

JS Functions

Total points 0/0

Functions

Email *

rpeddi7@gmail.com

✓ What will this code print to the console? *

```
const sleepTimer = (alarm) => {  
  console.log('My alarm is set for: ' + alarm);  
}
```

```
sleepTimer('8:30AM');
```

- ☒ My alarm is set for: 8:30AM
- ☐ My alarm is set for: undefined
- ☐ My alarm is set for: alarm
- ☐ My alarm is set for: sleepTimer

**Feedback**

Correct! '8:30AM' is the argument passed to sleepTimer().



✓ Which of the following is a parameter in the block of code below? *

```
let input = 8;  
const controlVal = input / 2 + 3;  
const multiplier = (number, phase) => {  
  const val = number * controlVal + phase;  
  console.log(val);  
};
```

☐ controlVal

☐ input

☐ val

☒ number



Feedback



Correct!



✓ What is wrong with the code snippet provided? *

```
const greeting = => {  
  console.log('Hello Programmer!');  
};
```

- ☐ The ordering of = and => should be switched.
- ☐ The function expression cannot be declared with a const keyword.
- ☐ The curly braces {} should be parentheses ().
- ☒ The greeting function is missing a set of () between the = and =>. ✓

Feedback



Correct! When there are no parameters, an arrow function needs to have a set of empty parentheses ().

✓ Which correctly represents the most condensed form of the function? Recall * that this syntax is also known as 'concise body.'

- ☐ const areaOfCircle = radius => { return Math.PI * radius * radius };
- ☒ const areaOfCircle = radius => Math.PI * radius * radius; ✓
- ☐ const areaOfCircle = radius => { Math.PI * radius * radius };
- ☐ const areaOfCircle = radius => return Math.PI * radius * radius;

Feedback



Correct! Concise body syntax (with one parameter) does not use parentheses, curly braces, or the return keyword.



- ✓ This code prints: **I ran 3 miles at an average of undefined per mile.** Why does the text include **undefined**? *

```
const workoutJournal = (miles, avgTime) => {  
  console.log('I ran ' + miles + ' miles at an average of ' + avgTime + ' per  
  mile.');
```

```
workoutJournal('3');
```

- ☐ The miles parameter is not declared.
- ☐ workoutJournal is not defined.
- ☐ workoutJournal does not print the value of both arguments.
- ☒ The call to workoutJournal is missing a second argument. ✓

Feedback



Correct!



✓ Which of the following most accurately describes the **volumeOfCube()** function? *

```
function volumeOfCube (side) {  
  return side * side * side;  
}
```

```
volumeOfCube(5);  
// Output: 125.
```

- ☐ It is a function expression.
- ☒ It is a function declaration. ✓
- ☐ It is an arrow function.
- ☐ It is an anonymous function.

Feedback



Correct! This is a function declaration. A function expression requires a variable assignment like `const volumeOfCube() = function () {...`

✓ What's the purpose of a parameter? *

- ☐ To call a function.
- ☒ To allow a function to accept data. ✓
- ☐ To specify actual values passed to a function.

Feedback

Arguments are the values passed into a function. Parameters act as placeholders for argument values.



✓ Which of the following best describes what a function in JavaScript is used for? *

- ☐ A function allows for the use of mathematical operators.
- ☒ A function is a reusable piece of code that can accept input and performs a specific task. ✓
- ☐ A function creates new variables.
- ☐ A function stores data.

Feedback



Correct!

✓ What will be printed to the console? *

```
const eatFruit = (fruit = 'apple') => {  
  console.log(`This ${fruit} is delicious!`);  
};
```

`eatFruit();`

- ☒ This apple is delicious! ✓
- ☐ This fruit is delicious!
- ☐ This is delicious!
- ☐ This undefined is delicious!

Feedback



Correct! `eatFruit()` has a default value of 'apple' for the fruit parameter.



✓ Which of the following is the correct way to call the function below? *

```
const multiplier = (number) => {  
  console.log(3 * number);  
};
```

☐ multiplier{5}

☒ multiplier(5)



☐ multiplier 5

☐ multiplier[5]

Feedback



Correct!

This form was created inside of Mr. & Mrs. Cloud.

Google Forms



