Perl 6.d (Diwali)

Coming in 2018

New in 6.d: easier subclassing of IO::Handle type

You no longer have to override every single read or write method to create a custom handle, trying to ensure you didn't miss any of them. In 6.d, simply override `.WRITE` and `.READ`/`.EOF` methods and you're good to go.

Here we create a custom `IO::URL` handle reading from which gives you plain text content from a given URL. We override only `.READ` and `.EOF`, but you can fetch the data with `.slurp`, `.split`, `.comb`, `.lines`, `.get`, `.words`, `.readchars`, `.getc`, and `.read`! Feeling adventurous? Assign the handle to `$*IN` and use `prompt()`. It'll work!

```
class IO::URL is IO::Handle {
    has $.URL is required;
    has Buf $!content;
    submethod TWEAK {
        use WWW;        # ecosystem module for web page fetching
        use DOM::Tiny; # ecosystem module for HTML parsing
        $!content := Buf.new:
            DOM::Tiny.parse(get $!URL).all-text(:trim).encode;
        self.encoding: 'utf8';
    }
    method READ(\bytes) { $!content.splice: 0, bytes }
    method EOF         { not $!content }
}


my $fh := IO::URL.new: :URL<www.perl6.org>;

# .slurp and print all the content from the website. We can use all other
# read methods, such as .lines, or .get, or .readchars. All of them work
# correctly, even though we only defined .READ and .EOF
$fh.slurp.say;
```