

## Project Overview: Cron Expression Parser

### Overview

The Cron Expression Parser is a JavaScript project designed to parse and interpret cron expressions, providing functionality to expand cron fields and validate input expressions. This project is structured with modularity in mind, making it easy to understand and extend.

### Project Structure

```
cron-expression-parser
|-- src
|   |-- expandCronField.js
|   |-- main.js
|   |-- tableFormatter.js
|   |-- inputValidator.js
|   |-- parseCronExpression.js
|-- test
|   |-- expandCronField.test.js
|   |-- main.test.js
|   |-- inputValidator.test.js
|   |-- parseCronExpression.test.js
|-- package.json
|-- README.md
```

### Source Code

#### 1. `expandCronField.js`

- Responsible for expanding a specific field in a cron expression.

#### 2. `main.js`

- Main entry point for the Cron Expression Parser application.

#### 3. `tableFormatter.js`

- Provides utility functions for formatting data into tables.

#### 4. `inputValidator.js`

- Implements functions to validate the input cron expressions.

#### 5. `parseCronExpression.js`

- Core module for parsing and interpreting cron expressions.

### Test Suite

#### 1. `expandCronField.test.js`

- Test cases for the `expandCronField` module.

#### 2. `main.test.js`

- Test cases for the main application logic.

#### 3. `inputValidator.test.js`

- Test cases for the `inputValidator` module.

#### 4. `parseCronExpression.test.js`

- Test cases for the core `parseCronExpression` module.

### Additional Files

#### 1. `package.json`

- Contains project metadata and dependencies.

#### 2. `Readme`

- Project documentation providing an overview, installation instructions, and usage guidelines.

### Getting Started

#### 1. Installation:

```
npm install
```

#### 2. Run Tests:

```
npm test
```

#### 3. Usage:

- Refer to the documentation in [Readme.md](#) for detailed instructions on using the Cron Expression Parser.

Thank you for using the Cron Expression Parser!