

[Hummingbird](#) / Environment

## Structure

# Environment

Access environment variables

```
struct Environment
```

## Topics

### Structures

```
struct Error
```

### Initializers

```
init()
```

Initialize from environment variables

```
init(dictionaryLiteral: (String, String)...) 
```

Initialize from dictionary literal

```
init(from: Decoder) throws
```

Initialize from Decodable

```
init(values: [String : String])
```

Initialize from dictionary

## Instance Methods

```
func get(String) -> String?
```

Get environment variable with name

```
func get<T>(String, as: T.Type) -> T?
```

Get environment variable with name as a certain type

```
func merging(with: Environment) -> Environment
```

Merge two environment variable sets together and return result

```
func require(String) throws -> String
```

Require environment variable with name

```
func require<T>(String, as: T.Type) throws -> T
```

Require environment variable with name as a certain type

```
func set(String, value: String?)
```

Set environment variable

## Type Methods

```
static func dotEnv(String) async throws -> Environment
```

Create Environment initialised from the `.env` file

## Default Implementations

☰ CustomStringConvertible Implementations

---

## Relationships

## Conforms To

Swift.Copyable  
Swift.CustomStringConvertible  
Swift.Decodable  
Swift.ExpressibleByDictionaryLiteral  
Swift.Sendable

---

## See Also

### Miscellaneous

`protocol InitializableFromSource`

---

A type that can be initialized from another type