

Please generate a short comment in one sentence for the following function: {Code}

**Figure 1: Code Prompt (Sun et al.)**

The issue report is proposed before the code is committed to the codebase and can provide supplementary information for the code comments of the code. Please read the following issue, then based on the issue, please generate code comments for the code.

Issue: {Issue}

Code: {Code}

**Figure 2: Code-Issue Prompt (No Retrieval)**

The issue report is proposed before the code is committed to codebase and can provide supplementary information about the code to its code comments. Please read the following issue sentences searched from issue using the code, then generate code comments for the code.

Issue Sentences: {Issue Sentences}

Code: {Code}

**Figure 3: Code-Issue Retrieval Prompt**

I will give you a code snippet and an issue report. Please read the issue report sentence by sentence carefully, then please identify the sentences in the issue report that can help generate code comments for the code snippets.

Remember that you need to focus on the most relevant sentences that can provide the following aspects of supplementary information about the code snippet:

- (1) Functionality: describe what the code does in terms of functionality. For example: {Functionality Example}.
- (2) Concept: describe the meaning of variables or terms used in code. For example: {Concept Example}.
- (3) Directive: describe what are allowed or not allowed to do with the code. For example: {Directive Example}.
- (4) Rationale: describe the purpose or design rationale of the code. For example: {Rationale Example}.
- (5) Implication: describe the potential implication of the code, such as performance issues, etc. For example: {Implication Example}.

Please output information for each aspect in the following format. First, for each aspect, you need to output its name in one line, such as (Functionality). Second, for each aspect, you need to output "Relevant Sentences" in one line, then output the relevant sentences line by line starting with -. Third, for each aspect, you need to output "Code Comments" in one line, then output the corresponding aspect of code comments for the code snippet based on the identified relevant sentences in the following lines.

Issue: {Issue} Code: {Code}

**Figure 4: IsComment Prompt**

**Code:**

```
protected int getCacheDuration(){
    return 3600;
}
```

**TF-IDF Retrieval:**

- (1) It adds one additional parameter to the MIME-Multipart DataFormat.
- (2) includeHeaders: A regex that defines which Camel headers are also included as MIME headers into the MIME multipart.
- (3) public void setIncludeHeaders(String includeHeaders) {\n this.includeHeaders = includeHeaders;\n }
- (4) Option to add headers to MIME-Multipart in DataFormat
- (5) In some cases it might be useful to add Camel headers to a MIME Multipart when the MIME headers are part of the message body.

**DPR Retrieval:**

- (1) This will only work if headersInline is set to true.
- (2) includeHeaders: A regex that defines which Camel headers are also included as MIME headers into the MIME multipart.
- (3) It adds one additional parameter to the MIME-Multipart DataFormat:
- (4) In some cases it might be useful to add Camel headers to a MIME Multipart when the MIME headers are part of the message body.
- (5) Default is to include no Camel headers.

**DistilRoBERTa Retrieval:**

- (1) includeHeaders: A regex that defines which Camel headers are also included as MIME headers into the MIME multipart.
- (2) This will only work if headersInline is set to true.
- (3) Default is to include no Camel headers.
- (4) Option to add headers to MIME-Multipart in DataFormat.
- (5) In some cases it might be useful to add Camel headers to a MIME Multipart when the MIME headers are part of the message body.

**IsComment Retrieval:**

(Functionality)

Option to add headers to MIME-Multipart in DataFormat. In some cases it might be useful to add Camel headers to a MIME Multipart when the MIME headers are part of the message body. The attached patch allows to transfer header (and attachment) information with protocols that support neither headers nor attachments (e.g. FTP)

(Concept)

includeHeaders: A regex that defines which Camel headers are also included as MIME headers into the MIME multipart.

(Directive)

This will only work if headersInline is set to true.

Default is to include no Camel headers.

(Rationale)

The attached patch allows to transfer header (and attachment) information with protocols that support neither headers nor attachments (e.g. FTP).

(Implication)

No

**Figure 5: The issue content retrieved by different approaches.**