DS-210: PROGRAMMING FOR DATA SCIENCE

LECTURE 27

1. SEPARATING MODULES INTO MULTIPLE FILES

2. CRATES: WHAT ARE THEY?

3. USING EXTERNAL CRATES: `rand` (RANDOM NUMBERS) AND `csv` (READING CSV)

STARTING POINT: EXAMPLES FROM LAST WEEK

- Creating a graph representation
- Counting triangles
- Also: get first neighbor of a vertex

[live demo: one-file example]

MOVING A MODULE TO SEPARATE FILE

Content of module `abc`

- either in `src/abc.rs`
- or `src/abc/mod.rs`

MOVING A MODULE TO SEPARATE FILE

Content of module `abc`

- either in `src/abc.rs`
- or `src/abc/mod.rs`

In `main.rs` replace it with `mod abc;`

MOVING A MODULE TO SEPARATE FILE

Content of module `abc`

- either in `src/abc.rs`
- or `src/abc/mod.rs`

In `main.rs` replace it with `mod abc;`

Submodules:

Example `abc::def`

- either in `src/abc/def.rs`
- or in `src/abc/def/mod.rs`

Use `mod` recursively

MOVING A MODULE TO SEPARATE FILE

Content of module `abc`

- either in `src/abc.rs`
- or `src/abc/mod.rs`

In `main.rs` replace it with `mod abc;`

Submodules:

Example `abc::def`

- either in `src/abc/def.rs`
- or in `src/abc/def/mod.rs`

Use `mod` recursively

[live demo: splitting the sample file into `main.rs` and three submodules]

WHAT ARE CRATES?

Crates provided by a project:

- Each binary produced by a project (function `main` is the starting point)
    - So far we have seen single binaries
- A single library crate: can be used by other projects

SHARED CRATES

Where to find crates:

- Official list: https://crates.io
- Unofficial list: https://lib.rs

Documentation:

- https://docs.rs

CRATE `rand`: RANDOM NUMBERS

See: https://crates.io/crates/rand

CRATE `rand`: RANDOM NUMBERS

See: https://crates.io/crates/rand

Tell Rust you want to use it:

- edit `Cargo.toml`
- add `rand="0.8.5"` below dependencies

CRATE `rand`: RANDOM NUMBERS

See: https://crates.io/crates/rand

Tell Rust you want to use it:

- edit `Cargo.toml`
- add `rand="0.8.5"` below dependencies

To generate a random integer from 1 through 100:

- add `use rand::Rng`
- then `rand::thread_rng().gen_range(1..=100)`

CRATE `rand`: RANDOM NUMBERS

See: https://crates.io/crates/rand

Tell Rust you want to use it:

- edit `Cargo.toml`
- add `rand="0.8.5"` below dependencies

To generate a random integer from 1 through 100:

- add `use rand::Rng`
- then `rand::thread_rng().gen_range(1..=100)`

[live demo: add `get_random` to `graphs::neighbors`]

CRATE `csv` AND `serde`: READING A CSV FILE

See:

- https://crates.io/crates/rand
- https://crates.io/crates/serde

[live demo (to be continued in the next lecture)]