BLOCK NONCE #4: Use the standard library

What?

Rust (and Python) have large standard libraries. Use them to replace short loops and common patterns.

Why?

When we use the standard library the code is shorter, easier to write and easier to understand.

How?

```rust
if !words_per_length.contains_key(&len) {
    words_per_length.insert(len, 0);
}
*words_per_length.get_mut(&len).unwrap() += 1;
```

becomes

```rust
*words_per_length.entry(len).or_insert(0) += 1;
```

```rust
let mut index = 1;
for item in shopping_list.iter() {
    println!("{}. {}", index, item);
    index += 1
}
```

becomes

```rust
for (index, item) in shopping_list.iter().enumerate() {
    println!("{}. {}", index + 1, item);
}
```

There are many others:
- `vec!["A", "B", "C"].join(", ")` → `"A, B, C"`
- `vec![1, 2, 3, 4].iter().chunks(2)` → `[[1, 2], [3, 4]]`
- `number.next_multiple_of(x)`

> TIP💡 Most of the examples here have Python equivalents