

analyzer.py

example

```
$ ./smile love
```

```
:)
```

```
$ ./smile hate
```

```
:(
```

```
$ ./smile Stanford
```

```
:|
```

TODO

- read distribution code
- initialize Analyzer
- analyze

TODO

- read distribution code
- initialize Analyzer
- analyze

positive-words.txt

negative-words.txt

- comments preceded by ";"
- each line ends with a newline
- some lines potentially blank

smile

- class Analyzer from module analyzer
- function colored from package termcolor
- main
 - ensures expected number of command line arguments
 - analyzes word and retrieves score
 - prints result, colored accordingly

analyzer.py

- `__init__`
 - ▣ load positive and negative words
- `analyze`
 - ▣ assign each word in text a value (-1, 0, 1)
 - ▣ calculate text's total score

TODO

- read distribution code
- initialize Analyzer**
- analyze

positive, negative words

- `list, dict, set...`
 - `self.positives, self.negatives`
- omit leading/trailing whitespace
 - `str.strip`
- don't include the comments
 - `str.startswith`

TODO

- read distribution code
- initialize Analyzer
- analyze

tokenize text

```
tokens = self.tokenizer.tokenize(text)
```

iteration

```
with open("file.txt") as lines:  
    for line in lines:  
        # TODO
```

analyze

- iterate over tokens
 - ▣ `str.lower`
- check if word is positive or negative:
 - ▣ if token in `self.positives`
- return score

TODO

- ☑ read distribution code
- ☑ initialize Analyzer
- ☑ analyze

this was analyzer.py