

API Report File for “@angular/elements”

Do not edit this file. It is a report generated by API Extractor.

```
import { Injector } from '@angular/core';
import { Observable } from 'rxjs';
import { Subscription } from 'rxjs';
import { Type } from '@angular/core';
import { Version } from '@angular/core';

// @public
export function createCustomElement<P>(component: Type<any>, config: NgElementConfig): NgElement {

// @public
export abstract class NgElement extends HTMLElement {
    abstract attributeChangedCallback(attrName: string, oldValue: string | null, newValue: string): void;
    abstract connectedCallback(): void;
    abstract disconnectedCallback(): void;
    protected ngElementEventsSubscription: Subscription | null;
    protected abstract ngElementStrategy: NgElementStrategy;
}

// @public
export interface NgElementConfig {
    injector: Injector;
    strategyFactory?: NgElementStrategyFactory;
}

// @public
export interface NgElementConstructor<P> {
    new (injector?: Injector): NgElement & WithProperties<P>;
    readonly observedAttributes: string[];
}

// @public
export interface NgElementStrategy {
    // (undocumented)
    connect(element: HTMLElement): void;
    // (undocumented)
    disconnect(): void;
    // (undocumented)
    events: Observable<NgElementStrategyEvent>;
    // (undocumented)
    getInputValue(propName: string): any;
    // (undocumented)
}
```

```

    setInputValue(propName: string, value: string): void;
}

// @public
export interface NgElementStrategyEvent {
    // (undocumented)
    name: string;
    // (undocumented)
    value: any;
}

// @public
export interface NgElementStrategyFactory {
    create(injector: Injector): NgElementStrategy;
}

// @public (undocumented)
export const VERSION: Version;

// @public
export type WithProperties<P> = {
    [property in keyof P]: P[property];
};

// (No @packageDocumentation comment for this package)

```