

Hachioji.pm #63

Jan. 28, 2016

About me

Hideaki Ohno

**A.K.A. `hide_o_55` / `hideo55`
Perl / C++ / Node.js / Go**

Rustの機運

Rust概観

- C++に近いのでとっつきやすい
- C++に無い部分も突飛なものではなく、他の言語である程度成功している概念が導入されている
 - トレイトとか

Rustの概観

- 代入ではなく束縛
 - そのままだとimmutable
 - mutableにするには`let mut`を使う
- 値と変数を紐付けて、束縛したものがその値の所有権をもつ
 - 値の寿命は所有権を持つ変数のスコープによって決定される

Rust概観

- 所有権と借用
 - いわゆる参照渡し
 - 借用されている値の寿命が保証される
 - だめな場合はコンパイル時にエラーとなる

Rust概観

- マッチとif letの組み合わせ
 - Option<T>型の扱いがかなり楽になる
 - エラーハンドリングとかでも使いやすそう

Rust概観

- 標準でテスト関連機能と、パッケージマネージャがある
 - 現代的言語ではないと話にならないともいう
 - パッケージ(クレート)は <https://crates.io/>

とりあえず今年はRustや
っていきます