HOMEWORK 2

ASYNC DEVELOPMENT

Tasks

1. Create a directory called `data`. We will assume that CSV files with new products will be uploaded in this directory for processing by our application.
2. In your application create a local module called `dirwatcher`. Create class `DirWatcher` that should be able to watch a given **path** with a given **delay** and emit a ‘`changed`‘ event if directory contents have been changed (implement method `watch(path, delay)` by yourself, try not to use native *fs.watch()*).
    a. When the path is checked for the first time all files should be treated as new.
3. Create a module called `importer`. Create class `Importer`. It should be able to listen to `DirWatcher` events and start importing CSV files (converting the data to JavaScript objects) on ‘`dirwatcher:changed`’ event.
    a. Implement `import(path)`: should return a **promise** with imported data from file at `path`.
    b. Implement `importSync(path)`: should be synchronous and return all imported data from file at `path`.
4. In **app.js**:
    a. Import all of the above modules.
    b. Create a `Dirwatcher` and `Importer` for processing files asynchronously from `data` directory.
    c. Log imported data to console.

Note: every CSV file in a directory should be processed only once.
Note: feel free to use already implemented library for transforming CSV into JSON.

Evaluation criteria

1. CSV files are placed in appropriate directory.
2. `DirWatcher` module is implemented and matches described criteria.
3. `Importer` module is implemented and matches described criteria.
4. Application logic is implemented for a fixed (predefined) number of CSV files.
5. Application logic is implemented for arbitrary number of CSV files (all tasks and subtasks are implemented properly) which could be added/changed/removed at any time.