

Background processing using web workers



Contents

Adding a web worker

[Web workers](#) allow you to run CPU-intensive computations in a background thread, freeing the main thread to update the user interface. If you find your application performs a lot of computations, such as generating CAD drawings or doing heavy geometrical calculations, using web workers can help increase your application's performance.

The CLI does not support running Angular itself in a web worker.

Adding a web worker

To add a web worker to an existing project, use the Angular CLI `ng generate` command.

```
ng generate web-worker <location>
```



You can add a web worker anywhere in your application. For example, to add a web worker to the root component, `src/app/app.component.ts`, run the following command.

```
ng generate web-worker app
```



The command performs the following actions.

- Configures your project to use web workers, if it isn't already.
- Adds the following scaffold code to `src/app/app.worker.ts` to receive messages.

`src/app/app.worker.ts`

```
addEventListener('message', ({ data }) => {  
  const response = `worker response to ${data}`;  
  postMessage(response);  
});
```



- Adds the following scaffold code to `src/app/app.component.ts` to use the worker.



```
if (typeof Worker !== 'undefined') {  
  // Create a new  
  const worker = new Worker('./app.worker', { type: 'module' });  
  worker.onmessage = ({ data }) => {  
    console.log(`page got message: ${data}`);  
  };  
  worker.postMessage('hello');  
} else {  
  // Web workers are not supported in this environment.  
  // You should add a fallback so that your program still executes correctly.  
}
```

After you generate this initial scaffold, you must refactor your code to use the web worker by sending messages to and from the worker.

Some environments or platforms, such as @angular/platform-server used in [Server-side Rendering](#), don't support web workers. To ensure that your application will work in these environments, you must provide a fallback mechanism to perform the computations that the worker would otherwise perform.

RESOURCES

About
Resource Listing
Press Kit
Blog
Usage Analytics

HELP

Stack Overflow
Gitter
Report Issues
Code of Conduct

COMMUNITY

Events
Meetups
Twitter
GitHub
Contribute

LANGUAGES

简体中文版
正體中文版
日本語版
한국어

Super-powered by Google ©2010-2020. Code licensed under an MIT-style License. Documentation licensed under CC BY 4.0.

Version 10.0.10-local+sha.84d1ba792b.