, we stop.

In the debugger, we find a Step In button among

the step-by-step operation options.

What does pressing it do?

Option A :

The program will be restarted from the first instruction.

Option B :

The program will be executed to the end,

regardless of whether there are more breakpoints

in the rest of the code or not.

Option C :

Exactly one instruction immediately after the breakpoint

will be executed and the program will be paused again.

Option D :

The program will execute to the end or to the next breakpoint.

Correct Answer: C

Explanation/Reference:

Topics: debugger breakpoint step through code

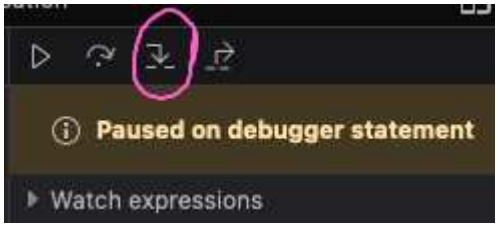
Try it yourself:

```
debugger;

let x = 7;

let y = 23;

let z = 42;
```



The JavaScript debugger enables you to step through JavaScript code

and examine or modify its state to help track down bugs.

https://developer.mozilla.org/en-US/docs/Tools/Debugger

At a breakpoint you can do useful things like studying the value of different

variables at that point, allowing you to work out why a problem is occurring.

https://developer.mozilla.org/en-US/docs/Tools/Debugger/How_to/Set_a_breakpoint

Step through code
When the debugger is stopped at a breakpoint,
you can step through it using four buttons in the toolbar.
https://developer.mozilla.org/en-US/docs/Tools/Debugger/How_to/Step_through_code

QUESTION: 24

Review the following code:

```
for (let a = 4; a < 4; a++) {
    console.log("test");
}
```

How many times will "test" be displayed in the console

as a result of its execution?

Option A :

Four

Option B :

One

Option C :

It will not be displayed at all.

Option D :

Three

Correct Answer: C

Explanation/Reference:

Topics: for less than operator

postfix increment operator

Try it yourself:

```
for (let a = 4; a < 4; a++) {
```

```
  console.log("test");
}
```

Explanation:

The condition is never met.

In the beginning a is 4 and thereby not less than 4

and the loop never starts.

The for statement defines a code block

that is executed as long as a condition is true

https://www.w3schools.com/jsref/jsref_for.asp

The less than operator returns true if the left operand is

less than the right operand, and false otherwise.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Less_than

The increment operator increments its operand and returns a value.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Increment

QUESTION: 25

We have defined an arrow function:

let multiply = (m, n) => m * n;

What will a regular declaration of the corresponding function look like?

Option A :

```
function multiply(m, n) {
  return m * n;
```

Option B :

```
function multiply(m, n) {
  m * n;
```

```
}
```

Option C :

```
function multiply(m, n) {
    m = m * n;
```

Option D :

```
function multiply(m, n) {
    n = m * n;
```

Correct Answer: A

Explanation/Reference:

Topics: arrow function function declaration

Try it yourself:

```
let multiply = (m, n) => m * n;
console.log(multiply(3, 4))  // 12

function multiply1(m, n) {
   return m * n;
}
console.log(multiply1(3, 4))  // 12
function multiply2(m, n) {
   m = m * n;
}
console.log(multiply2(3, 4))  // undefined
```

```
function multiply3(m, n) {
   n = m * n;
}
console.log(multiply3(3, 4))  // undefined
function multiply4(m, n) {
   m * n;
}
console.log(multiply4(3, 4))  // undefined
```

Explanation:

There is only one function declaration with a return value.

Arrow functions allow us to write shorter function syntax.

https://www.w3schools.com/js/js_arrow_function.asp

The function declaration (function statement) defines a function.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/function

QUESTION: 26

Point out the correct declaration of the temperature variable:

Option A :

temperature is variable;

Option B :

variable temperature;

Option C :

declare temperature;

Option D :

let temperature;

Correct Answer: D

Explanation/Reference:

Try it yourself:

let temperature;

// declare temperature;

// Uncaught SyntaxError: unexpected token: identifier

// variable temperature;

// Uncaught SyntaxError: unexpected token: identifier

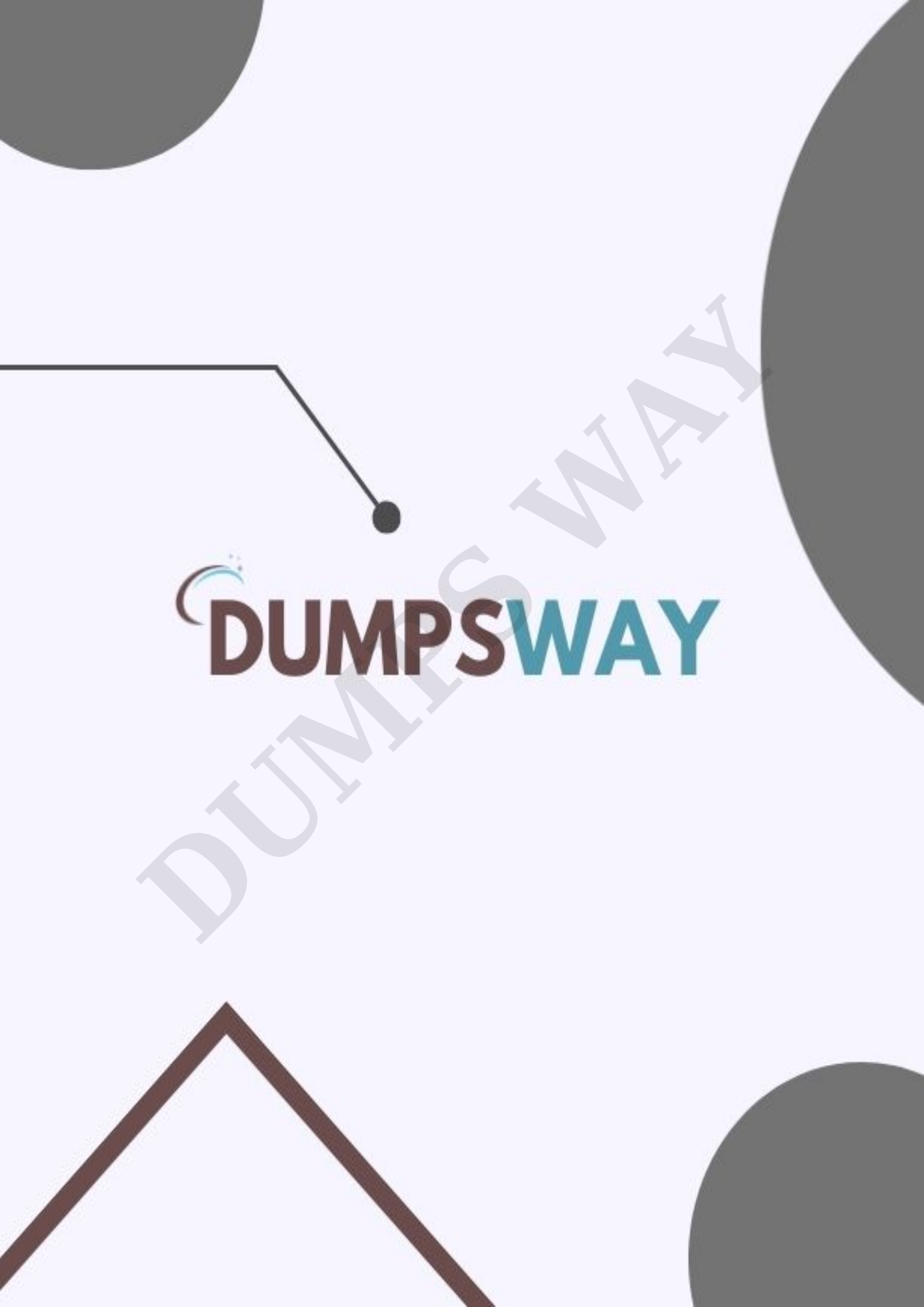
// temperature is variable;

// Uncaught SyntaxError: unexpected token: identifier

Explanation:

The let statement declares a variable.

https://www.w3schools.com/jsref/jsref_let.asp


DUMPSWAY
DUMPS WAY