In [1]:
```
from collections import defaultdict
```

In [2]:
```
def build_inverted_index(documents):
    inverted_index = defaultdict(list)

    for doc_id, text in documents.items():
        words = text.split()  # Tokenize the text by splitting words (basic toke
        for word in words:
            word = word.lower()  # Convert to lowercase for case insensitivity
            if doc_id not in inverted_index[word]:
                inverted_index[word].append(doc_id)

    return inverted_index
```

In [3]:
```
def retrieve_documents(query, inverted_index):
    query_words = query.split()  # Split query into words
    relevant_docs = set()

    for word in query_words:
        word = word.lower()  # Convert to lowercase
        if word in inverted_index:
            if not relevant_docs:
                relevant_docs = set(inverted_index[word])  # Initialize with fir
            else:
                relevant_docs.intersection_update(inverted_index[word])  # Perfo

    return relevant_docs if relevant_docs else "No documents match the query."
```

In [4]:
```
documents = {
    1: "Natural language processing is a field of artificial intelligence.",
    2: "Inverted indexing is used for document retrieval.",
    3: "Document retrieval is efficient using inverted index structures.",
    4: "Artificial intelligence and machine learning are popular fields in compu
}
```

In [5]:
```
inverted_index = build_inverted_index(documents)
```

In [13]:
```
query = "artificial"
relevant_docs = retrieve_documents(query, inverted_index)
```

In [14]:
```
print("\nSample Inverted Index (partial):")
for word in list(inverted_index.keys())[:10]:  # Printing a few sample entries
    print(f"'{word}': {inverted_index[word]}")
```

```
Sample Inverted Index (partial):
'natural': [1]
'language': [1]
'processing': [1]
'is': [1, 2, 3]
'a': [1]
'field': [1]
'of': [1]
'artificial': [1, 4]
'intelligence.': [1]
'inverted': [2, 3]
```

In [15]:
```
print(f"Query: '{query}'")
print(f"Relevant Documents: {relevant_docs}")
```

```
Query: 'artificial'
Relevant Documents: {1, 4}
```

In [ ]:
```

```