EMBER OCTANE: BUILD TOOLING CHANGES

IMPROVEMENT: IMPORTING THIRD PARTY MODULES

BEFORE: BROCCOLI AND EMBER-ADDONS

https://github.com/mike-north/ember-lodash

https://simplabs.com/blog/2017/02/13/npm-libs-in-ember-cli.html

AFTER: EMBER-AUTO-IMPORT

https://github.com/ef4/ember-auto-import

Some advantages:

- Defer loading large libraries until they are needed
- Remove large wrapper libraries

Before:

200	GET	🔒 mwc-ember.ololocal.net	📄 vendor-ee46caea90fb8411edce7b4abbeaaac5.js	HTTP/1.1	js	337.30 KB	1.33 MB
200	GET	🔒 mwc-ember.ololocal.net	📄 mobile-web-71a0f0d8f9b1af534f72c2956aeede4f.js	HTTP/1.1	js	231.87 KB	1.27 MB

After:

200	GET	🔒 mwc-ember.ololocal.net	📄 vendor-aff11f819932c7f3ff6dd4813a5d29ce.js	HTTP/1.1	js	283.83 KB	1.10 MB
200	GET	🔒 mwc-ember.ololocal.net	📄 mobile-web-fbda8f2c9e5c4eb2d2b7da93050fe9ef.js	HTTP/1.1	js	232.68 KB	1.28 MB

IMPROVEMENT? USING TYPESCRIPT ANNOTATIONS FOR EMBER CLASSES

BABEL? TYPESCRIPT?

BABEL

Takes future JS language features and turns it into current JS features

TYPESCRIPT
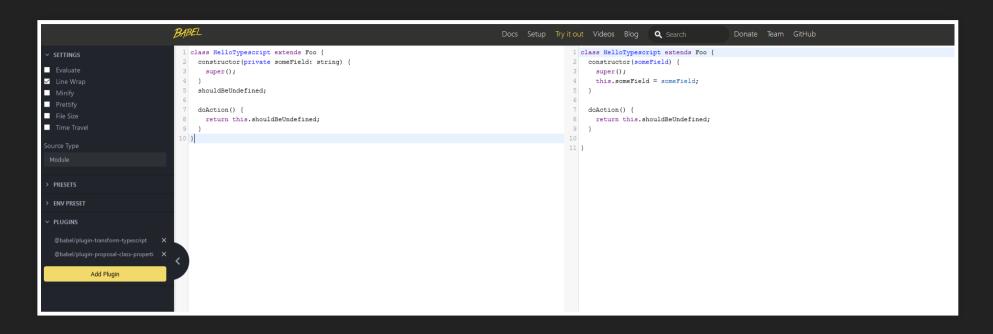
Can also emit current JS while providing type checking.

Supports a smaller set of transforms

TS playground https://bit.ly/2TUuljk

Babel playground https://bit.ly/2TRT2gC

```typescript
export default class AccountOverview extends Component.extend(AccountInfoMixin) {
    // arguments
    model!: Model;
    unlinkMembership!: (membershipId: number) => void;
    loginProviderName?: string;
```

```
element: unlink
@@ -1333,6 +1335,9 @@
    let AccountOverview = (_dec = (0, _object.computed)('loginProviderName'), _dec2 = (0, _obj
      constructor(...args) {
        super(...args);
+       this.model = void 0;
+       this.unlinkMembership = void 0;
+       this.loginProviderName = void 0;
        this.updateInfoDesc = 'Update Information';

        initializerDefineProperty(this, "tagClass", _descriptor, this);
```

BABEL

SETTINGS

☐ Evaluate
☑ Line Wrap
☐ Minify
☐ Prettify
☐ File Size
☐ Time Travel

Source Type

Module

PRESETS

ENV PRESET

PLUGINS

@babel/plugin-transform-typescript   ✕
@babel/plugin-proposal-class-properti   ✕

Add Plugin

```
1  class HelloTypescript extends Foo {
2    constructor(private someField: string) {
3      super();
4    }
5    shouldBeUndefined;
6
7    doAction() {
8      return this.shouldBeUndefined;
9    }
10 }
```

```
1  class HelloTypescript extends Foo {
2    constructor(someField) {
3      super();
4      this.someField = someField;
5    }
6
7    doAction() {
8      return this.shouldBeUndefined;
9    }
10
11 }
```

BABEL

SETTINGS

☐ Evaluate
☑ Line Wrap
☐ Minify
☐ Prettify
☐ File Size
☐ Time Travel

Source Type

Module

PRESETS

ENV PRESET

PLUGINS

@babel/plugin-proposal-class-properti   ✕
@babel/plugin-transform-typescript   ✕

Add Plugin

```
1  class HelloTypescript extends Foo {
2    constructor(private someField: string) {
3      super();
4    }
5    shouldBeUndefined;
6
7    doAction() {
8      return this.shouldBeUndefined;
9    }
10 }
```

```
1  function _defineProperty(obj, key, value) { if (key in obj) { Object.defineProperty(obj, key, {
2  value: value, enumerable: true, configurable: true, writable: true }); } else { obj[key] = value;
3  } return obj; }
3  class HelloTypescript extends Foo {
4    constructor(someField) {
5      super();
6      this.someField = someField;
7
8      _defineProperty(this, "shouldBeUndefined", void 0);
9    }
10
11   doAction() {
12     return this.shouldBeUndefined;
13   }
14
15 }
```

```javascript
const app = new EmberApp(defaults, {
  hinting: IS_TEST,
  tests: IS_TEST,
  babel: {
    sourceMaps: IS_PROD,
    plugins: ['@babel/plugin-transform-typescript'],
  },
  'ember-qunit': { useLintTree: false }, // Disable JSHint; we're usin
  'ember-cli-babel': {
    sourceMaps: IS_PROD,
  }
```

QUESTIONS?