

Git

We prefer a **rebase workflow** and occasional **feature branches**. Most work happens directly on the `main` branch. For that reason, we recommend setting the `pull.rebase` setting to `merges`.

```
git config --global pull.rebase merges
```

Indentation

We use tabs, not spaces.

Names

- Use PascalCase for `type` names
- Use PascalCase for `enum` values
- Use camelCase for `function` and `method` names
- Use camelCase for `property` names and `local variables`
- Use whole words in names when possible

Types

- Do not export `types` or `functions` unless you need to share it across multiple components
- Do not introduce new `types` or `values` to the global namespace

Comments

- Use JSDoc style comments for `functions`, `interfaces`, `enums`, and `classes`

Strings

- Use "double quotes" for strings shown to the user that need to be externalized (localized)
- Use 'single quotes' otherwise
- All strings visible to the user need to be externalized

Style

- Use arrow functions `=>` over anonymous function expressions
- Only surround arrow function parameters when necessary. For example, `(x) => x + x` is wrong but the following are correct:

```
x => x + x
(x, y) => x + y
<T>(x: T, y: T) => x === y
```

- Always surround loop and conditional bodies with curly braces
- Open curly braces always go on the same line as whatever necessitates them
- Parenthesized constructs should have no surrounding whitespace. A single space follows commas, colons, and semicolons in those constructs. For example:

```
for (var i = 0, n = str.length; i < 10; i++) { }  
if (x < 10) { }  
function f(x: number, y: string): void { }
```