**When to Use Accessor Methods**

In cases where you need to expose or modify internal data but still want to maintain control over it (e.g., validation or computation before returning the value), accessors can be useful. However, they should be used sparingly and thoughtfully, mainly in situations where:

1. **You are adhering to a specific framework** that relies on getter/setter patterns, such as JavaBeans.
2. **You need controlled access** to internal data, such as read-only access or computed properties.
3. **Data needs to be exposed in a way that protects the class's integrity**, ensuring that any changes to the internal state go through proper validation or logic.

The overall message is to avoid mindlessly relying on accessors, and instead focus on designing objects that fully encapsulate both their data and behavior.