# Derek Ochal

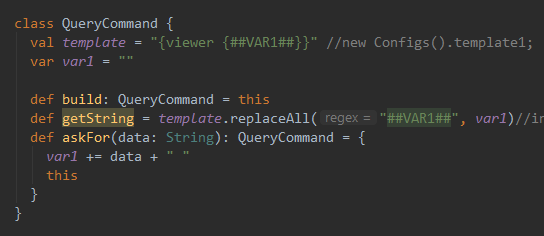
# Scala Framework White Paper

## Introduction

This project is a scala framework that uses monadic combinators to query Github API, thus hiding the complexity of the Github schema and GraphQL.

### Builder Pattern

Builder pattern was used to implement the QuerryCommand class. Github schema queries are complex objects and this pattern is apropriate for them.



QueryCommand objects have 2 variables of type string: template and var1.

They have 3 functions: build, getString, and askFor.

getString replaces any substring “##VAR1##” in the template with var1 and returns it.

askFor(string) adds the string to var1.

When the programmer writes “new QueryCommand()”, he only gets an unfinished query and he can’t use it until he modifies it.

When he writes “new QueryCommand().askFor(“email”)”, he gets a query object with var1 = “email “ in it. He can now use it. If he does, getString will be called and return “{viewer {email }}”.

When he writes “new QueryCommand().askFor(“name”).askFor(“login”)”, then var1 = “name login “ and this object’s getString function will return “{viewer {name login }}”.

Github case class also uses the Builder pattern. Its function jsonFromString was not implemented so the function displays the original string and builds a json object using constructors.

## Composite Pattern

When Representing json objects, composite pattern was used. Jsons are similar to trees and can be seen as nodes. When displaying the contents of this composite object, we can take advantage of functional programming paradigms of matching and aggregate functions. FoldLeft function used in getString is an implementation of preorder traversal used to display the contents of a json tree.

## Testing

1. Functional tests

The software was tested by using different access tokens and querying for different data, such as email. Results were compared with expected to test functional requirements.

1. Unit tests

Class QueryCommand was tested using Junit framework. Edge cases included passing an empty string or a null object.