DefinitelyTyped and @types

AUTHOR:  Todd Motto Follow @toddmotto

Join us on Slack!

DefinitelyTyped and @types

DefinitelyTyped is a must have resource for any TypeScript developer. It's essentially documentation (well, *.d.ts files) for most JavaScript packages available out there in the open source community.
This means when you next install a third-party package, you'll also want to install the type declaration files as well. Doing so will open up typed-everything (autocompletion, spell checks, IDE support and so on).
From jQuery and lodash through to front-end frameworks, you can be sure have the full power of TypeScript behind them.
It's also a community-driven project, should ever need to send a PR (Pull Request) to add any new typings!

Installing `@types`

Installation is via npm/yarn. An example using lodash:

```
npm install @types/lodash --save
# OR
yarn add @types/lodash
```

`@types` supports both global and module type definitions.

`@types` and Modules

Any `@types` files will automatically be added to the global scope, so you can use them instantly in `*.ts` files without any error. However, the recommended way is to treat new `@types` installations as module includes:

```
import * as lodash from 'lodash';
// Use "lodash"
```

"types" Compiler Option

We can be more explicit when it comes to providing type files by using the `types` property. This restricts the types that TypeScript will lookup when type-checking, which means only types listed here (and that are installed) will be available and any others ignored. This may be a preferable opt-in to avoid any global variables leaking.

```
{ "compilerOptions": { "types" : [ "lodash", "express" ] } }
```