

DualOptions

August 25, 2026

DualOptions

Kind: Class

Source: `lib/option.js`

This class is to make it easier to work with dual options, without changing the existing implementation. We support separate dual options for separate positive and negative options,

like `--build` and `--no-build`, which share a single option value. This works nicely for some

use cases, but is tricky for others where we want separate behaviours despite the single shared option value.

`DualOptions` manages paired positive and negative CLI options that share the same option attribute, such as `--build` and `--no-build`. It determines the value represented by the option that was supplied, allowing the parser to preserve distinct behavior for each side of a dual option.

Methods

Method	Signature	Returns
<code>valueFromOption</code>	<code>valueFromOption(value: undefined, option: undefined)</code>	<code>void</code>

Where it refuses work

- `DualOptions` stops the work with an early return when `!this.dualOptions.has(optionKey)`.

Diagram

```
graph LR
  A[CLI input] --> B[Option]
```

```
B --> C[DualOptions]
C --> D[valueFromOption]
D --> E[Shared option value]
```

Usage

```
import { Command } from 'commander';

const program = new Command();

program
  .option('--build', 'enable build output')
  .option('--no-build', 'disable build output');

program.parse();

const options = program.opts();

if (options.build) {
  console.log('Build output is enabled.');
```

```
} else {
  console.log('Build output is disabled.');
```

```
}
```

AI Coding Instructions

- Treat `DualOptions` as parser support code for positive and negative options that map to the same attribute.
- Keep paired option names aligned, such as `--build` and `--no-build`, so they resolve to the same option value.
- Use `valueFromOption()` when converting a parsed option occurrence into its stored value.
- Do not assume a negative option always means the parsed value is `false`; preserve the option-specific behavior handled by this class.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (1)

- `Command` — `lib/command.js :14`

