2) Create a Node.js file that appends the existing file with new.

Ans.—

The **fs.appendFile() method** is used to asynchronously append the given data to a file. A new file is created if it does not exist. The options parameter can be used to modify the behavior of the operation.

**Syntax:**

fs.appendFile( path, data[, options], callback )

**Parameters:**

This method accepts four parameters as mentioned above and described below:

* **path:** It is a String, Buffer, URL or number that denotes the source filename or file descriptor that will be appended to.
* **data:** It is a String or Buffer that denotes the data that has to be appended.
* **options:** It is an string or an object that can be used to specify optional parameters that will affect the output. It has three optional parameters:
  + **encoding:** It is a string which specifies the encoding of the file. The default value is ‘utf8’.
  + **mode:** It is an integer which specifies the file mode. The default value is ‘0o666’.
  + **flag:** It is a string which specifies the flag used while appending to the file. The default value is ‘a’.
* **callback:** It is a function that would be called when the method is executed.
  + **err:** It is an error that would be thrown if the method fails.

Below examples illustrate the **fs.appendFile() method** in Node.js:

**Example 1:** This example shows the appending of the given text to a file.

* Javascript

|  |
| --- |
| // Node.js program to demonstrate the  // fs.appendFile() method    // Import the filesystem module  const fs = require('fs');    // Get the file contents before the append operation  console.log("\nFile Contents of file before append:",      fs.readFileSync("example\_file.txt", "utf8"));    fs.appendFile("example\_file.txt", "World", (err) => {      if (err) {          console.log(err);      }      else {          // Get the file contents after the append operation          console.log("\nFile Contents of file after append:",              fs.readFileSync("example\_file.txt", "utf8"));      }  }); |

**Output:**

File Contents of file before append: Hello  
File Contents of file after append: HelloWorld

**Example 2:** This example shows the usage of the optional parameters to change the file encoding, mode, and flag of the operation.

* Javascript

|  |
| --- |
| // Node.js program to demonstrate the  // fs.appendFile() method    // Import the filesystem module  const fs = require('fs');    // Get the file contents before the append operation  console.log("\nFile Contents of file before append:",      fs.readFileSync("example\_file.txt", "utf8"));    fs.appendFile("example\_file.txt", " - GeeksForGeeks",      { encoding: "latin1", mode: 0o666, flag: "a" },      (err) => {          if (err) {              console.log(err);          }          else {              // Get the file contents after the append operation              console.log("\nFile Contents of file after append:",                  fs.readFileSync("example\_file.txt", "utf8"));          }      }); |

**Output:**

File Contents of file before append: This is a test file  
File Contents of file after append: This is a test file - GeeksForGeek

We have a Cheat Sheet on Node fs modules where we covered all the fs methods to check those please go through [**Node File System Complete Reference**](https://www.geeksforgeeks.org/node-js-file-system-complete-reference/) article

[*Now get an additional 30% off on all GfG courses of your choice. Also get 90% Course fee refund in just 90 days. Dual savings offer ending soon, avail today!*](https://www.geeksforgeeks.org/courses?utm_source=geeksforgeeks&utm_medium=bottomarticle&utm_campaign=three90)

“This course was packed with amazing and well-organized content! The project-based approach of this course made it even better to understand concepts faster. Also the instructor in the live classes is really good and knowledgeable.”- **Tejas | Deutsche Bank**  
  
With our revamped[Full Stack Development Program](https://www.geeksforgeeks.org/courses/full-stack-node?utm_source=geeksforgeeks&utm_medium=article_bottom_text_fullstacktags&utm_campaign=courses): **master Node.js and React that enables you to create dynamic web applications**.

So get ready for salary hike only with our [Full Stack Development Course](https://www.geeksforgeeks.org/courses/full-stack-node?utm_source=geeksforgeeks&utm_medium=article_bottom_text_fullstacktags&utm_campaign=courses).

* **path:** It is a String, Buffer, URL or number that denotes the source filename or file descriptor that will be appended to.
* **data:** It is a String or Buffer that denotes the data that has to be appended.
* **options:** It is an string or an object that can be used to specify optional parameters that will affect the output. It has three optional parameters:
  + **encoding:** It is a string which specifies the encoding of the file. The default value is ‘utf8’.
  + **mode:** It is an integer which specifies the file mode. The default value is ‘0o666’.
  + **flag:** It is a string which specifies the flag used while appending to the file. The default value is ‘a’.
* **callback:** It is a function that would be called when the method is executed.
  + **err:** It is an error that would be thrown if the method fails.

Below examples illustrate the **fs.appendFile() method** in Node.js:

**Example 1:** This example shows the appending of the given text to a file.

* Javascript

|  |
| --- |
| // Node.js program to demonstrate the  // fs.appendFile() method    // Import the filesystem module  const fs = require('fs');    // Get the file contents before the append operation  console.log("\nFile Contents of file before append:",      fs.readFileSync("example\_file.txt", "utf8"));    fs.appendFile("example\_file.txt", "World", (err) => {      if (err) {          console.log(err);      }      else {          // Get the file contents after the append operation          console.log("\nFile Contents of file after append:",              fs.readFileSync("example\_file.txt", "utf8"));      }  }); |

**Output:**

File Contents of file before append: Hello  
File Contents of file after append: HelloWorld

**Example 2:** This example shows the usage of the optional parameters to change the file encoding, mode, and flag of the operation.

* Javascript

|  |
| --- |
| // Node.js program to demonstrate the  // fs.appendFile() method    // Import the filesystem module  const fs = require('fs');    // Get the file contents before the append operation  console.log("\nFile Contents of file before append:",      fs.readFileSync("example\_file.txt", "utf8"));    fs.appendFile("example\_file.txt", " - GeeksForGeeks",      { encoding: "latin1", mode: 0o666, flag: "a" },      (err) => {          if (err) {              console.log(err);          }          else {              // Get the file contents after the append operation              console.log("\nFile Contents of file after append:",                  fs.readFileSync("example\_file.txt", "utf8"));          }      }); |

**Output:**

File Contents of file before append: This is a test file  
File Contents of file after append: This is a test file - GeeksForGeek

We have a Cheat Sheet on Node fs modules where we covered all the fs methods to check those please go through [**Node File System Complete Reference**](https://www.geeksforgeeks.org/node-js-file-system-complete-reference/) article

[*Now get an additional 30% off on all GfG courses of your choice. Also get 90% Course fee refund in just 90 days. Dual savings offer ending soon, avail today!*](https://www.geeksforgeeks.org/courses?utm_source=geeksforgeeks&utm_medium=bottomarticle&utm_campaign=three90)

“This course was packed with amazing and well-organized content! The project-based approach of this course made it even better to understand concepts faster. Also the instructor in the live classes is really good and knowledgeable.”- **Tejas | Deutsche Bank**  
  
With our revamped[Full Stack Development Program](https://www.geeksforgeeks.org/courses/full-stack-node?utm_source=geeksforgeeks&utm_medium=article_bottom_text_fullstacktags&utm_campaign=courses): **master Node.js and React that enables you to create dynamic web applications**.

So get ready for salary hike only with our [Full Stack Development Course](https://www.geeksforgeeks.org/courses/full-stack-node?utm_source=geeksforgeeks&utm_medium=article_bottom_text_fullstacktags&utm_campaign=courses).