



# Ripgrep haystack.rs: Code Companion

Reference code for the haystack.rs lecture. Sections correspond to the lecture document.

---

## Section 1: The Builder Pattern

```
/// A builder for constructing things to search over.
#[derive(Clone, Debug)]
pub(crate) struct HaystackBuilder {
    strip_dot_prefix: bool,
}

impl HaystackBuilder {
    /// Return a new haystack builder with a default configuration.
    pub(crate) fn new() -> HaystackBuilder {
        HaystackBuilder { strip_dot_prefix: false }
    }

    /// When enabled, if the haystack's file path starts with `.` then it is
    /// stripped.
    ///
    /// This is useful when implicitly searching the current working directory.
    pub(crate) fn strip_dot_prefix(
        &mut self,
        yes: bool,
    ) -> &mut HaystackBuilder {
        self.strip_dot_prefix = yes;
        self
    }
}
```

### Usage in HiArgs:

```
// From hiargs.rs
pub(crate) fn haystack_builder(&self) -> HaystackBuilder {
    let mut builder = HaystackBuilder::new();
    builder.strip_dot_prefix(self.paths.has_implicit_path);
    builder
}
```

### Why a builder for one field?























