

1. Introduction

- ① Project Purpose and Background: This project was undertaken to apply the knowledge learned in the seven weeks. The objective was to practice practical implementation based on the lessons covered.
- ② Goal: To develop a basic search engine that retrieves sentences similar to the user's query.

2. Requirements

- ① Preprocess sentences within the search target and store them in a list.
- ② Receive an input English string (query) from the user and preprocess it
- ③ Calculate the similarity between the query and sentences within the search target(Similarity is based on the count of the same "word.")
- ④ Rank the sentences based on similarity
- ⑤ Output the top 10 ranked sentences to the user from the ranked sentences

3. Design and Implementation

Implementation Details: (Detailed explanation of how each requirement was implemented, including relevant code snippets)

- ① User requirements: The system should be capable of searching for sentences similar to the user's query.
- ② Functional Requirements:
 - Preprocess sentences within the search target and store them in a list.

```
# 파일을 읽어서 토큰화된 문장 쌍을 생성하는 함수
def indexing(file_name):
    file_tokens_pairs = []
    file_tokens_pairs_lower = []
    lines = open(file_name, "r", encoding="utf8").readlines()
    for line in lines:
        tokens = preprocess(line)
        tokens_lower = preprocess(line.lower()) #preprocess함수 호출 #lower함수로 대소문자 구분 안 함
        file_tokens_pairs.append(tokens)
        file_tokens_pairs_lower.append(tokens_lower)
    return file_tokens_pairs, file_tokens_pairs_lower
```

■ Input:

- input: a file name with its path for search targets

■ Result:

- return value: a set of tokens for each sentence in the file and A set of lowercase-only tokens for each sentence in a file
- Explanation:
 - Read files in the path as a function "readlines"
 - Save the read file to the variables "file_tokens_pairs" and "file_tokens_pairs_lower" as a recurring statement (lower is lowercase)
- Receive an input English string (query) from the user and preprocess it.

```
# 2. 사용자로부터 쿼리 입력
query = input("영어 쿼리를 입력하세요.")
preprocessed_query = preprocess(query)
preprocessed_query_lower = preprocess(query.lower()) # lower 함수로 대소문자 구분 안 함
query_token_set = set(preprocessed_query)
query_token_set_lower = set(preprocessed_query_lower)

# 문장을 전처리하여 공백을 기준으로 토큰화하는 함수
def preprocess(sentence):
    preprocessed_sentence = sentence.strip().split(" ")
    return preprocessed_sentence
```

- Input: Value entered by the user
- Result: Functions that preprocess sentences to tokenize them based on spaces
- Explanation:
 - ◆ Save user input to variable "query"
 - ◆ Call function "preprocess"
 - ◆ Save preprocessed queries in lowercase and normal cases, respectively
- Calculate the similarity between the query and sentences within the search target (Similarity is based on the count of the same "word.")

```
# 쿼리와 문장 간의 유사도를 계산하는 함수
def calc_similarity(preprocessed_query, preprocessed_sentence):
    score_dict = {}
    for i in range(len(preprocessed_sentence)):
        file_token_set = set(preprocessed_sentence[i])
        all_tokens = preprocessed_query | file_token_set # file 단어 수 + query 단어 수 - 겹치는 수
        same_tokens = preprocessed_query & file_token_set # 겹치는 수
        similarity = len(same_tokens) / len(all_tokens)
        score_dict[i] = similarity # score_dict[i]에 저장
    return score_dict
```

Input:

- ◆ Preprocessed search queries in lower case and preprocessed search target statements in lower case

- Result:
 - ◆ Dictionary with file ID and similarity scores for queries and sentences
- Explanation:
 - ◆ $\text{all_token} = \text{Number of word in file} + \text{Number of word in query} - \text{Overlapping Count}$
 - ◆ $\text{same_token} = \text{Overlapping Count}$

- Rank the sentences based on similarity

4. 유사도 리스트 정렬

```
sorted_score_list = sorted(score_dict.items(), key = operator.itemgetter(1), reverse=True)
```

- Input: score_dict
- Result: sorted_score_list
- Explanation: Sort in order of similarity larger

- Output the top 10 ranked sentences to the user from the ranked sentences

- Input: This is not function. There is no input
- Result: Output the top 10 ranked sentences to the user from the ranked sentences
- Explanation: Output in order of "rank", "Index", "score", and "sentence"

4. **Testing** • (Testing : Testing involved verifying if the system provided the expected results based on user input.)

- ① Test Results for Each Functionality: (Description of test cases and corresponding screenshots for each requirement)

- Indexing

[['you''ll', 'be', 'picking', 'fruit', 'and', 'generally', 'helping', 'us', 'do', 'all', 'the', 'usual', 'farm', 'work.'], ['in', 'the', 'middle', 'ages', 'cities', 'were', 'not', 'very', 'clean', 'and', 'the', 'streets', 'were', 'filled', 'with', 'garbage.'], ['for', 'or', 'the', 'moment', 'they', 'may', 'yet', 'be', 'hiding', 'behind', 'their', 'apron', 'strings', 'but', 'sooner', 'or', 'later', 'their', 'society', 'will', 'catch', 'up', 'with', 'the', 'progressive', 'world.'], ['do', 'you', 'know', 'what', 'the', 'cow', 'answered?', 'said', 'the', 'minister.'], ['poland', 'and', 'italy', 'may', 'seem', 'like', 'very', 'different', 'countries.'], ['mr.', 'smith', 'and', 'i', 'stayed', 'the', 'whole', 'day', 'in', 'oxford.'], ['the', 'sight', 'of', 'a', 'red', 'traffic', 'signal', 'gave', 'him', 'an', 'idea.'], ['so', 'they', 'used', 'pumpkins', 'instead.'], ['2.', 'a', 'particular', 'occasion', 'of', 'state', 'of', 'affairs:'], ['they', 'might', 'not', 'offer', 'me', 'much', 'money.'], ['i'm', 'especially', 'interested', 'in', 'learning', 'horse-riding', 'skills', 'so', 'i', 'hope', 'you''ll', 'include', 'information', 'about', 'this.'], ['instead', 'the', 'devil', 'gave', 'him', 'a', 'single', 'candle', 'to', 'light', 'his', 'way', 'through', 'the', 'darkness.'], ['it', 'shines', 'over', 'the', 'sea.'], ['he', 'too', 'was', 'arrested', 'and', 'a', 'bomb', 'was', 'thrown', 'at', 'his', 'house.'], ['it', 'seems', 'that', 'the', 'high', 'temperature', 'and', 'pressure', 'on', 'the', 'star', 'made', 'its', 'carbon', 'surface', 'turn', 'to', 'diamond.'], ['the', 'pig', 'was', 'unpopular', 'while', 'the', 'cow', 'was', 'loved', 'by', 'everyone.'], ['books', 'give', 'a', 'lot', 'of', 'things', 'to', 'us.'], ['jimmy', 'and', 'timmy', 'were', 'identical', 'twins.'], ['it', 'is', 'a', 'chemical', 'that', 'causes', 'cancer.'], ['zige', 'from', 'germany', 'and', 'brazilian', 'superstars', 'ronaldo', 'and', 'roberto', 'carlos', 'belonged', 'to', 'the', 'bald', 'club.'], ['now', 'the', 'taliban', 'are', 'gone', 'and', 'things', 'have', 'begun', 'to', 'change.'], ['is', 'your', 'skin', 'clear', 'smooth', 'and', 'shining', 'with', 'health?'], ['as', 'a', 'result', 'there', 'is', 'a', 'great', 'deal', 'of', 'traffic', 'and', 'usually', 'not', 'enough', 'roads', 'and', 'most', 'of', 'the', 'roads', 'are', 'too', 'narrow.'], ['the', 'new', 'law', 'said', 't

- Receive an input English string (query) from the user and preprocess it

영어 쿼리를 입력하세요.APPLE BANANA CAT DOG
 ['APPLE', 'BANANA', 'CAT', 'DOG']
 A. ['apple', 'banana', 'cat', 'dog']

- Calculate the similarity between the query and sentences within the search target(Similarity is based on the count of the same "word.")

영어 쿼리를 입력하세요.The Sea
 [0: 0.06666666666666667, 1: 0.06666666666666667, 2: 0.04166666666666664, 3: 0.1, 4: 0.0, 5: 0.09090909090909091, 6: 0.08333333333333333, 7: 0.0, 8: 0.0, 9: 0.0, 10: 0.06666666666666667, 11: 0.16666666666666666, 12: 0.0, 13: 0.05555555555555555, 14: 0.09090909090909091, 15: 0.0, 16: 0.0, 17: 0.0, 18: 0.06666666666666667, 19: 0.08333333333333333, 20: 0.0, 21: 0.047619047619047616, 22: 0.06666666666666666, 23: 0.0, 24: 0.0, 25: 0.07142857142857142, 26: 0.0, 27: 0.07692307692307693, 28: 0.07692307692307693, 29: 0.03703703703703703, 30: 0.0, 31: 0.0, 32: 0.05263157894736842, 33: 0.0, 34: 0.07142857142857142, 35: 0.0, 36: 0.07692307692307693, 37: 0.11111111111111111, 38: 0.07142857142857142, 39: 0.0, 40: 0.047619047619047616, 41: 0.0, 42: 0.0, 43: 0.1, 44: 0.0, 45: 0.0, 46: 0.0, 47: 0.0, 48: 0.0, 49: 0.0, 50: 0.11111111111111111, 51: 0.0, 52: 0.125, 53: 0.09090909090909091, 54: 0.08333333333333333, 55: 0.0, 56: 0.16666666666666666, 57: 0.14285714285714285, 58: 0.11111111111111111, 59: 0.0, 60: 0.06666666666666667, 61: 0.0, 62: 0.0, 63: 0.0, 64: 0.0, 65: 0.0, 66: 0.0, 67: 0.05, 68: 0.0, 69: 0.04, 70: 0.0, 71: 0.0, 72: 0.08333333333333333, 73: 0.0, 74: 0.05555555555555555, 75: 0.0, 76: 0.0, 77: 0.11111111111111111, 78: 0.0, 79: 0.0, 80: 0.05555555555555555, 81: 0.0, 82: 0.08333333333333333, 83: 0.06666666666666667, 84: 0.05555555555555555, 85: 0.0, 86: 0.0, 87: 0.0, 88: 0.0, 89: 0.0, 90: 0.0, 91: 0.09090909090909091, 92: 0.04545454545454545, 93: 0.0, 94: 0.0, 95: 0.08333333333333333, 96: 0.0, 97: 0.04, 98: 0.047619047619047616, 99: 0.07692307692307693, 100: 0.0, 101: 0.0, 102: 0.05555555555555555, 103: 0.05, 104: 0.0, 105: 0.0, 106: 0.058823529411764705, 107: 0.0, 108: 0.0, 109: 0.11111111111111111, 110: 0.05555555555555555, 111: 0.0, 112: 0.0, 113: 0.05555555555555555, 114: 0.08333333333333333, 115: 0.2, 116: 0.0, 117: 0.16666666666666666, 118: 0.0, 119: 0.08333333333333333, 120: 0.09090909090909091, 121: 0.2, 122: 0.08333333333333333, 123: 0.1, 124: 0.07692307692307693, 125: 0.0, 126: 0.0, 127: 0.0, 128: 0.043478260869565216, 129: 0.058823529411764705, 130: 0.0, 131: 0.07142857142857142, 132: 0.07142857142857142, 133: 0.0, 134: 0.07692307692307693, 135: 0.08333333333333333, 136: 0.1, 137: 0.0625, 138: 0.0, 139: 0.0, 140: 0.0, 141: 0.0, 142: 0.058823529411764705, 143: 0.0, 144: 0.08333333333333333, 145: 0.04545454545454545, 146: 0.0, 147: 0.0, 148: 0.07142857142857142, 149: 0.0, 150: 0.11111111111111111, 151: 0.0625, 152: 0.11111111111111111, 153: 0.06666666666666667, 154: 0.0, 155: 0.0, 156: 0.0, 157: 0.0, 158: 0.0, 159: 0.0, 160: 0.0, 161: 0.058823529411764705, 162: 0.14285714285714285, 163: 0.0, 164: 0.0, 165: 0.125, 166: 0.0625, 167: 0.0, 168: 0.0, 169: 0.0, 170: 0.0, 171: 0.08333333333333333, 172: 0.0, 173: 0.0, 174: 0.05, 175: 0.08333333333333333, 176: 0.0, 177: 0.0, 178: 0.2, 179: 0.07142857142857142, 180: 0.0, 181: 0.0, 182: 0.0, 183: 0.11111111111111111, 184: 0.07692307692307693, 185: 0.07692307692307693, 186: 0.0, 187: 0.0, 188: 0.0, 189: 0.0, 190: 0.0, 191: 0.07142857142857142, 192: 0.07692307692307693, 193: 0.07692307692307693, 194: 0.0, 195: 0.0, 196: 0.0, 197: 0.05, 198: 0.0, 199: 0.125, 200: 0.0, 201: 0.0, 202: 0.07142857142857142, 203: 0.09090909090909091, 204: 0.0, 205: 0.0, 206: 0.0, 207: 0.06666666666666667, 208: 0.0, 209: 0.0, 210: 0.08333333333333333, 211: 0.05555555555555555, 212: 0.0, 213: 0.0, 214: 0.0, 215: 0.0, 216: 0.0, 217: 0.0, 218: 0.0, 219: 0.0, 220: 0.0, 221: 0.0, 222: 0.0, 223: 0.0, 224: 0.0, 225: 0.0, 226: 0.0, 227: 0.0, 228: 0.0, 229: 0.0, 230: 0.0, 231: 0.0, 232: 0.0, 233: 0.0, 234: 0.0, 235: 0.0, 236: 0.0, 237: 0.0, 238: 0.0, 239: 0.0, 240: 0.0, 241: 0.0, 242: 0.0, 243: 0.0, 244: 0.0, 245: 0.0, 246: 0.0, 247: 0.0, 248: 0.0, 249: 0.0, 250: 0.0, 251: 0.0, 252: 0.0, 253: 0.0, 254: 0.0, 255: 0.0, 256: 0.0, 257: 0.0, 258: 0.0, 259: 0.0, 260: 0.0, 261: 0.0, 262: 0.0, 263: 0.0, 264: 0.0, 265: 0.0, 266: 0.0, 267: 0.0, 268: 0.0, 269: 0.0, 270: 0.0, 271: 0.0, 272: 0.0, 273: 0.0, 274: 0.0, 275: 0.0, 276: 0.0, 277: 0.0, 278: 0.0, 279: 0.0, 280: 0.0, 281: 0.0, 282: 0.0, 283: 0.0, 284: 0.0, 285: 0.0, 286: 0.0, 287: 0.0, 288: 0.0, 289: 0.0, 290: 0.0, 291: 0.0, 292: 0.0, 293: 0.0, 294: 0.0, 295: 0.0, 296: 0.0, 297: 0.0, 298: 0.0, 299: 0.0, 300: 0.0, 301: 0.0, 302: 0.0, 303: 0.0, 304: 0.0, 305: 0.0, 306: 0.0, 307: 0.0, 308: 0.0, 309: 0.0, 310: 0.0, 311: 0.0, 312: 0.0, 313: 0.0, 314: 0.0, 315: 0.0, 316: 0.0, 317: 0.0, 318: 0.0, 319: 0.0, 320: 0.0, 321: 0.0, 322: 0.0, 323: 0.0, 324: 0.0, 325: 0.0, 326: 0.0, 327: 0.0, 328: 0.0, 329: 0.0, 330: 0.0, 331: 0.0, 332: 0.0, 333: 0.0, 334: 0.0, 335: 0.0, 336: 0.0, 337: 0.0, 338: 0.0, 339: 0.0, 340: 0.0, 341: 0.0, 342: 0.0, 343: 0.0, 344: 0.0, 345: 0.0, 346: 0.0, 347: 0.0, 348: 0.0, 349: 0.0, 350: 0.0, 351: 0.0, 352: 0.0, 353: 0.0, 354: 0.0, 355: 0.0, 356: 0.0, 357: 0.0, 358: 0.0, 359: 0.0, 360: 0.0, 361: 0.0, 362: 0.0, 363: 0.0, 364: 0.0, 365: 0.0, 366: 0.0, 367: 0.0, 368: 0.0, 369: 0.0, 370: 0.0, 371: 0.0, 372: 0.0, 373: 0.0, 374: 0.0, 375: 0.0, 376: 0.0, 377: 0.0, 378: 0.0, 379: 0.0, 380: 0.0, 381: 0.0, 382: 0.0, 383: 0.0, 384: 0.0, 385: 0.0, 386: 0.0, 387: 0.0, 388: 0.0, 389: 0.0, 390: 0.0, 391: 0.0, 392: 0.0, 393: 0.0, 394: 0.0, 395: 0.0, 396: 0.0, 397: 0.0, 398: 0.0, 399: 0.0, 400: 0.0, 401: 0.0, 402: 0.0, 403: 0.0, 404: 0.0, 405: 0.0, 406: 0.0, 407: 0.0, 408: 0.0, 409: 0.0, 410: 0.0, 411: 0.0, 412: 0.0, 413: 0.0, 414: 0.0, 415: 0.0, 416: 0.0, 417: 0.0, 418: 0.0, 419: 0.0, 420: 0.0, 421: 0.0, 422: 0.0, 423: 0.0, 424: 0.0, 425: 0.0, 426: 0.0, 427: 0.0, 428: 0.0, 429: 0.0, 430: 0.0, 431: 0.0, 432: 0.0, 433: 0.0, 434: 0.0, 435: 0.0, 436: 0.0, 437: 0.0, 438: 0.0, 439: 0.0, 440: 0.0, 441: 0.0, 442: 0.0, 443: 0.0, 444: 0.0, 445: 0.0, 446: 0.0, 447: 0.0, 448: 0.0, 449: 0.0, 450: 0.0, 451: 0.0, 452: 0.0, 453: 0.0, 454: 0.0, 455: 0.0, 456: 0.0, 457: 0.0, 458: 0.0, 459: 0.0, 460: 0.0, 461: 0.0, 462: 0.0, 463: 0.0, 464: 0.0, 465: 0.0, 466: 0.0, 467: 0.0, 468: 0.0, 469: 0.0, 470: 0.0, 471: 0.0, 472: 0.0, 473: 0.0, 474: 0.0, 475: 0.0, 476: 0.0, 477: 0.0, 478: 0.0, 479: 0.0, 480: 0.0, 481: 0.0, 482: 0.0, 483: 0.0, 484: 0.0, 485: 0.0, 486: 0.0, 487: 0.0, 488: 0.0, 489: 0.0, 490: 0.0, 491: 0.0, 492: 0.0, 493: 0.0, 494: 0.0, 495: 0.0, 496: 0.0, 497: 0.0, 498: 0.0, 499: 0.0, 500: 0.0, 501: 0.0, 502: 0.0, 503: 0.0, 504: 0.0, 505: 0.0, 506: 0.0, 507: 0.0, 508: 0.0, 509: 0.0, 510: 0.0, 511: 0.0, 512: 0.0, 513: 0.0, 514: 0.0, 515: 0.0, 516: 0.0, 517: 0.0, 518: 0.0, 519: 0.0, 520: 0.0, 521: 0.0, 522: 0.0, 523: 0.0, 524: 0.0, 525: 0.0, 526: 0.0, 527: 0.0, 528: 0.0, 529: 0.0, 530: 0.0, 531: 0.0, 532: 0.0, 533: 0.0, 534: 0.0, 535: 0.0, 536: 0.0, 537: 0.0, 538: 0.0, 539: 0.0, 540: 0.0, 541: 0.0, 542: 0.0, 543: 0.0, 544: 0.0, 545: 0.0, 546: 0.0, 547: 0.0, 548: 0.0, 549: 0.0, 550: 0.0, 551: 0.0, 552: 0.0, 553: 0.0, 554: 0.0, 555: 0.0, 556: 0.0, 557: 0.0, 558: 0.0, 559: 0.0, 560: 0.0, 561: 0.0, 562: 0.0, 563: 0.0, 564: 0.0, 565: 0.0, 566: 0.0, 567: 0.0, 568: 0.0, 569: 0.0, 570: 0.0, 571: 0.0, 572: 0.0, 573: 0.0, 574: 0.0, 575: 0.0, 576: 0.0, 577: 0.0, 578: 0.0, 579: 0.0, 580: 0.0, 581: 0.0, 582: 0.0, 583: 0.0, 584: 0.0, 585: 0.0, 586: 0.0, 587: 0.0, 588: 0.0, 589: 0.0, 590: 0.0, 591: 0.0, 592: 0.0, 593: 0.0, 594: 0.0, 595: 0.0, 596: 0.0, 597: 0.0, 598: 0.0, 599: 0.0, 600: 0.0, 601: 0.0, 602: 0.0, 603: 0.0, 604: 0.0, 605: 0.0, 606: 0.0, 607: 0.0, 608: 0.0, 609: 0.0, 610: 0.0, 611: 0.0, 612: 0.0, 613: 0.0, 614: 0.0, 615: 0.0, 616: 0.0, 617: 0.0, 618: 0.0, 619: 0.0, 620: 0.0, 621: 0.0, 622: 0.0, 623: 0.0, 624: 0.0, 625: 0.0, 626: 0.0, 627: 0.0, 628: 0.0, 629: 0.0, 630: 0.0, 631: 0.0, 632: 0.0, 633: 0.0, 634: 0.0, 635: 0.0, 636: 0.0, 637: 0.0, 638: 0.0, 639: 0.0, 640: 0.0, 641: 0.0, 642: 0.0, 643: 0.0, 644: 0.0, 645: 0.0, 646: 0.0, 647: 0.0, 648: 0.0, 649: 0.0, 650: 0.0, 651: 0.0, 652: 0.0, 653: 0.0, 654: 0.0, 655: 0.0, 656: 0.0, 657: 0.0, 658: 0.0, 659: 0.0, 660: 0.0, 661: 0.0, 662: 0.0, 663: 0.0, 664: 0.0, 665: 0.0, 666: 0.0, 667: 0.0, 668: 0.0, 669: 0.0, 670: 0.0, 671: 0.0, 672: 0.0, 673: 0.0, 674: 0.0, 675: 0.0, 676: 0.0, 677: 0.0, 678: 0.0, 679: 0.0, 680: 0.0, 681: 0.0, 682: 0.0, 683: 0.0, 684: 0.0, 685: 0.0, 686: 0.0, 687: 0.0, 688: 0.0, 689: 0.0, 690: 0.0, 691: 0.0, 692: 0.0, 693: 0.0, 694: 0.0, 695: 0.0, 696: 0.0, 697: 0.0, 698: 0.0, 699: 0.0, 700: 0.0, 701: 0.0, 702: 0.0, 703: 0.0, 704: 0.0, 705: 0.0, 706: 0.0, 707: 0.0, 708: 0.0, 709: 0.0, 710: 0.0, 711: 0.0, 712: 0.0, 713: 0.0, 714: 0.0, 715: 0.0, 716: 0.0, 717: 0.0, 718: 0.0, 719: 0.0, 720: 0.0, 721: 0.0, 722: 0.0, 723: 0.0, 724: 0.0, 725: 0.0, 726: 0.0, 727: 0.0, 728: 0.0, 729: 0.0, 730: 0.0, 731: 0.0, 732: 0.0, 733: 0.0, 734: 0.0, 735: 0.0, 736: 0.0, 737: 0.0, 738: 0.0, 739: 0.0, 740: 0.0, 741: 0.0, 742: 0.0, 743: 0.0, 744: 0.0, 745: 0.0, 746: 0.0, 747: 0.0, 748: 0.0, 749: 0.0, 750: 0.0, 751: 0.0, 752: 0.0, 753: 0.0, 754: 0.0, 755: 0.0, 756: 0.0, 757: 0.0, 758: 0.0, 759: 0.0, 760: 0.0, 761: 0.0, 762: 0.0, 763: 0.0, 764: 0.0, 765: 0.0, 766: 0.0, 767: 0.0, 768: 0.0, 769: 0.0, 770: 0.0, 771: 0.0, 772: 0.0, 773: 0.0, 774: 0.0, 775: 0.0, 776: 0.0, 777: 0.0, 778: 0.0, 779: 0.0, 780: 0.0, 781: 0.0, 782: 0.0, 783: 0.0, 784: 0.0, 785: 0.0, 786: 0.0, 787: 0.0, 788: 0.0, 789: 0.0, 790: 0.0, 791: 0.0, 792: 0.0, 793: 0.0, 794: 0.0, 795: 0.0, 796: 0.0, 797: 0.0, 798: 0.0, 799: 0.0, 800: 0.0, 801: 0.0, 802: 0.0, 803: 0.0, 804: 0.0, 805: 0.0, 806: 0.0, 807: 0.0, 808: 0.0, 809: 0.0, 810: 0.0, 811: 0.0, 812: 0.0, 813: 0.0, 814: 0.0, 815: 0.0, 816: 0.0, 817: 0.0, 818: 0.0, 819: 0.0, 820: 0.0, 821: 0.0, 822: 0.0, 823: 0.0, 824: 0.0, 825: 0.0, 826: 0.0, 827: 0.0, 828: 0.0, 829: 0.0, 830: 0.0, 831: 0.0, 832: 0.0, 833: 0.0, 834: 0.0, 835: 0.0, 836: 0.0, 837: 0.0, 838: 0.0, 839: 0.0, 840: 0.0, 841: 0.0, 842: 0.0, 843: 0.0, 844: 0.0, 845: 0.0, 846: 0.0, 847: 0.0, 848: 0.0, 849: 0.0, 850: 0.0, 851: 0.0, 852: 0.0, 853: 0.0, 854: 0.0, 855: 0.0, 856: 0.0, 857: 0.0, 858: 0.0, 859: 0.0, 860: 0.0, 861: 0.0, 862: 0.0, 863: 0.0, 864: 0.0, 865: 0.0, 866: 0.0, 867: 0.0, 868: 0.0, 869: 0.0, 870: 0.0, 871: 0.0, 872: 0.0, 873: 0.0, 874: 0.0, 875: 0.0, 876: 0.0, 877: 0.0, 878: 0.0, 879: 0.0, 880: 0.0, 881: 0.0, 882: 0.0, 883: 0.0, 884: 0.0, 885: 0.0, 886: 0.0, 887: 0.0, 888: 0.0, 889: 0.0, 890: 0.0, 891: 0.0, 892: 0.0, 893: 0.0, 894: 0.0, 895: 0.0, 896: 0.0, 897: 0.0, 898: 0.0, 899: 0.0, 900: 0.0, 901: 0.0, 902: 0.0, 903: 0.0, 904: 0.0, 905: 0.0, 906: 0.0, 907: 0.0, 908: 0.0, 909: 0.0, 910: 0.0, 911: 0.0, 912: 0.0, 913: 0.0, 914: 0.0, 915: 0.0, 916: 0.0, 917: 0.0, 918: 0.0, 919: 0.0, 920: 0.0, 921: 0.0, 922: 0.0, 923: 0.0, 924: 0.0, 925: 0.0, 926: 0.0, 927: 0.0, 928: 0.0, 929: 0.0, 930: 0.0, 931: 0.0, 932: 0.0, 933: 0.0, 934: 0.0, 935: 0.0, 936: 0.0, 937: 0.0, 938: 0.0, 939: 0.0, 940: 0.0, 941: 0.0, 942: 0.0, 943: 0.0, 944: 0.0, 945: 0.0, 946: 0.0, 947: 0.0, 948: 0.0, 949: 0.0, 950: 0.0, 951: 0.0, 952: 0.0, 953: 0.0, 954: 0.0, 955: 0.0, 956: 0.0, 957: 0.0, 958: 0.0, 959: 0.0, 960: 0.0, 961: 0.0, 962: 0.0, 963: 0.0, 964: 0.0, 965: 0.0, 966: 0.0, 967: 0.0, 968: 0.0, 969: 0.0, 970: 0.0, 971: 0.0, 972: 0.0, 973: 0.0, 974: 0.0, 975: 0.0, 976: 0.0, 977: 0.0, 978: 0.0, 979: 0.0, 980: 0.0, 981: 0.0, 982: 0.0, 983: 0.0, 984: 0.0, 985: 0.0, 986: 0.0, 987: 0.0, 988: 0.0, 989: 0.0, 990: 0.0, 991: 0.0, 992: 0.0, 993: 0.0, 994: 0.0, 995: 0.0, 996: 0.0, 997: 0.0, 998: 0.0, 999: 0.0, 1000: 0.0]

- Rank the sentences based on similarity
- Output the top 10 ranked sentences to the user from the ranked sentences

rank	Index	score	sentence
1	115	0.2	The parks are beautiful.
2	121	0.2	The boss replied, "Really?"
3	178	0.2	This puzzled the pig.
4	656	0.2	The boy sat down.
5	11	0.16666666666666666	It shines over the sea.
6	56	0.16666666666666666	Next comes the "horror" stage.
7	117	0.16666666666666666	Now, it's the lion's turn.
8	228	0.16666666666666666	The boy said, "Come in."
9	351	0.16666666666666666	"Do you have the time?"
10	479	0.16666666666666666	The second tried counting sheep.

A.

- ② Final Test Screenshot:: ([Comprehensive screenshot demonstrating the overall functionality of the program])

- A. If there is no similar sentence

영어 쿼리를 입력하세요.Hello
 There is no similar sentence.

- B. If there is a similar sentence

영어 쿼리를 입력하세요.The ApPLe aNd tHE TRee

rank	Index	score	sentence
1	150	0.2	The next morning the alarm rang and woke the student.
2	152	0.2	And the public will react variously to the news.
3	266	0.2	My wife and I do the housework together.
4	498	0.2	I abandoned ours and ran over to grab the tree she had set aside.
5	690	0.2	We like cooking meals and washing the dishes.
6	218	0.181818181818182	The sun keeps you from seeing the moon and stars.
7	441	0.181818181818182	So I opened the cage, and it flew away.
8	574	0.181818181818182	One day Harry and Jane went to the park.
9	5	0.16666666666666666	Mr. Smith and I stayed the whole day in Oxford.
10	452	0.16666666666666666	The shirt is 10 dollars and the pants are 15 dollars.

영어 쿼리를 입력하세요.Hello

There is no similar sentence.

5. Results and Conclusion

- ① 1. Result: Functioned search engine programs and implemented all case-insensitive requirements
- ② 2. Conclusion: I'm sorry that I can't distinguish special characters