Rust

- Rust is a strongly typed language

- Type conversions are very important

- Type inference plays important role

- Type conversion methods are defined as traits in *corelib* and *stdlib* under `core::convert` and `std::convert` modules

Conversion traits

- By-value conversion

  - `Into`

  - `From`

  - `TryInto`/`TryFrom` (unstable/RFC 1542)

- Reference conversion

  - `AsRef`/`AsMut` - convert something to a reference in a generic way