**Assignment:**

Implementation of Easy-Parser. Existing tools for parsing such as lex and yacc are make the process of parsing complicated. Goal of this project is to focus on the problem of lexical analysis which includes identifying a token and then generating a code needed to be executed by creating own tool used for parsing.

**Project Section:**

* Conversion of the input file given to the compiler into an array of tokens.
* Each word from a valid statement in the input program file will be treated as a single token.
* Separating all tokens individually into categories.
* Clean-up spaces and comments from code.
* Identify and Verify the tokens from list of standard tokens dictionary using pattern matching process.
* Check Syntax of the input file and generate error accordingly.
* Parenthesis will be handled by stack balance method.

**Project Work Breakdown:**

**To have a group call for discussing the project work twice a week, i.e. on Sunday and Thursday.**

**Manav Parekh**:

* Creating Token Dictionary.
* Grouping into categories.
* Git-Hub repo. And Google drive.

**Tanay Parekh:**

* Create compiler class to separate tokens from code, including pattern matching to comparing with token dictionary.
* Implement Parenthesis-balancing method using stack.

**Raveena Mehta:**

* Clean-up blank space and comments.
* Check the syntax of the input file and generate error accordingly once found.

**Goal (I = Individua l, D= Dependent):**

* Setup git\_repo and google drive (D)
* Creating token dictionary (I)
* Clean-up blank space and comments (I)
* Creating Compiler with all matching methods and parenthesis-balancing method(I)
* Check the syntax of input and generate error if any(I).
* Creating report and presentation (D).