# tslib

This is a runtime library for [TypeScript](https://www.typescriptlang.org/) that contains all of the TypeScript helper functions.

This library is primarily used by the --importHelpers flag in TypeScript. When using --importHelpers, a module that uses helper functions like \_\_extends and \_\_assign in the following emitted file:

var \_\_assign = (this && this.\_\_assign) || Object.assign || function(t) {

for (var s, i = 1, n = arguments.length; i < n; i++) {

s = arguments[i];

for (var p in s) if (Object.prototype.hasOwnProperty.call(s, p))

t[p] = s[p];

}

return t;

};

exports.x = {};

exports.y = \_\_assign({}, exports.x);

will instead be emitted as something like the following:

var tslib\_1 = require("tslib");

exports.x = {};

exports.y = tslib\_1.\_\_assign({}, exports.x);

Because this can avoid duplicate declarations of things like \_\_extends, \_\_assign, etc., this means delivering users smaller files on average, as well as less runtime overhead. For optimized bundles with TypeScript, you should absolutely consider using tslib and --importHelpers.

# Installing

For the latest stable version, run:

## npm

# TypeScript 3.9.2 or later

npm install tslib

# TypeScript 3.8.4 or earlier

npm install tslib@^1

# TypeScript 2.3.2 or earlier

npm install tslib@1.6.1

## yarn

# TypeScript 3.9.2 or later

yarn add tslib

# TypeScript 3.8.4 or earlier

yarn add tslib@^1

# TypeScript 2.3.2 or earlier

yarn add tslib@1.6.1

## bower

# TypeScript 3.9.2 or later

bower install tslib

# TypeScript 3.8.4 or earlier

bower install tslib@^1

# TypeScript 2.3.2 or earlier

bower install tslib@1.6.1

## JSPM

# TypeScript 3.9.2 or later

jspm install tslib

# TypeScript 3.8.4 or earlier

jspm install tslib@^1

# TypeScript 2.3.2 or earlier

jspm install tslib@1.6.1

# Usage

Set the importHelpers compiler option on the command line:

tsc --importHelpers file.ts

or in your tsconfig.json:

{

"compilerOptions": {

"importHelpers": true

}

}

#### For bower and JSPM users

You will need to add a paths mapping for tslib, e.g. For Bower users:

{

"compilerOptions": {

"module": "amd",

"importHelpers": true,

"baseUrl": "./",

"paths": {

"tslib" : ["bower\_components/tslib/tslib.d.ts"]

}

}

}

For JSPM users:

{

"compilerOptions": {

"module": "system",

"importHelpers": true,

"baseUrl": "./",

"paths": {

"tslib" : ["jspm\_packages/npm/tslib@2.x.y/tslib.d.ts"]

}

}

}

## Deployment

* Choose your new version number
* Set it in package.json and bower.json
* Create a tag: git tag [version]
* Push the tag: git push --tags
* Create a [release in GitHub](https://github.com/microsoft/tslib/releases)
* Run the [publish to npm](https://github.com/microsoft/tslib/actions?query=workflow%3A%22Publish+to+NPM%22) workflow

Done.

# Contribute

There are many ways to [contribute](https://github.com/Microsoft/TypeScript/blob/master/CONTRIBUTING.md) to TypeScript.

* [Submit bugs](https://github.com/Microsoft/TypeScript/issues) and help us verify fixes as they are checked in.
* Review the [source code changes](https://github.com/Microsoft/TypeScript/pulls).
* Engage with other TypeScript users and developers on [StackOverflow](http://stackoverflow.com/questions/tagged/typescript).
* Join the [#typescript](http://twitter.com/#!/search/realtime/%23typescript) discussion on Twitter.
* [Contribute bug fixes](https://github.com/Microsoft/TypeScript/blob/master/CONTRIBUTING.md).

# Documentation

* [Quick tutorial](http://www.typescriptlang.org/Tutorial)
* [Programming handbook](http://www.typescriptlang.org/Handbook)
* [Homepage](http://www.typescriptlang.org/)