What does defining getter/setter without a field allow?

You can use assignment and field access syntax even though you're running an arbitrary function.

How can you tell if an object is functional or stateful without seeing the implementation?

Functional objects' methods always return the same result for the same input, independent of temporal context.

Give a `var` a default value.

```
var x : Int = _
```

What do non-private `var`s automatically get?

What happens to field?

They get getters and setters with some visibility.
```
for var named x
getter named "x"
setter named "x_="
```
(but can assign with = normally)

The field itself is marked `private[this]`.

List the uses for Scala properties (overridable getters/setters).

- Enforce an invariant.
- Send events on access or modification.
- Access log.