

Mastering advanced concepts in TypeScript



Jiri Lojda

Integrations Engineer
Kontent.ai

<https://github.com/JiriLojda>

Agenda

- Advantages of strict types
- Improving example #1 (Creating events)
 - Mapping union and object types
 - Conditional types
- Improving example #2 (Saving entities)
 - String templates
 - Iterating with conditional types
 - Connecting advanced types with runtime
- Bonus task

Application showcase.

<https://github.com/JiriLojda/react-summit-23>

Types are not your enemy

- Catch your bugs
- Explore APIs

Task #1

- Bug when creating event from an existing event
 - The name field should be populated with the existing event's name

Task #1 Takeaways

- Union types are great
 - Help capture the reality
- We can iterate over union types with conditional types
- We can easily transform types to another types
 - Reduces duplication => prevents bugs

Task #2

- Bug removing an event

Task #2 Takeaways

- TypeScript can parse literal string types
- We can do recursive iterations on a type level
- With advanced types we sometimes need to override a type ("as T") in a local function
- The "satisfies" keyword makes it easy to apply restrictions to arbitrary shapes

Bonus Task

- Type-level operations on numbers

Stay safe, use types.

Get in touch.



Jiri Lojda

Integrations Engineer
Kontent.ai

<https://github.com/JiriLojda>

<https://www.linkedin.com/in/jiri-lojda>