

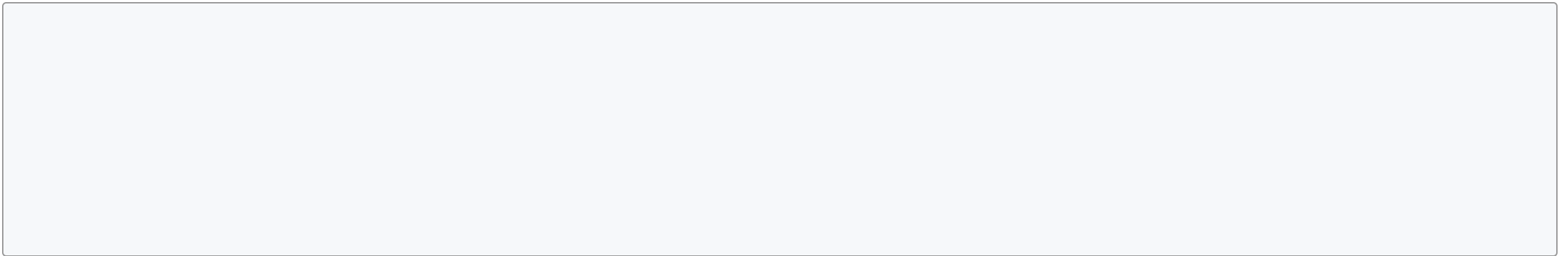
# Visitor pattern

- It's a design pattern specifically designed to traverse a tree
- It's an **essential** feature in all compilers in order to traverse AST

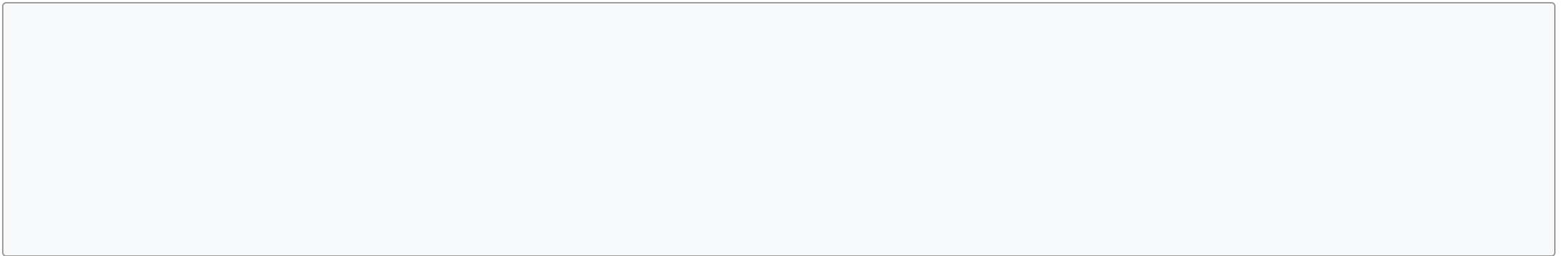
# How to use

- Create a class that extends `CoolTreeVisitor`
- If we have an object of type `CoolTreeNode` then we can use its `accept`
- Override any visit method of `CoolTreeVisitor` to intercept AST node  
type(`<CoolTreeVisitor>`)` method
  - This gives us a way to intercept the node of specific type
  - Do not forget to call `accept(this)` on the argument otherwise traverse will stop for the subsequent nodes

## Example AST Visitor



## Use our Visitor



# Lab assignmnet

In this lab we will write a program to dump a formatted (or pretty printed) a Cool program given it's AST