4-1至3-什么是loader

```
import layer from './components/layer/layer.js';

import layer from './components/layer/layer.js';

const App = function () {

console.log(layer)

layer.html
layer.js
layer.less

new App()

import layer from './components/layer/layer.js';

const App = function () {

console.log(layer)

layer.html

app.js

index.html

app.js

index.html

app.js

index.html

app.js

a
```

layer. less

```
> in dist
                          .layer {-
> im node_modules
                          width: 600px;
🕶 🛅 src
                          height: 200px;

▼ m components

   🕶 🛅 layer
     layer.html
                   5 ·> div {-
                          width: 400px;
     layer.less
                          height: 100px;
                          background-color: red;
 ackage.ison
 webpack.config.js
                          }
```

layer. js

```
× layer.less
                                                        x layer.html
> iii dist
                          import tpl from './layer.html';
> im node_modules
🕶 🛅 src
 v 🖿 components
                         function layer () {-
                            return {
   layer.html
                             name: 'layer',
      layer.js
                            tpl: tpl
     ☐ layer.less
 package.json
 webpack.config.js
                           export default layer;
```

 ${\tt config}$

```
> dist
                         var htmlWebpackPlugin = require('html-webpack-plugin');
src
                         module.exports = {

▼ im components

  v 🛅 laye
                          entry: './src/app.js',
                          output: {
     layer.js
                           path: './dist',
     ayer.less
                           filename: 'js/[name].bundle.js'
  app.js
 package.json
                          plugins: [
 webpack.config.js
                           new htmlWebpackPlugin({
                             filename: 'index.html',
                             template: 'index.html',
                              inject: 'body'
```

loader 是为了解决Webpact不支持的格式。 说明文档如下:

https://webpack.js.org/configuration/module/ 不过视频中采用的是laders被弃用了, 需用rules.

```
module: {

loaders: [-

test: /\.js$/,-

loader: 'babel',-

exclude: './node_modules/',-

query: {

presets: ['latest'] [

}
```

js插件转换 babeljs相当 通过这个插件可把ES6可解析生成 ES5语法

https://babeljs.io/ https://babeljs.io/docs/setup/#installation

npm install --save-dev babel-loader babel-core

如果指定query中的preset:['latest'], 需安装latest这插件,现在被env代替了

https://babeljs.io/docs/plugins/

也可以用下面这种 方式指定

参考vue打包, 注意include需要用绝对路径

视频中的loaders被rules代替,其它感觉没啥区别