

8-I18N

Internacionalización y puesta en producción

- # 1. Traducciones y contenido
 - # 2. Adaptaciones culturales de tiempo y moneda
-

1 Traducciones y contenido

1.1 xi18n

apps\warehouse\src\app\app.component.html

```
<header>
  <h1 i18n>Welcome to the Angular Builders Warehouse</h1>
</header>

<router-outlet></router-outlet>
<footer>
  <a href="https://angular.builders"
      target="blank">Angular.Builders: </a>
  <span i18n>a store of resources for developers and software architects.</span>
</footer>
```

apps\warehouse\tsconfig.app.json

```
{
  "angularCompilerOptions": {
    "enableIvy": false
  }
}
```

package.json

```
{
  "i18n:warehouse": "ng xi18n warehouse --output-path src/locale",
}
```

1.2 Build configurations

angular.json

```
"production-es": {
  "fileReplacements": [
    {
      "replace": "apps/warehouse/src/environments/environment.ts",
      "with": "apps/warehouse/src/environments/environment.prod.es.ts"
    },
  ],
  "outputPath": "dist/apps/warehouse/es/",
  "i18nFile": "apps/warehouse/src/locale/messages.es.xlf",
  "i18nFormat": "xlf",
  "i18nLocale": "es",
  "baseHref": "es",
}
```

```
{
  "build:warehouse-es": "ng build warehouse --configuration=production-es",
  "start:warehouse-es": "npm run build:warehouse-es && angular-http-server --open -p 8082 --path ./dist/apps/warehouse/es",
}
```

2 Adaptaciones culturales de tiempo y moneda

2.1 Registro manual en app.module o Auto registro en angular.json

Manual

```
import { registerLocaleData } from '@angular/common';
import localeEs from '@angular/common/locales/es';

registerLocaleData(localeEs);
```

Automático

```
"i18nLocale": "es"
```

2.2 Tiempo, moneda y contenido

apps\warehouse\src\app\app.component.html

```
<article class="card">
  <p>{{ building.date | date:'long' }}</p>
  <p>${{ building.value | number }}<i> {{ building.status }}</i></p>
</article>
```

apps\warehouse\src\app\app.component.ts

```
public building = {
  date: Date.now(),
  value: 2345.897,
  status: 'buy'
};
constructor() {
  if (this.building.status === 'buy') {
    this.building.status = environment.buy;
  } else {
    this.building.status = environment.sell;
  }
}
```

```
{
  buy: 'for buy',
  sell: 'for sell'
}
{
  buy: 'para comprar',
  sell: 'para vender'
}
```

Blog de apoyo: [Internacionalización y puesta en producción](#)

By [Alberto Basalo](#)

Next:

Web Components

Desarrollo en Angular

Consumo en cualquier framework