1. In JavaScript, there are many built-in *constructors* that create objects. A constructor is like a regular function, but starts with a capital letter, and is initialized with the new operator.

For example, you can use the Date() constructor with the new operator to create a new Date object that returns a string with the current date and time:

Example Code

const currentDate = new Date();

console.log(currentDate);

// Output:

// Mon Aug 23 2021 15:31:00 GMT-0400 (Eastern Daylight Time)

Create a new const variable called date and assign it a Date object with new Date().



Check Your Code (Ctrl + Enter)

Reset



Navigated to Step 3

# 2. Step 4

The Date object has a number of methods that allow you to get the date and time in different formats.

One of those is the .getDate() method, which returns a number between 1 and 31 that represents the day of the month for that date. For example:

Example Code

const date = new Date();

const dayOfTheMonth = date.getDate();

console.log(dayOfTheMonth); // 20

Using const, create a variable named day and assign it the day of the month from date with the .getDate() method.

3. The .getMonth() method returns a number between 0 and 11. This represents the month for the date provided, where 0 is January and 11 is December. Because the number this method returns is zero-based, you need to add 1 to it to get the expected month number.

Using const, create a variable named month and assign it the month from date with the .getMonth() method.

Remember to add 1 to the number returned by .getMonth().

4. In JavaScript, the change event is used to detect when the value of an HTML element has changed:

Example Code

element.addEventListener("change", () => {

});

Attach the addEventListener method to the dateOptionsSelectElement. The first argument of the event listener should be the string "change" and the second argument should be an empty arrow function.

5. When a user makes a selection from the dropdown menu, the function should get the user's value and display the date in their chosen date format. To do this, you can use the switch statement.

A switch statement is used to compare an expression against multiple possible values and execute different code blocks based on the match. It's commonly used for branching logic.

For example, here's how to compare the expression dayOfWeek against possible values:

Example Code

switch (dayOfWeek) {

case 1:

console.log("It's Monday!");

break;

case 2:

console.log("It's Tuesday!");

break;

// ...cases for other workdays

default:

console.log("It's the weekend!");

}

Create a switch statement and use dateOptionsSelectElement.value as the expression.

6. To format the date into yyyy-mm-dd, you will need to use the *split*, *reverse*, and *join* methods. But first, you will need to go through a few practice examples so you can better understand how to use them in the context of this project.

The split() method is used to divide a string into substrings based on a specified separator. It then returns these substrings as elements of an array.

Here is an example of taking the words "Hello World" and returning an array of one element:

Example Code

const greeting = "Hello World";

greeting.split(); // ["Hello World"]

Create a new const variable called exampleSentence and assign it the result of "selur pmaCedoCeerf".split().

Then add a console statement to log the value of exampleSentence. Open up the console to see the result.

7. The split method takes in a parameter known as a separator. The separator is used to tell the computer where each split should occur.

Here is an example of using an empty string as a separator:

Example Code

// returns ["h", "e", "l", "l", "o"]

"hello".split("");

Other examples of separators can include a space " ", or a hyphen "-". If you don't provide a separator, the method will return an array with the original string as the only element.

Update your split method, to use an empty string as a separator. Open up the console again to see the resul

8. To reverse an array of elements, you can use the reverse method. This method reverses the order of the elements in the array in place. The first element becomes the last, and the last element becomes the first.

Here is an example of using the reverse method:

Example Code

// returns [5, 4, 3, 2, 1]

[1, 2, 3, 4, 5].reverse();

Chain the reverse method to your split method. Open up the console again to see the result.

Remember that you learned how to chain methods in the previous project like this:

Example Code

method1().method2().method3();



Check Your Code (Ctrl + Enter)

Reset



Navigated to Step 20