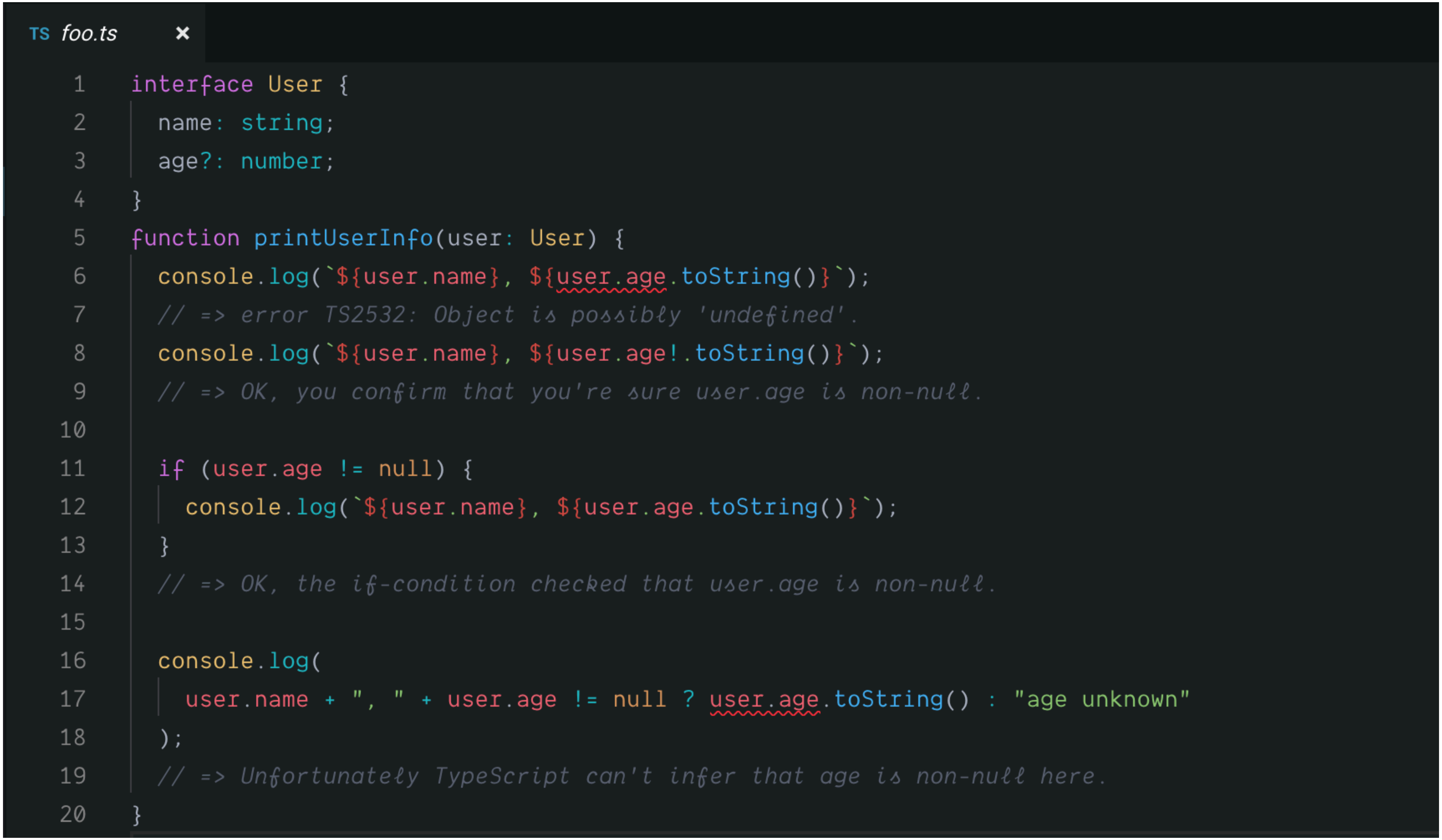
Path to strict type checking

Jakub Chodorowicz
github.com/chodorowicz
jakub@chodorowicz.com
@chodorowicz

Make it --strict

- `--strictNullChecks`
- `--noImplicitAny`
- `--noImplicitReturns`
- `--noImplicitThis`
- `--alwaysStrict`
- `--strictBindCallApply`
- `--strictFunctionTypes`
- `--strictPropertyInitialization`



```typescript
interface User {
  name: string;
  age?: number;
}
function printUserInfo(user: User) {
  console.log(`${user.name}, ${user.age.toString()}`);
  // => error TS2532: Object is possibly 'undefined'.
  console.log(`${user.name}, ${user.age!.toString()}`);
  // => OK, you confirm that you're sure user.age is non-null.

  if (user.age != null) {
    console.log(`${user.name}, ${user.age.toString()}`);
  }
  // => OK, the if-condition checked that user.age is non-null.

  console.log(
    user.name + ", " + user.age != null ? user.age.toString() : "age unknown"
  );
  // => Unfortunately TypeScript can't infer that age is non-null here.
}
```

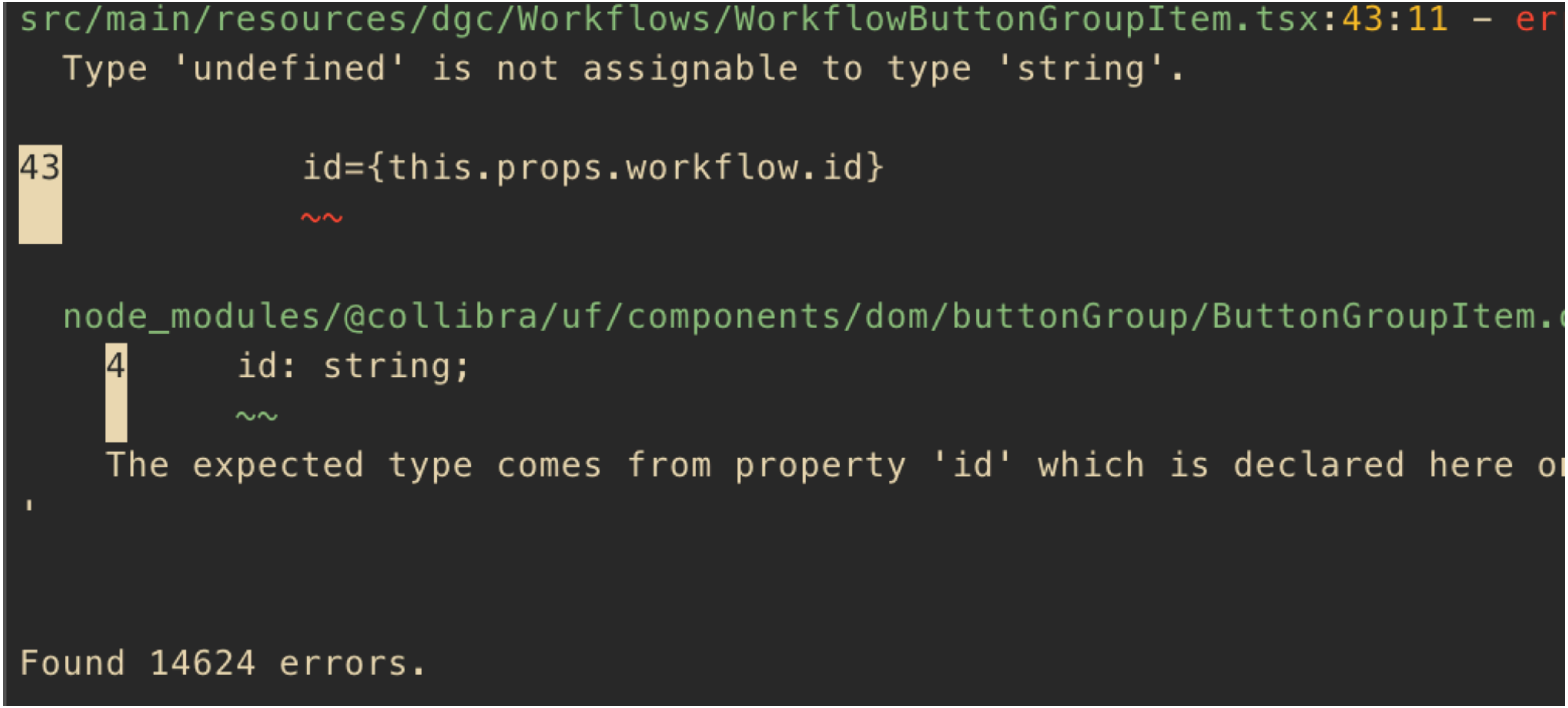
Projects often end up with no strict TS compiler settings

Problem in VS Code

- Thousands of errors when turning on `strictNullChecks`
- See more: https://github.com/Microsoft/vscode/issues/60565

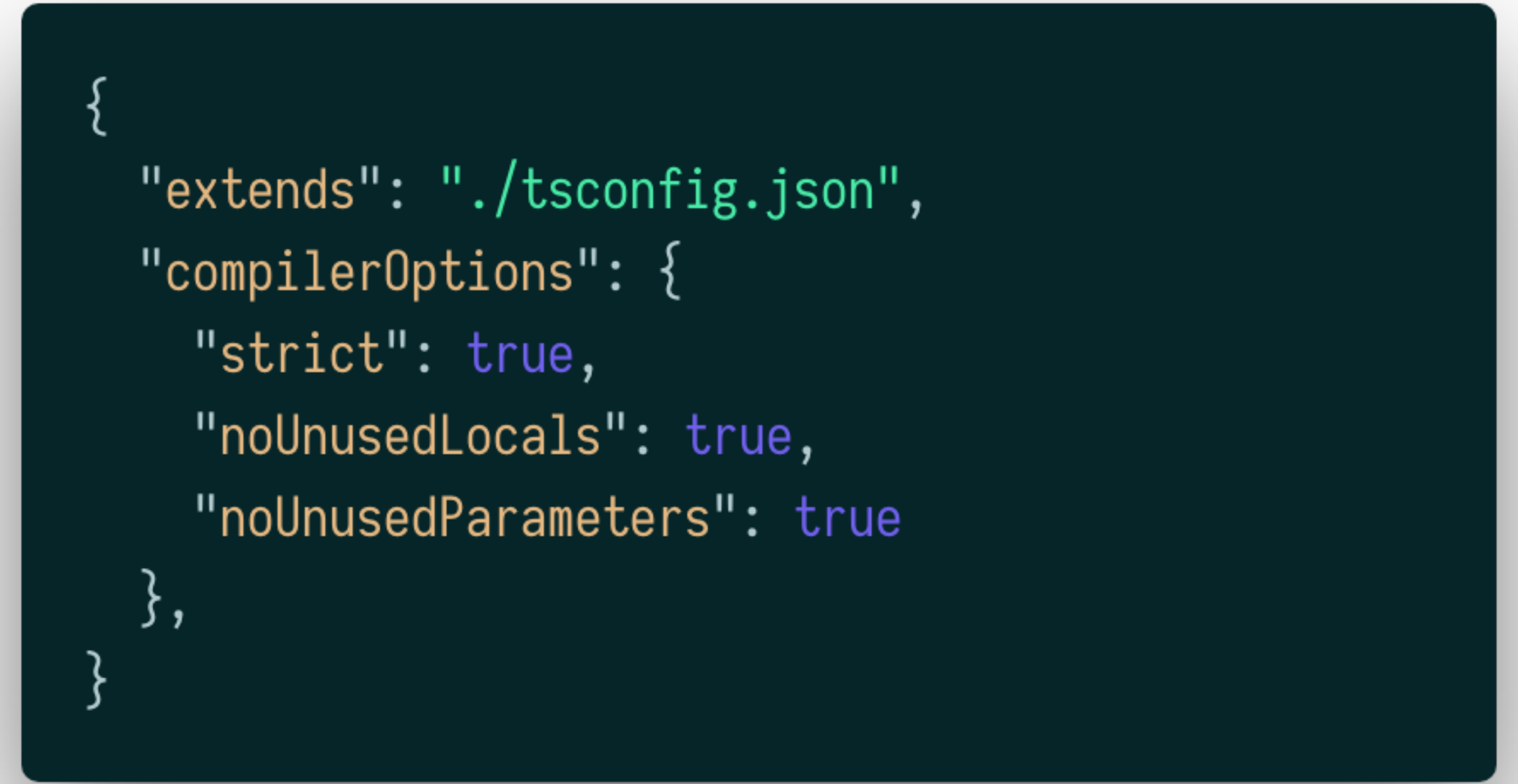
Problem in our codebase

🤕🔫



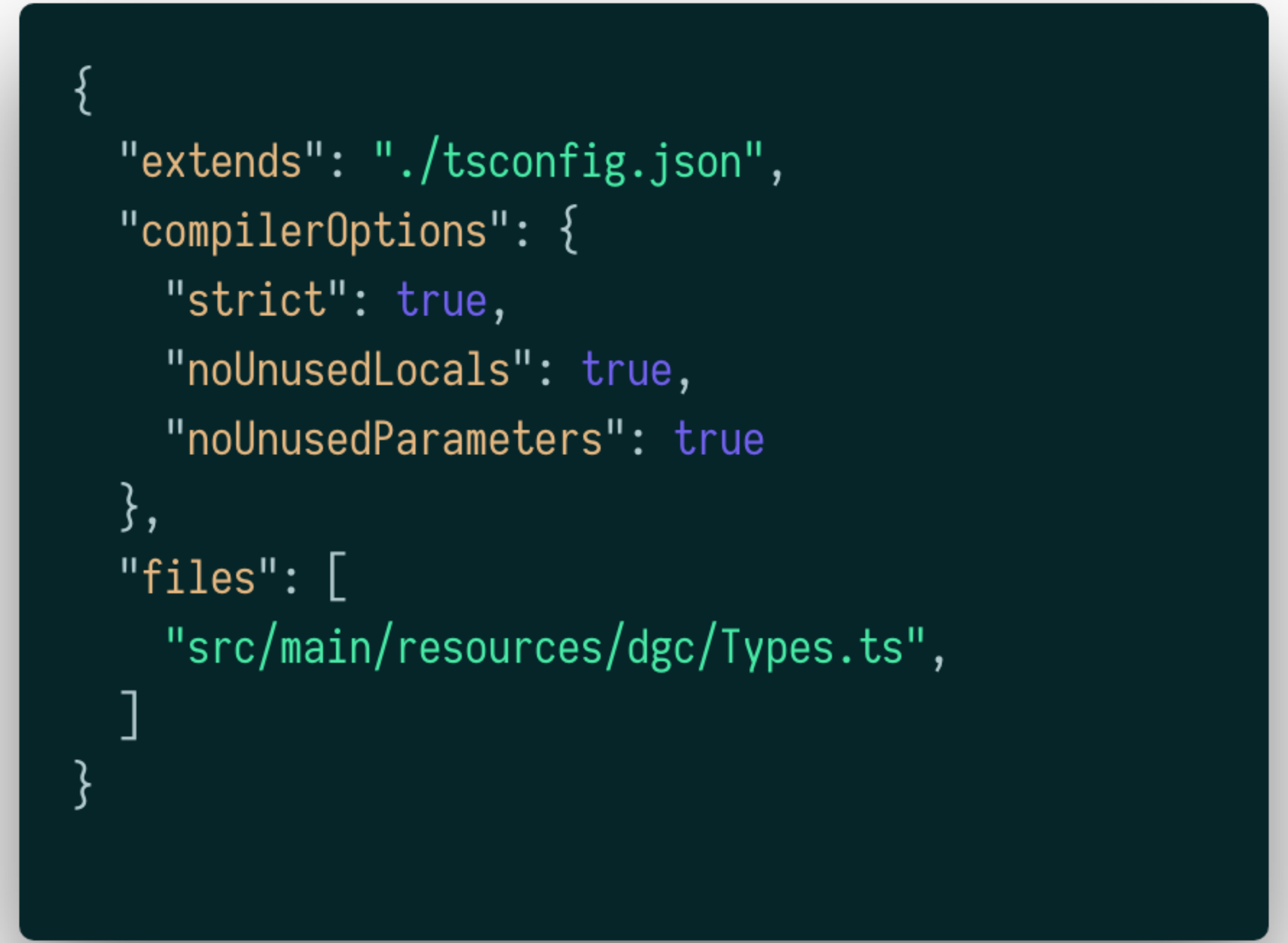
```
src/main/resources/dgc/Workflows/WorkflowButtonGroupItem.tsx:43:11 - er
  Type 'undefined' is not assignable to type 'string'.

43             id={this.props.workflow.id}
               ~~

  node_modules/@collibra/uf/components/dom/buttonGroup/ButtonGroupItem.
    4      id: string;
           ~~
    The expected type comes from property 'id' which is declared here o
'


Found 14624 errors.
```

Create a second TS config extending original



```
{
  "extends": "./tsconfig.json",
  "compilerOptions": {
    "strict": true,
    "noUnusedLocals": true,
    "noUnusedParameters": true
  },
}
```

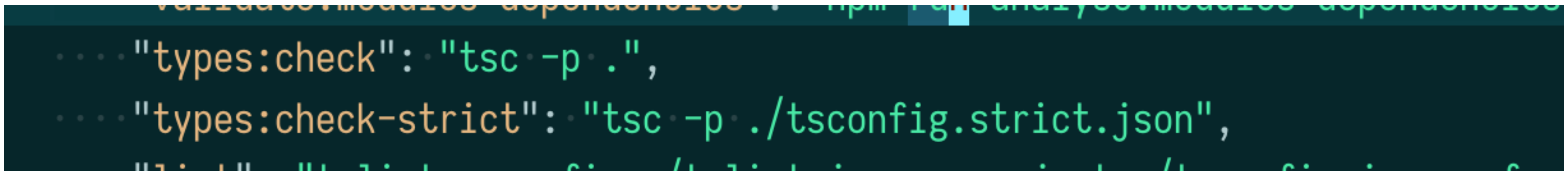
Start adding files one by one



```json
{
  "extends": "./tsconfig.json",
  "compilerOptions": {
    "strict": true,
    "noUnusedLocals": true,
    "noUnusedParameters": true
  },
  "files": [
    "src/main/resources/dgc/Types.ts",
  ]
}
```

Hook it up to some automation

- Use Husky and add it as precommit/prepush script
- Add this as additional step to CI (Jenkins/CircleCI/Travis)



```
    "types:check": "tsc -p .",
    "types:check-strict": "tsc -p ./tsconfig.strict.json",
```

Problem: strict settings are contagious

Some help

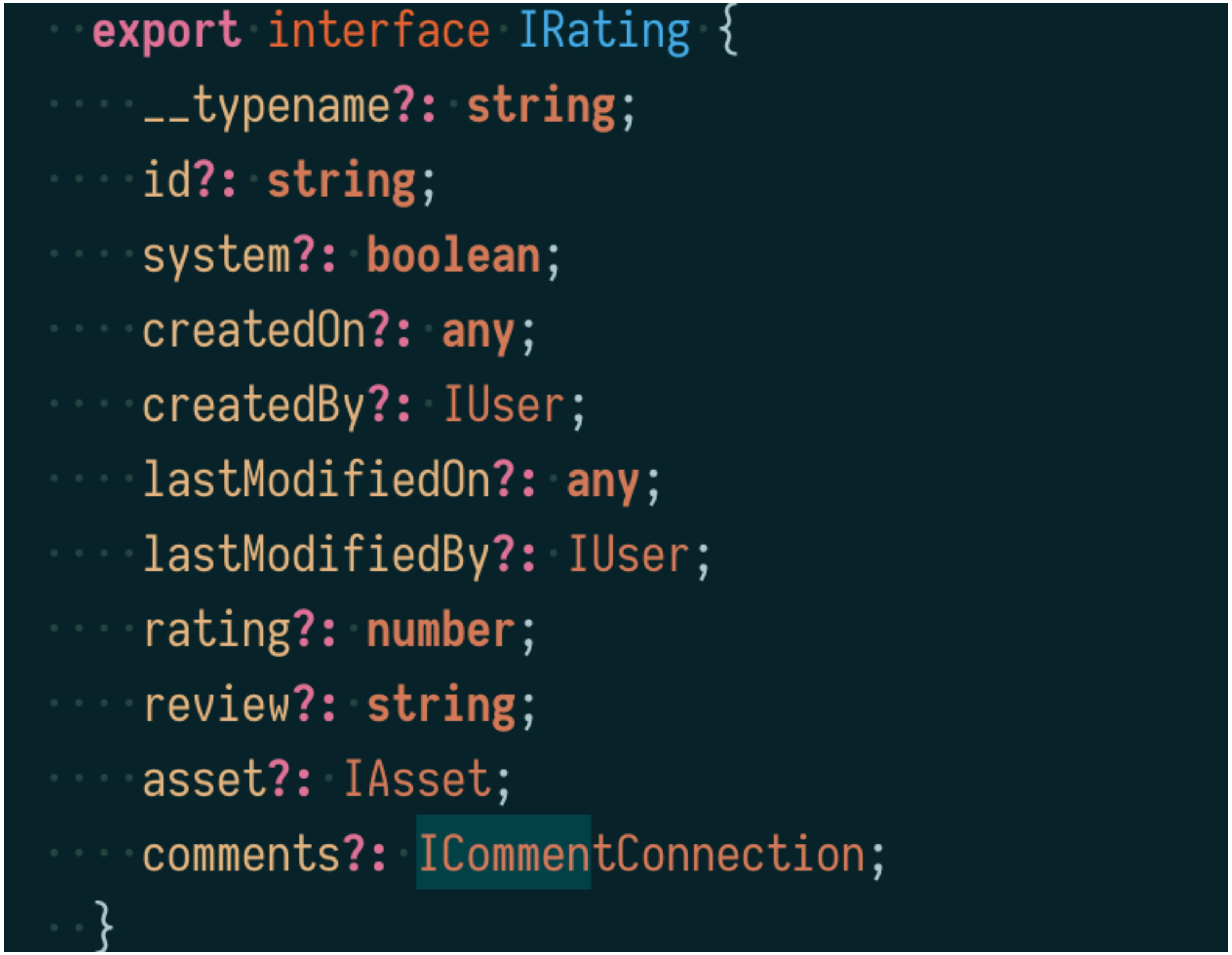
- https://github.com/mjbvz/vscode-strict-null-check-migration-tools

- index.js
  the main script prints of list of files that are eligible for strict null checks. This includes all files that only import files thare are already strict null checked.

- autoAdd.js
  Very simple script that tries to auto add any eligible file to the `tsconfig.strictNullChecks.json`. This iteratively compiles the `tsconfig` project with just that file added. If there are no errors, it is added to the `tsconfig`

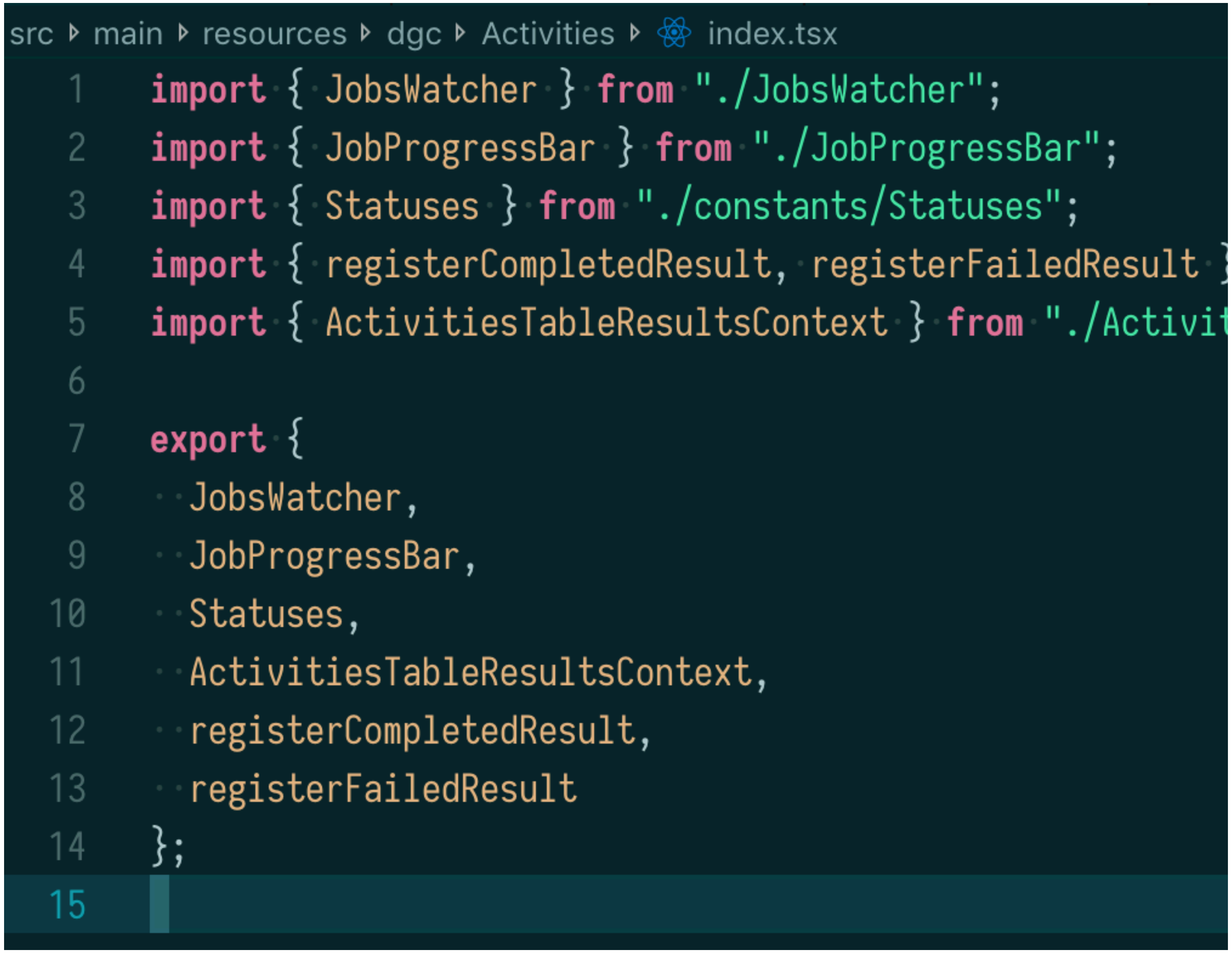
More problems and issues 🤒

Relay Classic GraphQL API types



```
export interface IRating {
    __typename?: string;
    id?: string;
    system?: boolean;
    createdOn?: any;
    createdBy?: IUser;
    lastModifiedOn?: any;
    lastModifiedBy?: IUser;
    rating?: number;
    review?: string;
    asset?: IAsset;
    comments?: ICommentConnection;
}
```

Barrel files



src ▸ main ▸ resources ▸ dgc ▸ Activities ▸ index.tsx

```tsx
import { JobsWatcher } from "./JobsWatcher";
import { JobProgressBar } from "./JobProgressBar";
import { Statuses } from "./constants/Statuses";
import { registerCompletedResult, registerFailedResult }
import { ActivitiesTableResultsContext } from "./Activit

export {
  JobsWatcher,
  JobProgressBar,
  Statuses,
  ActivitiesTableResultsContext,
  registerCompletedResult,
  registerFailedResult
};

```

The end

Jakub Chodorowicz
github.com/chodorowicz
jakub@chodorowicz.com
@chodorowicz