

Defiant v1.0.0 cheatsheet - <https://github.com/Serrin/Defiant>

Import	Javascript
import the defaultExport object	<code>import defaultExport from "./defiant.js"; globalThis.defiant = defaultExport;</code>
import with default with name	<code>import { default as defiant } from "./defiant.js"; globalThis.defiant = defiant;</code>
import all into a new defiant object	<code>import * as defiant from "./defiant.js"; globalThis.defiant = defiant;</code>
import functions	<code>import { Tuple, Record } from "./defiant.mjs"; globalThis.Tuple = Tuple; globalThis.Record = Record;</code>

Function	Description
<code>Tuple(...values);</code>	Creates a frozen array (tuple) from the given values.
<code>Tuple.of(...values);</code>	Alias for <code>Tuple(...)</code> .
<code>Tuple.from(iterable);</code>	Creates a tuple from an iterable or array-like object.
<code>Tuple.isTuple(value);</code>	Checks whether a value is a frozen array.
<code>Tuple.isEqual(x, y);</code>	Performs a deep equality check between two values (Arrays and Objects).
<code>Tuple.toArray(value);</code>	Converts a tuple to a mutable array (shallow copy).
<code>Record(object);</code>	Creates a frozen copy of an object.
<code>Record.fromEntries(entries);</code>	Creates a record from an iterable of key-value pairs.
<code>Record.fromObject(object);</code>	Alias for <code>Record(object)</code> .
<code>Record.isRecord(value);</code>	Checks whether a value is a frozen object.
<code>Record.isEqual(x, y);</code>	Alias for <code>Tuple.isEqual</code> .
<code>Record.toObject(value);</code>	Converts a record to a mutable object (shallow copy).