

model_with_word2Vec

November 30, 2024

```
[2]: # 1. Carregando as bibliotecas base
import numpy as np
import pandas as pd
import gensim

from sklearn import svm
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.decomposition import PCA
from sklearn.metrics import accuracy_score, recall_score, precision_score, f1_score, confusion_matrix, classification_report
from sklearn.preprocessing import StandardScaler
import xgboost as xgb

import seaborn as sns
from importlib import reload
import multiprocessing
import joblib

import scripts.preprocess as pp
import scripts.embeddings as eb
import scripts.convert as cv

reload(cv)
reload(pp)
reload(eb)
```

```
[2]: <module 'scripts.embeddings' from '/mnt/c/Users/luanm/projects/pece-monografia/src/models/scripts/embeddings.py'>
```

```
[3]: # 2. Carregar base de commits que será tratada

# Transformar blocos de código em commits
cv.convert_to_jsonl('./code_snippets', './inputs/commits.jsonl')

# Definir dataframe
df = pd.read_json('./inputs/commits.jsonl', lines=True)
df_labels = pd.read_csv('./inputs/labels.csv', sep=',')
```

```

labels = df_labels['Label']

print(f"Trecho de código com maior tamanho: {df['new_contents'].str.len().
      ↪min()}\n")
print(f"Trecho de código com maior tamanho: {df['new_contents'].str.len().
      ↪max()}\n")
print(f"Tamanho dos labels: {len(labels)}\n")

df.describe()

```

```

['GeneratedClass_1.cs', 'GeneratedClass_2.cs', 'GeneratedClass_3.cs',
'GeneratedClass_4.cs', 'GeneratedClass_5.cs', 'GeneratedClass_6.cs',
'GeneratedClass_7.cs', 'GeneratedClass_8.cs', 'GeneratedClass_9.cs',
'GeneratedClass_10.cs', 'GeneratedClass_11.cs', 'GeneratedClass_12.cs',
'GeneratedClass_13.cs', 'GeneratedClass_14.cs', 'GeneratedClass_15.cs',
'GeneratedClass_16.cs', 'GeneratedClass_17.cs', 'GeneratedClass_18.cs',
'GeneratedClass_19.cs', 'GeneratedClass_20.cs', 'GeneratedClass_21.cs',
'GeneratedClass_22.cs', 'GeneratedClass_23.cs', 'GeneratedClass_24.cs',
'GeneratedClass_25.cs', 'GeneratedClass_26.cs', 'GeneratedClass_27.cs',
'GeneratedClass_28.cs', 'GeneratedClass_29.cs', 'GeneratedClass_30.cs',
'GeneratedClass_31.cs', 'GeneratedClass_32.cs', 'GeneratedClass_33.cs',
'GeneratedClass_34.cs', 'GeneratedClass_35.cs', 'GeneratedClass_36.cs',
'GeneratedClass_37.cs', 'GeneratedClass_38.cs', 'GeneratedClass_39.cs',
'GeneratedClass_40.cs', 'GeneratedClass_41.cs', 'GeneratedClass_42.cs',
'GeneratedClass_43.cs', 'GeneratedClass_44.cs', 'GeneratedClass_45.cs',
'GeneratedClass_46.cs', 'GeneratedClass_47.cs', 'GeneratedClass_48.cs',
'GeneratedClass_49.cs', 'GeneratedClass_50.cs', 'GeneratedClass_51.cs',
'GeneratedClass_52.cs', 'GeneratedClass_53.cs', 'GeneratedClass_54.cs',
'GeneratedClass_55.cs', 'GeneratedClass_56.cs', 'GeneratedClass_57.cs',
'GeneratedClass_58.cs', 'GeneratedClass_59.cs', 'GeneratedClass_60.cs',
'GeneratedClass_61.cs', 'GeneratedClass_62.cs', 'GeneratedClass_63.cs',
'GeneratedClass_64.cs', 'GeneratedClass_65.cs', 'GeneratedClass_66.cs',
'GeneratedClass_67.cs', 'GeneratedClass_68.cs', 'GeneratedClass_69.cs',
'GeneratedClass_70.cs', 'GeneratedClass_71.cs', 'GeneratedClass_72.cs',
'GeneratedClass_73.cs', 'GeneratedClass_74.cs', 'GeneratedClass_75.cs',
'GeneratedClass_76.cs', 'GeneratedClass_77.cs', 'GeneratedClass_78.cs',
'GeneratedClass_79.cs', 'GeneratedClass_80.cs', 'GeneratedClass_81.cs',
'GeneratedClass_82.cs', 'GeneratedClass_83.cs', 'GeneratedClass_84.cs',
'GeneratedClass_85.cs', 'GeneratedClass_86.cs', 'GeneratedClass_87.cs',
'GeneratedClass_88.cs', 'GeneratedClass_89.cs', 'GeneratedClass_90.cs',
'GeneratedClass_91.cs', 'GeneratedClass_92.cs', 'GeneratedClass_93.cs',
'GeneratedClass_94.cs', 'GeneratedClass_95.cs', 'GeneratedClass_96.cs',
'GeneratedClass_97.cs', 'GeneratedClass_98.cs', 'GeneratedClass_99.cs',
'GeneratedClass_100.cs', 'GeneratedClass_101.cs', 'GeneratedClass_102.cs',
'GeneratedClass_103.cs', 'GeneratedClass_104.cs', 'GeneratedClass_105.cs',
'GeneratedClass_106.cs', 'GeneratedClass_107.cs', 'GeneratedClass_108.cs',

```

```
'GeneratedClass_109.cs', 'GeneratedClass_110.cs', 'GeneratedClass_111.cs',  
'GeneratedClass_112.cs', 'GeneratedClass_113.cs', 'GeneratedClass_114.cs',  
'GeneratedClass_115.cs']
```

Successfully converted 115 files to ./inputs/commits.jsonl

Trecho de código com maior tamanho: 0

Trecho de código com maior tamanho: 22251

Tamanho dos labels: 115

```
[3]:          new_contents  
count          115  
unique         108  
top  
freq           5
```

```
[4]: # 3. Preprocessar commits  
df['new_contents'] = df['new_contents'].apply(pp.clean_code)  
  
df.head()
```

```
[4]:          new_contents  
0  
1  <NEWLINE> using <LIBRARY> <NEWLINE>  
2  
3          <ATTRIBUTE> <NEWLINE>  
4    public <VARIABLENAME> <NEWLINE>
```

```
[5]: # 4. Escrever dataframe em csv para validação  
df.to_csv('./outputs/data_preprocessed.csv', sep=";", index=False)  
  
x_cleaned = df['new_contents']
```

```
[6]: # 5. Tokenização  
x_tokenized = [[c for c in s.split(" ") if c != ""] for s in x_cleaned]  
  
for x in x_tokenized:  
    print(x)
```

```
[]  
['<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>']  
[]  
['<ATTRIBUTE>', '<NEWLINE>']  
['public', '<VARIABLENAME>', '<NEWLINE>']  
['public', 'IntPtr', 'FindControlHandle(IntPtr', 'windowsHandle,',  
'<VARIABLENAME>', ')', '<NEWLINE>']
```

```

['public', '<VARIABLENAME>', '{', 'get;', 'set;', '}']
['try', '<NEWLINE>', '{', '<NEWLINE>', 'await', '<PRIVATEVARIABLE>',
'<METHOD_EXEC>:', '(>;', '<NEWLINE>', '<VARIABLENAME>', '=', 'true;',
'<NEWLINE>', '}', '<NEWLINE>', 'catch', '(', '<VARIABLENAME>', ')', '<NEWLINE>',
'{', '<NEWLINE>', '<VARIABLENAME>', '=', 'false;', '<NEWLINE>', '}',
'<NEWLINE>']
['<NEWLINE>', 'return', 'this;', '<NEWLINE>']
['protected', '<VARIABLENAME>', '<VARIABLENAME>', 'CreateContract(',
'<VARIABLENAME>', '<VARIABLENAME>', ')']
['<VARIABLENAME>', '=', '<CLASSINIT>', '<UpdateInfo>();', '<NEWLINE>',
'<VARIABLENAME>', '=', '<CLASSINIT>', '<string>();', '<NEWLINE>']
['static', 'void', 'Main(string[]', '<VARIABLENAME>', ')', '<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:True>', '{']
['<METHOD_DEF:LENGHT_GTT:True>', '{']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>']
['private', 'struct', '<VARIABLENAME>', '<NEWLINE>', '{',
'<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<VARIABLENAME>', ';', '<NEWLINE>', '<VARIABLENAME>', '=', '<VARIABLENAME>',
';', '<NEWLINE>', '}', '<NEWLINE>', 'public', '<VARIABLENAME>',
'<VARIABLENAME>', '{', 'get;', '}', '<NEWLINE>', 'public', '<VARIABLENAME>',
'{', 'get;', '}', '<NEWLINE>', '}']
['<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', 'return', 'new',
'<METHOD_EXEC>:', '(<PARAM>,<PARAM>);', '<NEWLINE>', '}']
['<NEWLINE>', '#region', '<NEWLINE>', '#endregion', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'namespace',
'<NAMESPACE>', '<NEWLINE>', '{', '<NEWLINE>', 'public', '<VARIABLENAME>',
'class', '<CLASSNAME>', ':', '<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>',
'public', 'static', '<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>',
'public', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', ';',
'<NEWLINE>', 'private', '<VARIABLENAME>', '=', '<OBJECTPROPERTYACCESS>', ';',
'<NEWLINE>', 'private', '<VARIABLENAME>', ';', '<NEWLINE>', 'private',
'<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>', 'private',
'<VARIABLENAME>', ';', '<NEWLINE>', '<NEWLINE>', 'private', '<VARIABLENAME>',
'<VARIABLENAME>', '=', '<STRINGVALUE>', ';', '<METHOD_DEF:LENGHT_GTT:False>',
'{', '<NEWLINE>', '<METHOD_EXEC>:', '(>;', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '<PRIVATEVARIABLE>',
'<METHOD_EXEC>:', '(<PARAM>);', '<NEWLINE>', '<PRIVATEVARIABLE>',

```

```

'METHOD_EXEC:', '(<PARAM>)>';', '<NEWLINE>', 'METHOD_EXEC:', '()>';',
'<NEWLINE>', '<OBJECTPROPERTYACCESS>', '=', 'new', '<CLASSMETHODINVOCATION>',
';', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '=', '<OBJECTPROPERTYACCESS>', ';',
'<NEWLINE>', 'METHOD_EXEC:', '()>';', '<NEWLINE>', '<OBJECTPROPERTYACCESS>',
 '=', 'true;';', '<NEWLINE>', '<VARIABLENAME>', '=', '<OBJECTPROPERTYACCESS>', ';',
'<NEWLINE>', '<VARIABLENAME>', '=', '<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>',
'<OBJECTPROPERTYACCESS>', '=', 'new', 'METHOD_EXEC:', '()>';', '<NEWLINE>',
'<OBJECTPROPERTYACCESS>', '+=', '<VARIABLENAME>', ';', '<NEWLINE>',
'<OBJECTPROPERTYACCESS>', '+=', '<VARIABLENAME>', ';', '<NEWLINE>', 'foreach',
'(', '<VARIABLENAME>', '<VARIABLENAME>', 'in', '<OBJECTPROPERTYACCESS>', ')',
'<NEWLINE>', '{', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '+=', '<VARIABLENAME>',
';', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '+=', '<VARIABLENAME>', ';',
'<NEWLINE>', '}', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '+=', '(sender,',
'EXTERNALCLASSREF>)', '=>', '<NEWLINE>', '{', '<NEWLINE>', 'if', '(',
'<VARIABLENAME>', '&&', '!', '<OBJECTPROPERTYACCESS>',
'ActorCharacterChooser().Any())', '<NEWLINE>', '{', '<NEWLINE>',
'<VARIABLENAME>', '=', '<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>',
'<VARIABLENAME>', '=', 'new', 'METHOD_EXEC:', '(<PARAM>', '<PARAM>',
'<PARAM>', '<PARAM>)>';', '<NEWLINE>', '<PRIVATEVARIABLE>', 'METHOD_EXEC:',
'(<PARAM>', '<PARAM>', '<PARAM>', '<PARAM>)>';', '<NEWLINE>', '}',
'<NEWLINE>', '};', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '+=',
'<VARIABLENAME>', ';', '<NEWLINE>', '}', 'METHOD_DEF:LENGHT_GTT:True>', '{',
'<NEWLINE>', '<VARIABLENAME>', '=', '<PRIVATEVARIABLE>',
'CLASSMETHODINVOCATION>', ');', '<NEWLINE>', 'if', '(', '<VARIABLENAME>', '!=',
'<VARIABLENAME>', ')', '<NEWLINE>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<VARIABLENAME>', ';', '<NEWLINE>', '<PRIVATEVARIABLE>', 'METHOD_EXEC:',
'()>';', '<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>', 'public',
'<VARIABLENAME>', '<VARIABLENAME>', '{', 'get;', 'private', 'set;', '}',
'METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'METHOD_EXEC:', '()>';',
'<NEWLINE>', '}', 'METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', 'using',
'(', '<VARIABLENAME>', '=', '<CLASSINIT>', '()>', '<NEWLINE>', '{', '<NEWLINE>',
'if', '(', '<OBJECTPROPERTYACCESS>', '==', '<OBJECTMETHODINVOCATION>', ')',
'<NEWLINE>', '{', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '=', '-',
'<NUMBERVALUE>', ';', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '=', '-',
'<NUMBERVALUE>', ';', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '=', '-',
'<NUMBERVALUE>', ';', '<NEWLINE>', 'METHOD_EXEC:', '(<PARAM>)>';', '<NEWLINE>',
'return', 'true;';', '<NEWLINE>', '}', '<NEWLINE>', 'return', 'false;';',
'<NEWLINE>', '}', '<NEWLINE>', '}', 'METHOD_DEF:LENGHT_GTT:True>', '{',
'<NEWLINE>', '<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', '<VARIABLENAME>',
 '=', 'false;';', '<NEWLINE>', 'if', '(!', '<OBJECTMETHODINVOCATION>', ')',
'<NEWLINE>', '{', '<NEWLINE>', '<VARIABLENAME>', '=', 'METHOD_EXEC:',
'(<PARAM>)>';', '<NEWLINE>', '}', '<NEWLINE>', 'if', '(!loaded)', '<NEWLINE>',
'{', '<NEWLINE>', 'if', '(!ChooseProject())', '<NEWLINE>', '{', '<NEWLINE>',
'METHOD_EXEC:', '()>';', '<NEWLINE>', 'return;';', '<NEWLINE>', '}', '<NEWLINE>',
'<NEWLINE>', '<VARIABLENAME>', '=', '<OBJECTPROPERTYACCESS>', ';',
'<NEWLINE>', 'if', '((', '<OBJECTPROPERTYACCESS>', '>',
'<OBJECTPROPERTYACCESS>', ')', '&&', '(', '<OBJECTPROPERTYACCESS>', '>',
'<OBJECTPROPERTYACCESS>', ')', '&&', '(IsOnScreen(' , '<VARIABLENAME>', ')))',

```

```

'<NEWLINE>', '{', '<NEWLINE>', '<VARIABLENAME>', '=', '<OBJECTPROPERTYACCESS>',
';', '<NEWLINE>', '<VARIABLENAME>', '=', '<OBJECTPROPERTYACCESS>', ';',
'<NEWLINE>', '<VARIABLENAME>', '=', '<VARIABLENAME>', ';', '<NEWLINE>', '}',
'<NEWLINE>', 'else', '<NEWLINE>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '}', '<NEWLINE>', '<VARIABLENAME>',
'=', '<CLASSINIT>', '(@', '<STRINGVALUE>', '<OBJECTPROPERTYACCESS>',
'<STRINGVALUE>', '<PRIVATEVARIABLE>', '..', '<STRINGVALUE>', '<VARIABLENAME>',
'<VARIABLENAME>', 'in', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>',
'<VARIABLENAME>', '<STRINGVALUE>', '<OBJECTPROPERTYACCESS>', '<STRINGVALUE>',
'<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', 'work,',
'while', '<VARIABLENAME>', '<VARIABLENAME>', '<PRIVATEVARIABLE>',
'<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>',
'<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>',
'<VARIABLENAME>', ':-)', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>',
'<VARIABLENAME>', 'for', '<VARIABLENAME>', '<VARIABLENAME>', 'recordings,',
'<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', '<PRIVATEVARIABLE>',
'<STRINGVALUE>', '.zip', '<STRINGVALUE>', '.zip', '<STRINGVALUE>',
'<STRINGVALUE>', '<VARIABLENAME>', '<STRINGVALUE>', '<VARIABLENAME>',
'<VARIABLENAME>', '(*', '<STRINGVALUE>', ')*', '<STRINGVALUE>',
'<OBJECTPROPERTYACCESS>', '<STRINGVALUE>', '<VARIABLENAME>', '<VARIABLENAME>',
'<VARIABLENAME>', 'does', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>',
'<VARIABLENAME>', 'for', '{', '<NUMBERVALUE>', '}', '<VARIABLENAME>',
'<VARIABLENAME>', '<VARIABLENAME>', 'for', '{', '<NUMBERVALUE>', '}',
'<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>',
'<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>',
'<VARIABLENAME>', '<PRIVATEVARIABLE>', '<STRINGVALUE>', '<VARIABLENAME>', '{',
'<NUMBERVALUE>', '}', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>',
'for', '<VARIABLENAME>', '<VARIABLENAME>', '<VARIABLENAME>', 'name,', '{',
'<NUMBERVALUE>', '}', 'for', '<VARIABLENAME>', '<VARIABLENAME>', 'in',
'<VARIABLENAME>', '<VARIABLENAME>', '<STRINGVALUE>', 'https:', '<NEWLINE>', '}',
'<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'if', '(',
'<EXTERNALCLASSREF>', '==', '<VARIABLENAME>', ')', 'return;', '<NEWLINE>',
'<VARIABLENAME>', '=', 'false;', '<NEWLINE>', '<VARIABLENAME>', '=', 'new',
'<ARRAYACCESS>', ';', '<NEWLINE>', 'foreach', '(', '<VARIABLENAME>', 'in',
'<VARIABLENAME>', ')', '<NEWLINE>', '{', '<NEWLINE>', 'if', '(!initialized)',
'<NEWLINE>', '{', '<NEWLINE>', 'if', '(', '<EXTERNALCLASSREF>.Exists)',
'<NEWLINE>', '{', '<NEWLINE>', '<EXTERNALCLASSREF>.<METHOD_EXEC>', '()>;',
'<NEWLINE>', '}', '<NEWLINE>', '<ARRAYACCESS>', '=', '<STRINGVALUE>', ';',
'<NEWLINE>', '<ARRAYACCESS>', '=', '<STRINGVALUE>', ';', '<NEWLINE>',
'<ARRAYACCESS>', '=', '<STRINGVALUE>', ';', '<NEWLINE>',
'<EXTERNALCLASSREF>.<METHOD_EXEC>', '(<PARAM>)>;', '<NEWLINE>', 'initialized',
'=', 'true;', '<NEWLINE>', '}' ]
['<NEWLINE>', 'public', '<VARIABLENAME>', '=>', 'this', '==', '<VARIABLENAME>',
';', '<NEWLINE>', 'public', '<VARIABLENAME>', '<VARIABLENAME>', '()', '=>',
'<NUMBERVALUE>', ';', '<NEWLINE>', 'internal', '<VARIABLENAME>',
'<VARIABLENAME>', '=>', 'false;', '<NEWLINE>', 'internal', '<VARIABLENAME>',

```

```

'<VARIABLENAME>', 'GetMinHashEntryOrDefault()', '=>', '<VARIABLENAME>', ';',
'<NEWLINE>', 'internal', '<VARIABLENAME>', '<VARIABLENAME>',
'GetMaxHashEntryOrDefault()', '=>', '<VARIABLENAME>', ';', '<NEWLINE>',
'internal', '<VARIABLENAME>', '<VARIABLENAME>', 'GetEntryOrNull(',
'<VARIABLENAME>', ')', '=>', '<VARIABLENAME>', ';']
['<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '<CLASSMETHODINVOCATION>',
';', '<NEWLINE>', '}']
['<NEWLINE>', 'public', '<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>', 'get',
'<NEWLINE>', '{', '<NEWLINE>', 'return', '<OBJECTPROPERTYACCESS>', ';',
'<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>', 'public', '<VARIABLENAME>',
'<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>', 'get', '<NEWLINE>', '{',
'<NEWLINE>', 'return', '<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '}',
'<NEWLINE>', 'set', '<NEWLINE>', '{', '<NEWLINE>', '<OBJECTPROPERTYACCESS>',
'=', '<VARIABLENAME>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}']
['static', 'void', 'ThrowInvalidOperationException(', '<VARIABLENAME>', ')',
'=>', 'throw', 'new', '<METHOD_EXEC:', '(<PARAM>)>', '<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<OBJECTPROPERTYACCESS>',
'=', '<VARIABLENAME>', ';', '<NEWLINE>', '<VARIABLENAME>', '=', 'index;',
'<NEWLINE>', '}']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'try', '<NEWLINE>', '{',
'<NEWLINE>', '<VARIABLENAME>', '=', '<OBJECTMETHODINVOCATION>', ',',
'<NUMBERVALUE>', ',', '<VARIABLENAME>', ')', '!=', '-', '<NUMBERVALUE>', ');',
'<NEWLINE>', 'return', '<VARIABLENAME>', ';', '<NEWLINE>', '}', '<NEWLINE>',
'catch', '<NEWLINE>', '{', '<NEWLINE>', 'return', 'false;', '<NEWLINE>', '}',
'<NEWLINE>', '}', '<NEWLINE>']
['public', 'interface', '<VARIABLENAME>', '{', '<NEWLINE>', '<VARIABLENAME>',
'<VARIABLENAME>', '{', 'get;', 'set;', '}', '<NEWLINE>', '}', '<NEWLINE>']
['string[]', '<VARIABLENAME>', '=', '{', '<STRINGVALUE>', ',', '<STRINGVALUE>',
',', '<STRINGVALUE>', ',', '<NEWLINE>', '<STRINGVALUE>', ',', '<STRINGVALUE>',
',', '<STRINGVALUE>', ',', '<STRINGVALUE>', '};', '<NEWLINE>', 'foreach', '(',
'<VARIABLENAME>', 'in', '<VARIABLENAME>', ')', '{', '<NEWLINE>', 'if', '(!',
'<CLASSMETHODINVOCATION>', ')', '{', '<NEWLINE>', '<CLASSMETHODINVOCATION>',
';', '<NEWLINE>', '}', '<NEWLINE>', '}']
['<VARIABLENAME>', '=', '<CLASSINIT>', '<FilePath>', '<VARIABLENAME>', '>',
'((process', '<settings>', '=>', '<NEWLINE>', '{', '<NEWLINE>',
'<VARIABLENAME>', '=', '<METHOD_EXEC:', '(<PARAM>', '<PARAM>)>', '<NEWLINE>',
'if', '(', '<VARIABLENAME>', '!=', '<NUMBERVALUE>', ')', '{', '<NEWLINE>',
'throw', 'new', '<METHOD_EXEC:', '(<PARAM>)>', '<NEWLINE>', '}', '<NEWLINE>',
'});', '<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<NUMBERVALUE>', ';', '<NEWLINE>', 'while', '(', '<EXTERNALCLASSREF>.Success)',
'<NEWLINE>', '{', '<NEWLINE>', '<CLASSMETHODINVOCATION>', ');', '<NEWLINE>',
'for', '(', '<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<VARIABLENAME>', '<= >',
'<NUMBERVALUE>', ';', '<VARIABLENAME>', '++>', '<NEWLINE>', '{', '<NEWLINE>',
'<VARIABLENAME>', '=', '<EXTERNALCLASSREF>.', '<ARRAYACCESS>', ';', '<NEWLINE>',
'<CLASSMETHODINVOCATION>', ';', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', 'for', '(', '<VARIABLENAME>', '=',
'<NUMBERVALUE>', ';', '<VARIABLENAME>', '<', '<OBJECTPROPERTYACCESS>', ';',

```

```

'<VARIABLENAME>', '++)', '<NEWLINE>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<ARRAYACCESS>', ';', '<NEWLINE>', '<CLASSMETHODINVOCATION>', ';', '<NEWLINE>',
'}', '<NEWLINE>', '}', '<NEWLINE>', '<EXTERNALCLASSREF>', '=',
'<EXTERNALCLASSREF>.<METHOD_EXEC>', '()>;', '<NEWLINE>', '}', '<NEWLINE>', '}',
'<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'return', '<METHOD_EXEC>',
'(<PARAM>', '<PARAM>', '<PARAM>', '<PARAM>)>;', '<NEWLINE>', '}',
'<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<VARIABLENAME>', '??', 'throw', '<CLASSINIT>', '(nameof(', '<VARIABLENAME>',
'))';', '<NEWLINE>', '<VARIABLENAME>', '=', '<VARIABLENAME>', '??', 'throw',
'<CLASSINIT>', '(nameof(', '<VARIABLENAME>', '));', '<NEWLINE>',
'<VARIABLENAME>', '=', '<METHOD_EXEC>', '(<PARAM>', '<PARAM>)>;', '<NEWLINE>',
'if', '(', '<VARIABLENAME>', '!=', '<VARIABLENAME>', ')', '<NEWLINE>', '{',
'<NEWLINE>', '<VARIABLENAME>', '=', '<METHOD_EXEC>', '(<PARAM>)>;', '<NEWLINE>',
'<VARIABLENAME>', '=', 'new', '<METHOD_EXEC>', '(<PARAM>', '<PARAM>',
'<PARAM>', '<PARAM>', '<PARAM>', '<PARAM>', '<PARAM>',
'<PARAM>)>;', '<NEWLINE>', 'source?.<METHOD_EXEC>', '(<PARAM>)>;', '<NEWLINE>',
'return', '<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '}', '<NEWLINE>',
'return', 'false;', '<NEWLINE>', '}']
['<NEWLINE>', 'public', '<VARIABLENAME>', '<VARIABLENAME>', '{', 'get;', '}',
'<NEWLINE>', 'internal', '<VARIABLENAME>', '<VARIABLENAME>', '{', 'get;', '}',
'<NEWLINE>', 'internal', '<VARIABLENAME>', '<VARIABLENAME>', '{', 'get;', '}',
'<NEWLINE>', 'internal', '<VARIABLENAME>', '<Suffix>', '<VARIABLENAME>', '{',
'get;', '}', '<NEWLINE>', 'internal', '<VARIABLENAME>', '<Root>',
'<VARIABLENAME>', '{', 'get;', '}', '<NEWLINE>', 'private', '<VARIABLENAME>',
'<VARIABLENAME>', '{', 'get;', '}']
['<NEWLINE>', '<ATTRIBUTE>', '<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>',
'throw', 'new', '<METHOD_EXEC>', '(<PARAM>)>;', '<NEWLINE>', '}']
['<NEWLINE>', 'public', '<VARIABLENAME>', '<VARIABLENAME>', '=',
'<NUMBERVALUE>', ';', '<NEWLINE>', 'public', '<VARIABLENAME>', '<VARIABLENAME>',
 '=', '<NUMBERVALUE>', ';', '<NEWLINE>', 'public', '<VARIABLENAME>',
'<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<NEWLINE>', 'public',
'<VARIABLENAME>', '<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<NEWLINE>']
['<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>']
['<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'namespace', '<NAMESPACE>', '<NEWLINE>', '{', '<NEWLINE>',
'public', 'enum', '<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>', 'Stretch,',
'<NEWLINE>', 'Left,', '<NEWLINE>', 'Center,', '<NEWLINE>', 'Right,',
'<NEWLINE>', '}', '<NEWLINE>', 'public', 'enum', '<VARIABLENAME>', '<NEWLINE>',

```



```

'{' , '<NEWLINE>' , 'Stretch,' , '<NEWLINE>' , 'Top,' , '<NEWLINE>' , 'Center,' ,
'<NEWLINE>' , 'Bottom,' , '<NEWLINE>' , '}' , '<NEWLINE>' , '}' , '<NEWLINE>']
[]
['<METHOD_DEF:LENGHT_GTT:False>' , '{' , '<NEWLINE>' , '<VARIABLENAME>' , '=',
'<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<VARIABLENAME>' ,
';' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<VARIABLENAME>' , ';' , '<NEWLINE>' ,
'<VARIABLENAME>' , '=', '<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' ,
'=', '<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' , '=',
'<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<VARIABLENAME>' ,
';' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<VARIABLENAME>' , ';' , '<NEWLINE>' ,
'<VARIABLENAME>' , '=', '<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' ,
'=', '<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' , '=',
'<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<VARIABLENAME>' ,
';' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<VARIABLENAME>' , ';' , '<NEWLINE>' ,
'<VARIABLENAME>' , '=', '<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' ,
'=', '<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' , '=',
'<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<VARIABLENAME>' ,
';' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<VARIABLENAME>' , ';' , '<NEWLINE>' ,
'<VARIABLENAME>' , '=', '<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' ,
'=', '<VARIABLENAME>' , ';' , '<NEWLINE>' , '}' ]
['<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' ,
'<NEWLINE>' , 'using' , '<LIBRARY>']
['<NEWLINE>' , 'protected' , '<VARIABLENAME>' , ';' , '<NEWLINE>' , 'protected' ,
'<VARIABLENAME>' , ';' , '<NEWLINE>' , 'protected' , '<VARIABLENAME>' , ';' ]
[]
['while' , '(' , '<OBJECTMETHODINVOCATION>' , ')' , '<NEWLINE>' , '{' , '<NEWLINE>' ,
'++homograph;' , '<NEWLINE>' , '<VARIABLENAME>' , '=', '<OBJECTPROPERTYACCESS>' ,
'+' , '<STRINGVALUE>' , '+' , '<VARIABLENAME>' , ';' , '<NEWLINE>' , '}' ]
['private' , 'readonly' , '<VARIABLENAME>' , ';' , '<NEWLINE>' , 'private' , 'long' ,
'<VARIABLENAME>' , '=', '-' , '<NUMBERVALUE>' , ';' , '<NEWLINE>' , 'private' ,
'long' , '<VARIABLENAME>' , '=', '-' , '<NUMBERVALUE>' , ';' ]
['<NEWLINE>' , '<VARIABLENAME>' , '=', '<STRINGVALUE>' , ';' , '<NEWLINE>' ,
'<VARIABLENAME>' , '+=', '<OBJECTPROPERTYACCESS>' , '+' , '<VARIABLENAME>' , ';' ,
'<NEWLINE>' , '<VARIABLENAME>' , '+=', '<OBJECTPROPERTYACCESS>' , '+' ,
'<OBJECTPROPERTYACCESS>' , '+' , '<STRINGVALUE>' , ';' , '<NEWLINE>' ,
'<VARIABLENAME>' , '+=', '<OBJECTPROPERTYACCESS>' , '+' , '<VARIABLENAME>' , ';' ,
'<NEWLINE>' , '<VARIABLENAME>' , '+=', '<OBJECTPROPERTYACCESS>' , '+' ,
'<OBJECTPROPERTYACCESS>' , '+' , '<STRINGVALUE>' , ';' , '<NEWLINE>' ,
'<VARIABLENAME>' , '+=', '<OBJECTPROPERTYACCESS>' , '+' , '<VARIABLENAME>' , ';' ]
['<NEWLINE>' , 'for' , '(' , '<VARIABLENAME>' , '=', '<NUMBERVALUE>' , ';' ,
'<VARIABLENAME>' , '<' , '<OBJECTPROPERTYACCESS>' , ';' , '<VARIABLENAME>' , '+=',
'<NUMBERVALUE>' , ')' , '<NEWLINE>' , '{' , '<NEWLINE>' , '<VARIABLENAME>' , '=',
'<CLASSMETHODINVOCATION>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' , '=',
'<OBJECTMETHODINVOCATION>' , ';' , '<NEWLINE>' , 'if' , '(' ,
'<OBJECTPROPERTYACCESS>' , '>' , '<NUMBERVALUE>' , ')' , '<NEWLINE>' , '{' ,
'<NEWLINE>' , '<VARIABLENAME>' , '=', '<STRINGVALUE>' , '+' , '<VARIABLENAME>' , ';' ,
'<NEWLINE>' , '}' , '<NEWLINE>' , '<OBJECTMETHODINVOCATION>' , ';' , '<NEWLINE>' ,
'}']

```

```

['private', 'static', '<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>',
'private', 'static', 'long', '<VARIABLENAME>', ';', '<NEWLINE>', 'private',
'static', '<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>', 'private',
'static', '<VARIABLENAME>', ';']
[]
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<METHOD_EXEC:', '(<PARAM>',
'<PARAM>)>;', '<NEWLINE>', '}', '<NEWLINE>']
['<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'namespace', '<NAMESPACE>', '<NEWLINE>', '{', '<NEWLINE>',
'public', 'class', '<CLASSNAME>', ':', 'ModRandom,', '<VARIABLENAME>',
'<NEWLINE>', '{', '<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>']
['private', 'readonly', '<VARIABLENAME>', '<VARIABLENAME>', ';',
'<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '}']
['protected', 'readonly', '<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>',
'protected', 'readonly', '<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>',
'protected', 'readonly', '<VARIABLENAME>', '<VARIABLENAME>', ';']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<OBJECTPROPERTYACCESS>',
'=', '<EXTERNALCLASSREF>;', '<NEWLINE>', '<CLASSMETHODINVOCATION>', ';',
'<NEWLINE>', '}', '<NEWLINE>']
['<NEWLINE>', 'private', '<VARIABLENAME>', '<VARIABLENAME>', '=',
'<NUMBERVALUE>', ';', '<NEWLINE>', 'private', '<VARIABLENAME>',
'<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<NEWLINE>', 'private',
'<VARIABLENAME>', '<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<NEWLINE>',
'private', '<VARIABLENAME>', '<VARIABLENAME>', '=', '<NUMBERVALUE>', ';',
'<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'true;', '<NEWLINE>', '<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', '}']
['private', '<VARIABLENAME>', '<VARIABLENAME>', '{', 'get;', '}',
'<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<VARIABLENAME>', ';', '<NEWLINE>', '}', '<NEWLINE>']
['public', 'enum', '<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>',
'<VARIABLENAME>', '=', '0x01,', '<NEWLINE>', '<VARIABLENAME>', '=', '0x02,',
'<NEWLINE>', '<VARIABLENAME>', '=', '0x03,', '<NEWLINE>', '<VARIABLENAME>', '=',
'0x04,', '<NEWLINE>', '<VARIABLENAME>', '=', '<VARIABLENAME>', '<NEWLINE>', '}',
'<NEWLINE>']
['using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>']
['using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'namespace',
'<NAMESPACE>', '<NEWLINE>', '{', '<NEWLINE>', 'class', '<CLASSNAME>',
'<NEWLINE>', '{', '<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>']

```

```

['<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '<VARIABLENAME>', '=',
'<NUMBERVALUE>', ';', '<NEWLINE>', '<VARIABLENAME>', '=', '<STRINGVALUE>', ';',
'<NEWLINE>', '<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<NEWLINE>',
'<VARIABLENAME>', '=', 'false;', '<NEWLINE>', '}', '<NEWLINE>']
['<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'namespace', '<NAMESPACE>',
'<NEWLINE>', '{', '<NEWLINE>', 'public', 'class', '<CLASSNAME>', '<NEWLINE>',
'{', '<NEWLINE>', 'public', 'static', '<VARIABLENAME>', '=', '<STRINGVALUE>',
';', '<NEWLINE>', 'public', 'static', '<VARIABLENAME>', '=',
'<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', 'public', 'static',
'<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}',
'<NEWLINE>']
['public', '<VARIABLENAME>', '{', '<NEWLINE>', 'get', '{', '<NEWLINE>',
'return', '<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '}', '<NEWLINE>', 'set',
'{', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '=', '<VARIABLENAME>', ';',
'<NEWLINE>', '}', '<NEWLINE>', '}']
['<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '}']
['public', 'static', '<VARIABLENAME>', '<VARIABLENAME>', '=>', '<STRINGVALUE>',
';', '<NEWLINE>', 'public', 'static', '<VARIABLENAME>', '<VARIABLENAME>', '=>',
'$', '<STRINGVALUE>', ';', '<NEWLINE>', 'public', 'static', '<VARIABLENAME>',
'<VARIABLENAME>', '=', '<STRINGVALUE>', ';', '<NEWLINE>', 'private', 'static',
'<VARIABLENAME>', '<VARIABLENAME>', ';']
[]
[]
['for', '(', '<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<VARIABLENAME>', '<',
'<OBJECTPROPERTYACCESS>', ';', '<VARIABLENAME>', '++', '<NEWLINE>', '{',
'<NEWLINE>', '<OBJECTMETHODINVOCATION>', '+', '<STRINGVALUE>', '+',
'<ARRAYACCESS>', '.score,', 'temp,', '<OBJECTPROPERTYACCESS>', ');',
'<NEWLINE>', '<OBJECTPROPERTYACCESS>', '+=', '<NUMBERVALUE>', ';', '<NEWLINE>',
'}']
['<OBJECTPROPERTYACCESS>', 'e(', '<STRINGVALUE>', ',', '<STRINGVALUE>', ',',
'<NEWLINE>', 'new', '{', '<VARIABLENAME>', '=', '<STRINGVALUE>', ',',
'<VARIABLENAME>', '=', '<STRINGVALUE>', '}', '<NEWLINE>', 'new', '{',
'<VARIABLENAME>', '=', '<CLASSINIT>', '(', '<STRINGVALUE>', ')', '}', ')];']
['public', 'class', '<CLASSNAME>', '<NEWLINE>', '{', '<NEWLINE>', 'public',
'<VARIABLENAME>', '{', 'get;', 'set;', '}', '<NEWLINE>', '}']
['<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', 'if', '(', '<VARIABLENAME>',
'==', '<VARIABLENAME>', ')', '{', '<NEWLINE>', '<CLASSMETHODINVOCATION>', ';',
'<NEWLINE>', 'return;', '<NEWLINE>', '}']
['using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>',
'namespace', '<NAMESPACE>', '<NEWLINE>', '{', '<NEWLINE>', '}', '<NEWLINE>']
['using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'namespace', '<NAMESPACE>', '<NEWLINE>', '{',
'<NEWLINE>', '<ATTRIBUTE>', '<NEWLINE>', 'public', 'class', '<CLASSNAME>',
'<NEWLINE>', '{', '<NEWLINE>', '<ATTRIBUTE>', '<METHOD_DEF:LENGHT_GTT:False>',

```

```

'{' , '<NEWLINE>' , '<CLASSMETHODINVOCATION>' , ';' , '<NEWLINE>' , '}' , '<NEWLINE>' ,
'}' , '<NEWLINE>' , '}' , '<NEWLINE>']
['<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' ,
'<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' ,
'<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' ,
'<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'namespace' , '<NAMESPACE>' ,
'<NEWLINE>' , '{' , '<NEWLINE>' , 'public' , 'class' , '<CLASSNAME>' , '<NEWLINE>' ,
'{' , '<NEWLINE>' , '<ATTRIBUTE>' , '<NEWLINE>' , 'protected' , 'static' , 'extern' ,
'int' , '<METHOD_EXEC:' , '(<PARAM>' , '<PARAM>' , '<PARAM>)>' , '<NEWLINE>' , '}' ,
'<NEWLINE>' , '}' , '<NEWLINE>']
['<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' ,
'<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' ,
'<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' ,
'<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' ,
'<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'namespace' , '<NAMESPACE>' ,
'<NEWLINE>' , '{' , '<NEWLINE>' , 'public' , '<VARIABLENAME>' , 'class' ,
'<CLASSNAME>' , ':' , '<VARIABLENAME>' , '<NEWLINE>' , '{' ,
'<METHOD_DEF:LENGHT_GTT:False>' , '{' , '<NEWLINE>' , '<METHOD_EXEC:' , '()'>' ,
'<NEWLINE>' , '}' , '<NEWLINE>' , '}' , '<NEWLINE>' , '}' , '<NEWLINE>']
['class' , '<CLASSNAME>' , '{' , '}' , '<NEWLINE>']
['public' , 'enum' , '<VARIABLENAME>' , '{' , '<NEWLINE>' , 'FemaleFirst' ,
'<NEWLINE>' , 'MaleFirst' , '<NEWLINE>' , 'LastNames' , '<NEWLINE>' ,
'<VARIABLENAME>' , '<NEWLINE>' , '}' , '<NEWLINE>']
['namespace' , '<NAMESPACE>' , '<NEWLINE>' , '{' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , '<ATTRIBUTE>' , '<NEWLINE>' , 'public' , 'class' ,
'<CLASSNAME>' , '<NEWLINE>' , '{' , '<NEWLINE>' , '}' , '<NEWLINE>' , '}' ]
['using' , '<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'using' ,
'<LIBRARY>' , '<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'namespace' ,
'<NAMESPACE>' , '<NEWLINE>' , '{' , '<NEWLINE>' , 'class' , '<CLASSNAME>' ,
'<NEWLINE>' , '{' , '<NEWLINE>' , '<VARIABLENAME>' , '<VARIABLENAME>' , ';' ,
'<NEWLINE>' , '<VARIABLENAME>' , '<VARIABLENAME>' , ';' , '<NEWLINE>' ,
'<VARIABLENAME>' , '<VARIABLENAME>' , ';' , '<NEWLINE>' , '<VARIABLENAME>' ,
'<VARIABLENAME>' , ';' , '<NEWLINE>' , 'private' , 'static' , 'extern' , 'void' ,
'<METHOD_EXEC:' , '(<PARAM>' , '<PARAM>' , '<PARAM>' , '<PARAM>' , '<PARAM>)>' ,
'<NEWLINE>' , '}' , '<NEWLINE>' , '}' , '<NEWLINE>']
['<NEWLINE>' , 'using' , '<LIBRARY>' , '<NEWLINE>' , 'namespace' , '<NAMESPACE>' ,
'<NEWLINE>' , '{' , '<NEWLINE>' , 'public' , 'static' , 'class' , '<CLASSNAME>' ,
'<NEWLINE>' , '{' , '<NEWLINE>' , 'internal' , 'static' , '<VARIABLENAME>' ,
'<VARIABLENAME>' , '{' , 'get;' , 'private' , 'set;' , '}' , '<NEWLINE>' , 'internal' ,
'static' , '<VARIABLENAME>' , '{' , 'get;' , 'set;' , '}' ,

```

```

'<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '<VARIABLENAME>', '=', 'new',
'<METHOD_EXEC:', '(<PARAM>>;', '<NEWLINE>', '}',
'<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '}',
'<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '}',
'<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', 'if', '(', '<VARIABLENAME>',
')', '<NEWLINE>', '{', '<NEWLINE>', '<CLASSMETHODINVOCATION>', ';', '<NEWLINE>',
'<VARIABLENAME>', '=', 'false;', '<NEWLINE>', '}', '<NEWLINE>', '}',
'<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>']
['if', '(', '<OBJECTPROPERTYACCESS>', '==', '<NUMBERVALUE>', ')', '<NEWLINE>',
'{', '<NEWLINE>', 'return', '<VARIABLENAME>', ';', '<NEWLINE>', '}']
[]
['private', 'class', '<CLASSNAME>', '<NEWLINE>', '{', '<NEWLINE>', 'public',
'<VARIABLENAME>', '<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<NEWLINE>',
'public', '<VARIABLENAME>', '<VARIABLENAME>', '=', '<NUMBERVALUE>', ';',
'<NEWLINE>', 'public', '<VARIABLENAME>', '<VARIABLENAME>', '=', '<NUMBERVALUE>',
';', '<NEWLINE>', 'private', '<VARIABLENAME>', '<VARIABLENAME>', '=',
'<NUMBERVALUE>', '.05f;', '<NEWLINE>', '}']
['<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>',
'<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'namespace', '<NAMESPACE>',
'<NEWLINE>', '{', '<NEWLINE>', 'public', 'class', '<CLASSNAME>', ':',
'<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>', 'private', 'readonly',
'<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>', 'private', 'readonly',
'<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}',
'<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', '<OBJECTPROPERTYACCESS>', '=',
'<NUMBERVALUE>', ';', '<NEWLINE>', 'return', '<VARIABLENAME>', ';', '<NEWLINE>',
'}']
['private', '<VARIABLENAME>', ';', '<NEWLINE>', 'public', '<VARIABLENAME>', '{',
'<NEWLINE>', 'get', '{', 'return', '<VARIABLENAME>', ';', '}', '<NEWLINE>',
'internal', 'set', '{', '<NEWLINE>', '<VARIABLENAME>', '=', '<VARIABLENAME>',
';', '<NEWLINE>', 'if', '(', '<VARIABLENAME>', '!=', '<VARIABLENAME>', ')', '{',
'<NEWLINE>', 'PropertyChanged(this,', '<CLASSINIT>', '(', '<STRINGVALUE>',
'));', '<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>', '}']
['class', '<CLASSNAME>', '<NEWLINE>', '{', '<METHOD_DEF:LENGHT_GTT:True>', '{',
'<NEWLINE>', 'RunTests().<METHOD_EXEC:', '()', '<NEWLINE>',
'<CLASSMETHODINVOCATION>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}']
['private', 'readonly', '<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>']
['<ATTRIBUTE>', '<NEWLINE>', 'public', '<VARIABLENAME>', '{', 'get;', 'set;',
'}']
[]
['<NEWLINE>', 'public', 'interface', '<VARIABLENAME>', '<NEWLINE>', '{',
'<NEWLINE>', '#region', '<VARIABLENAME>', '<NEWLINE>', '<VARIABLENAME>',
'<NEWLINE>', '{', '<NEWLINE>', 'get;', '<NEWLINE>', 'set;', '<NEWLINE>', '}',

```

```

'<NEWLINE>', '<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>', 'get;',
'<NEWLINE>', 'set;', '<NEWLINE>', '}', '<NEWLINE>', '#endregion', '<NEWLINE>',
'}']
['namespace', '<NAMESPACE>', '<NEWLINE>', '{', '<NEWLINE>', '<VARIABLENAME>',
'class', '<CLASSNAME>', '<NEWLINE>', '{', '<NEWLINE>', '}', '<NEWLINE>', '}',
'<NEWLINE>']
['<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', 'if', '(!disposed)',
'<NEWLINE>', '{', '<NEWLINE>', 'if', '(', '<VARIABLENAME>', ')', '<NEWLINE>',
'{', '<NEWLINE>', '}', '<NEWLINE>', '<CLASSMETHODINVOCATION>', ';', '<NEWLINE>',
'<VARIABLENAME>', '=', 'true;', '<NEWLINE>', '}', '<NEWLINE>', '}']
['<ATTRIBUTE>', '<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>',
'<VARIABLENAME>', '<VARIABLENAME>', '=', '<CLASSINIT>', '().',
'<CLASSMETHODINVOCATION>', ';', '<NEWLINE>', '<VARIABLENAME>', '<VARIABLENAME>',
'=', '<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', '<OBJECTPROPERTYACCESS>',
'=', '<OBJECTPROPERTYACCESS>', ';', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', '<VARIABLENAME>', '=', 'true;',
'<NEWLINE>', '<CLASSMETHODINVOCATION>', ';', '<NEWLINE>', '}']
['<NEWLINE>', 'public', '<VARIABLENAME>', ';', '<NEWLINE>', 'public',
'<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>', 'get', '<NEWLINE>', '{',
'<NEWLINE>', 'return', '<VARIABLENAME>', ';', '<NEWLINE>', '}', '<NEWLINE>',
'set', '<NEWLINE>', '{', '<NEWLINE>', 'Set()', '=>', 'HTMLContent',
'<VARIABLENAME>', '_HTMLContent', '<VARIABLENAME>', ');', '<NEWLINE>', '}',
'<NEWLINE>', '}', '<NEWLINE>']
['using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'using',
'<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>', 'namespace',
'<NAMESPACE>', '<NEWLINE>', '{', '<NEWLINE>', 'public', 'class', '<CLASSNAME>',
':', '<VARIABLENAME>', '<NEWLINE>', '{', '<NEWLINE>', 'private', 'readonly',
'<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>', 'private', 'readonly',
'<VARIABLENAME>', '<VARIABLENAME>', ';', '<NEWLINE>', 'private', 'readonly',
'<VARIABLENAME>', ';', '<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>',
'<VARIABLENAME>', '=', '<VARIABLENAME>', ';', '<NEWLINE>', '<VARIABLENAME>',
'=', '<VARIABLENAME>', ';', '<NEWLINE>', '<VARIABLENAME>', '=',
'<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}']
'<NEWLINE>', '}']
['delegate', 'void', '<METHOD_EXEC:', '(<PARAM>)>;', '<NEWLINE>', 'delegate',
'void', '<METHOD_EXEC:', '()>;', '<NEWLINE>', 'delegate', 'void',
'<METHOD_EXEC:', '(<PARAM>,<PARAM>)>;', '<NEWLINE>', 'delegate', 'void',
'<METHOD_EXEC:', '(<PARAM>)>;', '<NEWLINE>', 'delegate', 'void',
'<METHOD_EXEC:', '(<PARAM>)>;', '<NEWLINE>', 'delegate', 'void',
'<METHOD_EXEC:', '()>;', '<NEWLINE>', 'delegate', 'void', '<METHOD_EXEC:',
'(<PARAM>,<PARAM>)>;', '<NEWLINE>', 'delegate', 'void', '<METHOD_EXEC:',
'()>;', '<NEWLINE>', 'delegate', 'void', '<METHOD_EXEC:', '(<PARAM>,<PARAM>)>;',
'<NEWLINE>', 'delegate', 'void', '<METHOD_EXEC:', '()>;', '<NEWLINE>', 'delegate', 'void', '<METHOD_EXEC:', '()>;']

```

```
[ '{', '<NEWLINE>', '<VARIABLENAME>', '=', '<CLASSMETHODINVOCATION>', ';',
'<NEWLINE>' ]
[ 'using', '<LIBRARY>', '<NEWLINE>', 'using', '<LIBRARY>', '<NEWLINE>',
'<ATTRIBUTE>', '<NEWLINE>', '<ATTRIBUTE>', '<NEWLINE>', '<ATTRIBUTE>',
'<NEWLINE>', '<ATTRIBUTE>', '<NEWLINE>', '<ATTRIBUTE>', '<NEWLINE>',
'<ATTRIBUTE>', '<NEWLINE>', '<ATTRIBUTE>', '<NEWLINE>', '<ATTRIBUTE>',
'<NEWLINE>', '<ATTRIBUTE>', '<NEWLINE>', '<ATTRIBUTE>', '<NEWLINE>',
'<ATTRIBUTE>', '<NEWLINE>', '<ATTRIBUTE>', '<NEWLINE>' ]
[ '<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<VARIABLENAME>', '=', 'new',
'<METHOD_EXEC:', '(<PARAM>,<PARAM>)>;', '<NEWLINE>', '<VARIABLENAME>', '=',
'<CLASSINIT>', '(_repository,<pathResolver>,<EXTERNALCLASSREF>,<
'<CLASSINIT>', '(pathResolver,<EXTERNALCLASSREF>));', '<NEWLINE>',
'<OBJECTPROPERTYACCESS>', '=', '<VARIABLENAME>', ';', '<NEWLINE>', 'return',
'<VARIABLENAME>', ';', '<NEWLINE>', '}', '<NEWLINE>' ]
[ '<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'if', '(',
'<OBJECTMETHODINVOCATION>', ')', '<NEWLINE>', '{', '<NEWLINE>', 'return',
'false;', '<NEWLINE>', '}' ]
[ '<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<STRINGVALUE>', ';', '<NEWLINE>', 'foreach', '(', '<VARIABLENAME>', 'in',
'<VARIABLENAME>', ')', '<NEWLINE>', '{', '<NEWLINE>', '<VARIABLENAME>', '+=',
'<VARIABLENAME>', ';', '<NEWLINE>', '}', '<NEWLINE>', 'return',
'<VARIABLENAME>', ';', '<NEWLINE>', '}', '<NEWLINE>' ]
[ '<VARIABLENAME>', '=', '<STRINGVALUE>', ';', '<NEWLINE>', 'for', '(',
'<VARIABLENAME>', '=', '<NUMBERVALUE>', ';', '<VARIABLENAME>', '<',
'<NUMBERVALUE>', ';', '<VARIABLENAME>', '++', '<NEWLINE>', '{', '<NEWLINE>',
'<VARIABLENAME>', '+=', '<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', '}',
'<NEWLINE>' ]
[ '<VARIABLENAME>', '=', '<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>' ]
[ '<VARIABLENAME>', '=', '<STRINGVALUE>', '+', '<VARIABLENAME>', '+',
'<STRINGVALUE>', '+', '<VARIABLENAME>', '+', '<STRINGVALUE>', ';', '<NEWLINE>' ]
[ '<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<STRINGVALUE>', ';', '<NEWLINE>', 'foreach', '(', '<VARIABLENAME>', 'in',
'<VARIABLENAME>', ')', '<NEWLINE>', '{', '<NEWLINE>', 'if', '(',
'<VARIABLENAME>', '%', '<NUMBERVALUE>', '==', '<NUMBERVALUE>', ')', '<NEWLINE>',
'<VARIABLENAME>', '+=', '<VARIABLENAME>', '+', '<STRINGVALUE>', ';',
'<NEWLINE>', '}', '<NEWLINE>', 'return', '<VARIABLENAME>', ';', '<NEWLINE>',
'}', '<NEWLINE>' ]
[ 'public', 'class', '<CLASSNAME>', '<NEWLINE>', '{',
'<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'return', '$',
'<STRINGVALUE>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}' ]
[ 'public', 'class', '<CLASSNAME>', '<NEWLINE>', '{',
'<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'return',
'<OBJECTPROPERTYACCESS>', '>', '<NUMBERVALUE>', '&&', '<OBJECTPROPERTYACCESS>',
'>', '<NUMBERVALUE>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>' ]
[ 'public', 'class', '<CLASSNAME>', '<NEWLINE>', '{',
'<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'return', '$',
'<STRINGVALUE>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}' ]
[ 'public', 'class', '<CLASSNAME>', '<NEWLINE>', '{', '<NEWLINE>', 'private',
```

```

'readonly', '<VARIABLENAME>', ';;', '<METHOD_DEF:LENGHT_GTT:False>', '{',
'<NEWLINE>', '}', '<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', 'return',
'<OBJECTPROPERTYACCESS>', '<', '<OBJECTPROPERTYACCESS>', ';;', '<NEWLINE>', '}',
'<NEWLINE>', '}' ]
['public', 'class', '<CLASSNAME>', '<NEWLINE>', '{', '<NEWLINE>', 'public',
'<VARIABLENAME>', '{', 'get;', 'set;', '}', '<METHOD_DEF:LENGHT_GTT:True>', '{',
'<NEWLINE>', 'return', '<OBJECTPROPERTYACCESS>', '+', '<OBJECTPROPERTYACCESS>',
';', '<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>']
['public', 'class', '<CLASSNAME>', '<NEWLINE>', '{',
'<METHOD_DEF:LENGHT_GTT:False>', '{', '<NEWLINE>', 'return', '$',
'<STRINGVALUE>', ';;', '<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>', 'public',
'class', '<CLASSNAME>', '<NEWLINE>', '{', '<NEWLINE>', 'public',
'<VARIABLENAME>', '{', 'get;', 'set;', '}', '<NEWLINE>', 'public',
'<VARIABLENAME>', '{', 'get;', 'set;', '}', '<NEWLINE>', '}' ]
['public', 'class', '<CLASSNAME>', '<NEWLINE>', '{',
'<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<CLASSMETHODINVOCATION>',
';', '<NEWLINE>', '<CLASSMETHODINVOCATION>', ';;', '<NEWLINE>', '}', '<NEWLINE>',
'<NEWLINE>', '<VARIABLENAME>', '=', 'new', '<METHOD_EXEC:', '()'>;',
'<NEWLINE>', '<OBJECTMETHODINVOCATION>', ';;', '<NEWLINE>']
['public', 'class', '<CLASSNAME>', '<NEWLINE>', '{',
'<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<CLASSMETHODINVOCATION>',
';', '<NEWLINE>', '<CLASSMETHODINVOCATION>', ';;', '<NEWLINE>',
'<CLASSMETHODINVOCATION>', ';;', '<NEWLINE>', '}', '<NEWLINE>', '}', '<NEWLINE>']
['public', 'class', '<CLASSNAME>', '<NEWLINE>', '{', '<NEWLINE>', 'private',
'readonly', '<VARIABLENAME>', '<VARIABLENAME>', ';;',
'<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'<VARIABLENAME>', ';;', '<NEWLINE>', '}', '<METHOD_DEF:LENGHT_GTT:True>', '{',
'<NEWLINE>', 'if', '(', '<VARIABLENAME>', '<', '<NUMBERVALUE>', '||',
'<VARIABLENAME>', '<', '<NUMBERVALUE>', ')', '<NEWLINE>', 'throw', 'new',
'<METHOD_EXEC:', '(<PARAM>>;', '<NEWLINE>', '<VARIABLENAME>', '=',
'<CLASSINIT>', '<NEWLINE>', '{', '<NEWLINE>', '<VARIABLENAME>', '=',
'accountNumber,', '<NEWLINE>', '<VARIABLENAME>', '=', 'transactionDate,',
'<NEWLINE>', '<VARIABLENAME>', '=', 'description,', '<NEWLINE>',
'<VARIABLENAME>', '=', 'debitAmount,', '<NEWLINE>', '<VARIABLENAME>', '=',
'creditAmount,', '<NEWLINE>', '<VARIABLENAME>', '=', 'currency,', '<NEWLINE>',
'<VARIABLENAME>', '=', 'transactionType,', '<NEWLINE>', '<VARIABLENAME>', '=',
'<VARIABLENAME>', '<NEWLINE>', '};', '<NEWLINE>', '<PRIVATEVARIABLE>',
'<CLASSMETHODINVOCATION>', ';;', '<NEWLINE>', '<PRIVATEVARIABLE>',
'<METHOD_EXEC:', '()'>;', '<NEWLINE>', '}', '<NEWLINE>', '}' ]
['<METHOD_DEF:LENGHT_GTT:True>', '{', '<NEWLINE>', 'if', '(', '<VARIABLENAME>',
'>=', '<VARIABLENAME>', ')', '<NEWLINE>', 'throw', 'new', '<METHOD_EXEC:',
'(<PARAM>>;', '<NEWLINE>', 'if', '(', '<VARIABLENAME>', '==', '<VARIABLENAME>',
'||', '<OBJECTPROPERTYACCESS>', '==', '<NUMBERVALUE>', ')', '<NEWLINE>',
'throw', 'new', '<METHOD_EXEC:', '(<PARAM>>;', '<NEWLINE>', 'if', '(',
'<OBJECTMETHODINVOCATION>', ')', '<NEWLINE>', 'throw', 'new', '<METHOD_EXEC:',
'(<PARAM>>;', '<NEWLINE>', 'if', '(', '<OBJECTMETHODINVOCATION>', ')',
'<NEWLINE>', 'throw', 'new', '<METHOD_EXEC:', '(<PARAM>>;', '<NEWLINE>', 'if',
'(', '<OBJECTMETHODINVOCATION>', ')', '<NEWLINE>', 'throw', 'new',

```



```
'<METHOD_EXEC:', '(<PARAM>>';', '<NEWLINE>', '<VARIABLENAME>', '=',
'<CLASSINIT>', '<string>()';', '<NEWLINE>', 'if', '(includeTransactions)',
'<NEWLINE>', '{', '<NEWLINE>', 'foreach', '(', '<VARIABLENAME>', 'in',
'<VARIABLENAME>', ')', '<NEWLINE>', '{', '<NEWLINE>',
'<OBJECTMETHODINVOCATION>', ';', '<NEWLINE>', '}', '<NEWLINE>', '}',
'<NEWLINE>', '<VARIABLENAME>', '=', '$', '<STRINGVALUE>', ';', '<NEWLINE>',
'<VARIABLENAME>', '=', '<VARIABLENAME>', '?', '<STRINGVALUE>', ':',
'<STRINGVALUE>', ';', '<NEWLINE>', 'return', '<CLASSINIT>', '<NEWLINE>', '{',
'<NEWLINE>', '<VARIABLENAME>', '=', 'title', '<NEWLINE>', '<VARIABLENAME>',
'=', 'startDate', '<NEWLINE>', '<VARIABLENAME>', '=', 'endDate', '<NEWLINE>',
'<VARIABLENAME>', '=', 'currency', '<NEWLINE>', '<VARIABLENAME>', '=',
'approverName', '<NEWLINE>', '<VARIABLENAME>', '=', '<OBJECTPROPERTYACCESS>',
',', '<NEWLINE>', '<VARIABLENAME>', '=', 'transactions', '<NEWLINE>',
'<VARIABLENAME>', '=', 'status', '<NEWLINE>', '<VARIABLENAME>', '=',
'<VARIABLENAME>', '<NEWLINE>', '};', '<NEWLINE>', '}', '<NEWLINE>']
```

[7]: # 5. Treinar gramática com Word2Vec

```
# Número de cores da máquina
cores = multiprocessing.cpu_count()

print(f"Número de cores: {cores}")

# Treinar o modelo
model = gensim.models.Word2Vec(
    vector_size=200,
    sg=1,
    workers=cores-3,
    window=20,
    seed=42,
    alpha=0.01,
    min_alpha=0.005)

model.build_vocab(x_tokenized, progress_per=10000)

model.train(x_tokenized, total_examples=model.corpus_count, epochs=20)

# Verificar similaridade
similarity_by_key_sample = model.wv.similar_by_key("public")
print(f"Exemplo de similaridade: + {similarity_by_key_sample}")

most_similar_sample = model.wv.similar_by_key("<METHOD_DEF:LENGHT_GTT:True>")
print(f"Exemplo de similaridade: + {most_similar_sample}")

# Número total de palavras do corpus
print(f"Número total de palavra do corpus: {model.corpus_total_words}")
```

Número de cores: 22

```
Exemplo de similaridade: + [('static', 0.9710027575492859), ('readonly',
0.965536892414093), ('private', 0.960656464099884), ('class',
0.9553068280220032), ('<CLASSNAME>', 0.9479354023933411), ('get;',
0.9467548727989197), ('protected', 0.9380561113357544),
('<METHOD_DEF:LENGHT_GTT:False>', 0.934917688369751), (':', 0.9297769069671631),
('set;', 0.9275515079498291)]
Exemplo de similaridade: + [('return', 0.9902473092079163), ('get',
0.9789010882377625), ('<CLASSMETHODINVOCATION>', 0.9636496901512146), ('false;',
0.9599233865737915), ('$ ', 0.949687123298645), ('-', 0.9341035485267639), ('!=',
0.9328575134277344), ('true;', 0.9291867613792419), ('<OBJECTMETHODINVOCATION>',
0.9246307015419006), (');', 0.9141374826431274)]
Número total de palavra do corpus: 3618
```

```
[8]: # 6. Criação dos vetores a partir do vocabulário
sample_vec = eb.textToVector(model.wv, "<METHOD_DEF:LENGHT_GTT:True> a", 200,
↪20)

[print(x) for x in sample_vec]

x_vecs = np.asarray([eb.textToVector(model.wv, " ".join(x), 200, 20) for x in
↪x_tokenized])
print(x_vecs.shape)

# Reduzir dimensão para treinamento do modelo:
pca_model = PCA(n_components=40)
pca_model.fit(x_vecs)
print("Taxa de variância: ", sum(pca_model.explained_variance_ratio_))

x_vecs = pca_model.transform(x_vecs)
x_vecs.shape
```

```
0.10540489852428436
0.08899611234664917
-0.007255664560943842
0.033143118023872375
-0.08103179931640625
0.08480826020240784
-0.0677221268415451
0.22409608960151672
-0.10810153931379318
-0.08014736324548721
-0.06087903678417206
0.04685894772410393
0.0034263217821717262
0.047304071485996246
-0.044142164289951324
-0.0687420666217804
-0.027599455788731575
```

-0.13089002668857574
-0.004449022002518177
-0.08552613109350204
0.0457443930208683
0.0507148802280426
0.07013718783855438
0.1088905856013298
0.07325717806816101
0.15723775327205658
0.028683437034487724
0.2376059889793396
0.005300899967551231
-0.11328337341547012
0.018895097076892853
-0.09092870354652405
-0.10494343191385269
-0.09191782772541046
-0.012645669281482697
-0.04958074539899826
0.06323317438364029
-0.0718737319111824
-0.09374599158763885
0.14969129860401154
0.07670287787914276
0.03668602183461189
0.1541939228773117
-0.13021618127822876
-0.02363527938723564
-0.04216143861413002
-0.011150466278195381
0.056124188005924225
0.1414034515619278
-0.0909084603190422
0.010876004584133625
-0.07992618530988693
-0.08862525969743729
-0.05127156898379326
-0.0949234887957573
-0.1525212824344635
0.09710077196359634
0.07104834914207458
-0.11722755432128906
0.034879185259342194
0.035160552710294724
0.07273440062999725
-0.11335406452417374
0.05214746296405792
0.009618214331567287

-0.11782445013523102
0.09698266535997391
2.7952075470238924e-05
-0.009434536099433899
0.00434051314368844
-0.00898045301437378
-0.05511792004108429
0.009642833843827248
-0.11072264611721039
0.06212245672941208
-0.11251037567853928
-0.0006976859294809401
-0.16249984502792358
0.04940810054540634
0.07243259251117706
0.058021385222673416
-0.11174342036247253
-0.01757698692381382
-0.04413675516843796
-0.11139596253633499
0.13552822172641754
-0.029377656057476997
-0.08751173317432404
-0.02150719426572323
-0.1055930107831955
-0.0159902460873127
0.08460839837789536
-0.0476415790617466
-0.034113772213459015
0.010462031699717045
0.07277660071849823
-0.0005277703749015927
0.13670067489147186
-0.07969294488430023
0.043438415974378586
0.024716749787330627
-0.13290905952453613
0.030362235382199287
-0.14995278418064117
-0.0773530974984169
-0.0817955732345581
-0.0031691882759332657
0.03979821130633354
-0.04297386482357979
0.03714076802134514
-0.14261780679225922
0.03437638655304909
0.12515029311180115

-0.07265599817037582
0.01750962808728218
0.05294365808367729
-0.07939837127923965
-0.1310189813375473
-0.002006816677749157
-0.06923814862966537
-0.058619964867830276
0.04379081353545189
0.057767175137996674
0.031318455934524536
0.11833199858665466
-0.18444561958312988
-0.061841756105422974
-0.10913275927305222
-0.14153216779232025
0.06143588200211525
-0.09475693851709366
-0.018251899629831314
0.09706225246191025
0.09869874268770218
0.10692460834980011
-0.09492030739784241
-0.07063708454370499
-0.18527361750602722
0.034507617354393005
0.02465207502245903
0.07381260395050049
-0.023114146664738655
0.0003321188560221344
-0.00469176284968853
-0.060817983001470566
0.067456915974617
-0.02571287751197815
0.12592780590057373
0.051827725023031235
-0.07348627597093582
-0.010495493188500404
-0.02722213603556156
0.02861623466014862
0.11786514520645142
0.14342521131038666
-0.03207091987133026
-0.1083911880850792
0.12683837115764618
0.07670167088508606
0.09868280589580536
0.00021547911455854774

-0.007480031345039606
0.001134776626713574
-0.01692402921617031
0.041332341730594635
0.003318696515634656
-0.04949476197361946
-0.15778748691082
-0.0713082030415535
-0.07330045849084854
0.058252230286598206
0.045701634138822556
-0.042351022362709045
0.021833719685673714
0.07482776790857315
-0.07374756038188934
0.10873977094888687
-0.08896314352750778
-0.021582752466201782
0.01688813418149948
0.0760854035615921
0.00950786005705595
-0.07174094766378403
0.12413357198238373
0.011599387973546982
-0.04077285900712013
-0.03875117376446724
-0.0517827570438385
0.08641645312309265
-0.025473227724432945
-0.11974393576383591
0.0051786270923912525
0.016081301495432854
-0.06369215250015259
0.09333264082670212
-0.12733836472034454
0.16406561434268951
0.050443969666957855
-0.014964147470891476
0.04537796974182129
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Taxa de variância: 0.9655063848340294

[8]: (115, 40)

```
[9]: vocabulary = list(model.wv.index_to_key)
      print(vocabulary)

      model.save("./outputs/word2vec_model.model")
```

```
['<NEWLINE>', '<VARIABLENAME>', ';', '{', '}', '=', 'using', '<LIBRARY>',
 '<OBJECTPROPERTYACCESS>', 'public', '<STRINGVALUE>', '<METHOD_EXEC>',
 '<NUMBERVALUE>', '(', ')', 'private', '<METHOD_DEF:LENGHT_GTT:True>', 'if',
 'class', '<CLASSNAME>', 'return', '<CLASSMETHODINVOCATION>',
 '<METHOD_DEF:LENGHT_GTT:False>', '<OBJECTMETHODINVOCATION>', '(<PARAM>)>;',
 'get;', '<ATTRIBUTE>', 'new', 'static', '()>;', '<PARAM>', '+', '+=',
 '<CLASSINIT>', '<NAMESPACE>', 'namespace', '(<PARAM>,<PARAM>)>;', 'void',
 'readonly', ',', 'set;', '=>', '<PRIVATEVARIABLE>', 'throw', 'internal',
 'delegate', 'false;', 'for', 'protected', 'in', '==', 'true;', '<ARRAYACCESS>',
 '<,<-,<>', 'foreach', ':', '!=', ')>;', 'get', '$']
```

```
[10]: # 7. Treinar SVM com GridSearchCV
X_train, X_test, y_train, y_test = train_test_split(
    x_vecs,
    labels,
    test_size=0.3,
    random_state=72,
    stratify=labels)

scaler = StandardScaler()

X_train_scaled = scaler.fit_transform(X_train)

X_test_scaled = scaler.transform(X_test)

clf = svm.SVC()

param_grid = {
    'C': [0.1, 1, 10], # Regularização
    'kernel': ['linear', 'rbf'], # Função de kernel
    'gamma': ['scale', 'auto'], # Parâmetro do kernel
}

grid_search = GridSearchCV(estimator=clf, param_grid=param_grid, cv=2,
    ↪scoring='f1_macro')

# Treinando o modelo com Grid Search
grid_search.fit(X_train_scaled, y_train)
```

```

# Melhor combinação de hiperparâmetros encontrada
best_model = grid_search.best_estimator_
y_pred = best_model.predict(X_test_scaled)

print(f"Melhores parâmetros: {grid_search.best_params_}")

print(classification_report(y_test, y_pred, zero_division=0))

```

```

Melhores parâmetros: {'C': 0.1, 'gamma': 'scale', 'kernel': 'linear'}

```

	precision	recall	f1-score	support
0	0.77	0.92	0.84	26
1	0.50	0.25	0.33	4
2	0.50	0.50	0.50	2
3	0.00	0.00	0.00	3
accuracy			0.74	35
macro avg	0.44	0.42	0.42	35
weighted avg	0.66	0.74	0.69	35

```

[11]: # 8. Treinar RandomForest com GridSearchCV
X_train, X_test, y_train, y_test = train_test_split(x_vecs, labels, test_size=0.
↳ 3, random_state=72, stratify=labels)

# Instanciando o classificador Random Forest
rf = RandomForestClassifier(random_state=42)

# Definir os parâmetros para o GridSearch
param_grid = {
    'n_estimators': [10, 50, 100, 200],          # Número de árvores na floresta
    'max_depth': [None, 10, 20, 30],             # Profundidade máxima das árvores
    'min_samples_split': [2, 5, 10],             # Mínimo de amostras para
↳ dividir um nó
    'min_samples_leaf': [1, 2, 4],               # Mínimo de amostras por folha
    'bootstrap': [True, False],                 # Usar amostragem bootstrap
}

# Configurar GridSearchCV
grid_search = GridSearchCV(estimator=rf, param_grid=param_grid,
    cv=2, scoring='f1_macro')

# Treinando com o GridSearch
grid_search.fit(X_train, y_train)

# Melhor combinação de hiperparâmetros encontrada
best_model = grid_search.best_estimator_

```

```

y_pred = best_model.predict(X_test)

print(f"Melhores parâmetros: {grid_search.best_params_}")

print(classification_report(y_test, y_pred, zero_division=0))

```

Melhores parâmetros: {'bootstrap': False, 'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100}

	precision	recall	f1-score	support
0	0.76	1.00	0.87	26
1	0.00	0.00	0.00	4
2	1.00	0.50	0.67	2
3	0.00	0.00	0.00	3
accuracy			0.77	35
macro avg	0.44	0.38	0.38	35
weighted avg	0.63	0.77	0.68	35

[12]: # 9. Treinar XGBoost com GridSearchCV

```

X_train, X_test, y_train, y_test = train_test_split(x_vecs, labels, test_size=0.
↳3, random_state=72, stratify=labels)

clf = xgb.XGBClassifier(eval_metric='mlogloss')

param_grid = {
    'n_estimators': [50, 100],          # Número de árvores
    'max_depth': [3, 9],                # Profundidade máxima das árvores
    'learning_rate': [0.01],            # Taxa de aprendizado
    'subsample': [0.8],                 # Subamostragem
    'colsample_bytree': [0.8],          # Colunas usadas em cada árvore
    'gamma': [0.1],                    # Regularização para reduzir
↳overfitting
}

# Configurar GridSearchCV
grid_search = GridSearchCV(
    estimator=clf, param_grid=param_grid,
    scoring='f1_macro', cv=2)

# Treinando com o GridSearch
grid_search.fit(X_train, y_train)

# Melhor combinação de hiperparâmetros encontrada
best_model = grid_search.best_estimator_

```



```

y_pred = best_model.predict(X_test)

print(f"Melhores parâmetros: {grid_search.best_params_}")

print(classification_report(y_test, y_pred, zero_division=0))

```

```

Melhores parâmetros: {'colsample_bytree': 0.8, 'gamma': 0.1, 'learning_rate':
0.01, 'max_depth': 3, 'n_estimators': 50, 'subsample': 0.8}

```

	precision	recall	f1-score	support
0	0.79	1.00	0.88	26
1	0.00	0.00	0.00	4
2	0.00	0.00	0.00	2
3	0.50	0.33	0.40	3
accuracy			0.77	35
macro avg	0.32	0.33	0.32	35
weighted avg	0.63	0.77	0.69	35