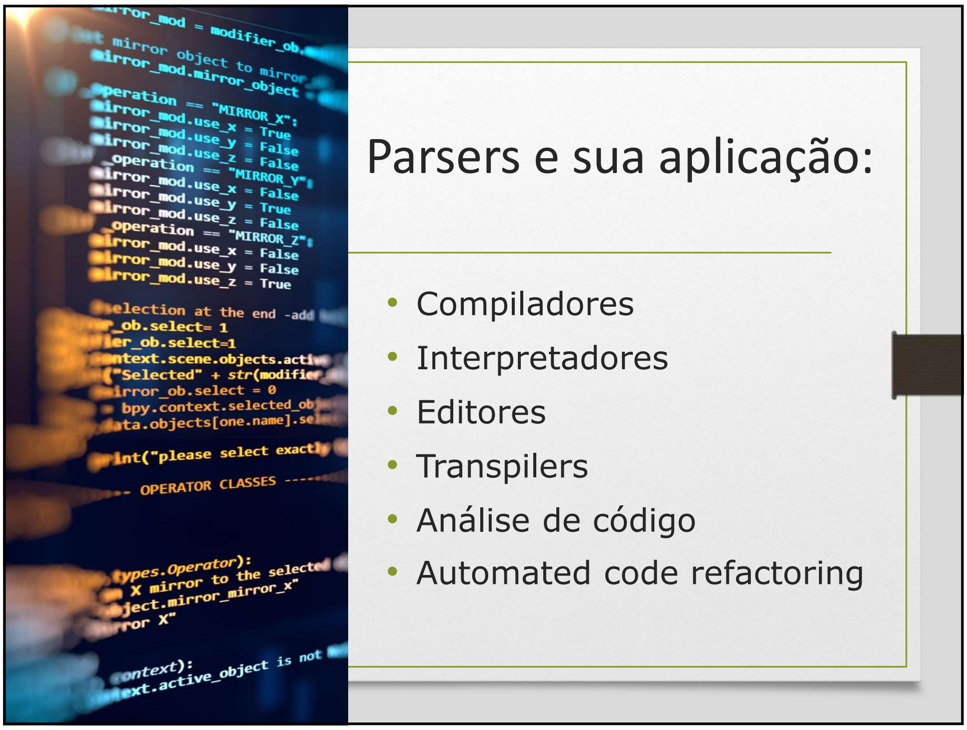
Compiladores Tópicos Avançados

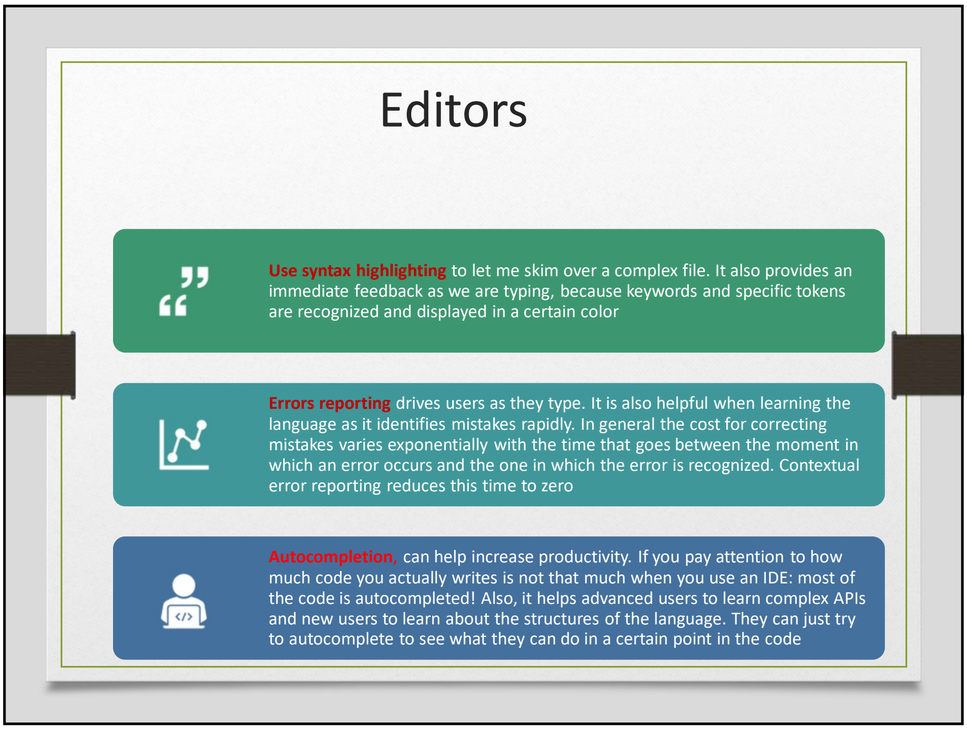
20



Parsers e sua aplicação:

- Compiladores
- Interpretadores
- Editores
- Transpilers
- Análise de código
- Automated code refactoring

21



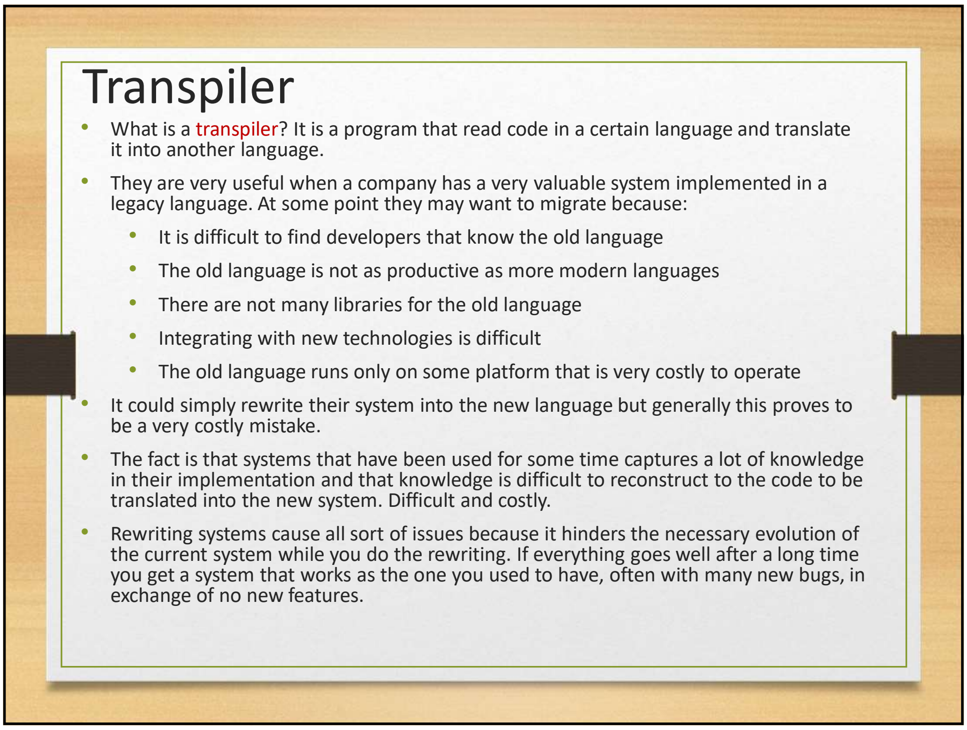
Editors

Use syntax highlighting to let me skim over a complex file. It also provides an immediate feedback as we are typing, because keywords and specific tokens are recognized and displayed in a certain color

Errors reporting drives users as they type. It is also helpful when learning the language as it identifies mistakes rapidly. In general the cost for correcting mistakes varies exponentially with the time that goes between the moment in which an error occurs and the one in which the error is recognized. Contextual error reporting reduces this time to zero

Autocompletion, can help increase productivity. If you pay attention to how much code you actually writes is not that much when you use an IDE: most of the code is autocompleted! Also, it helps advanced users to learn complex APIs and new users to learn about the structures of the language. They can just try to autocomplete to see what they can do in a certain point in the code

22



Transpiler

- What is a transpiler? It is a program that read code in a certain language and translate it into another language.
- They are very useful when a company has a very valuable system implemented in a legacy language. At some point they may want to migrate because:
  - It is difficult to find developers that know the old language
  - The old language is not as productive as more modern languages
  - There are not many libraries for the old language
  - Integrating with new technologies is difficult
  - The old language runs only on some platform that is very costly to operate
- It could simply rewrite their system into the new language but generally this proves to be a very costly mistake.
- The fact is that systems that have been used for some time captures a lot of knowledge in their implementation and that knowledge is difficult to reconstruct to the code to be translated into the new system. Difficult and costly.
- Rewriting systems cause all sort of issues because it hinders the necessary evolution of the current system while you do the rewriting. If everything goes well after a long time you get a system that works as the one you used to have, often with many new bugs, in exchange of no new features.

23