Identifier Names

. The questions in this section are regarding entities, their characteristics, and their behaviors. Please watch the following video to understand these concepts. If you have any further questions or need any clarifications, please ask the researcher after you watch the video. A transcript of the video is available at https://raw.githubusercontent.com/sgun571/Code-Review/master/Identifier%20Names.

To summarize the video, low quality names are,

- Entity /Characteristic/ Behavior names that are meaningless.
- Entity/Characteristic/Behavior names that are non-specific.
- Entity/Characteristic/Behavior names with typos.

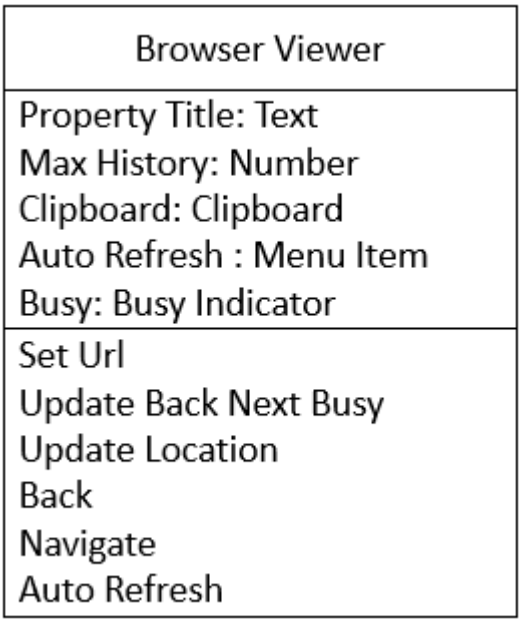
- Entity names that are not nouns or noun phrases.
- Entity names unsuitable to represent an entity.

- Characteristic names that are not nouns or noun phrases.
- Characteristic names that do not seem to belong to the entity.

- Behavior names that are not verbs or verb phrases.
- Behavior names that do not seem to belong to the entity.

Q1. **Training Question:** In the entity given below, please select the entity, characteristic, and behavior names that are of low-quality by clicking on them once (This will highlight the name in green implying the name is of low-quality).

"Browser Viewer" represents the browser window.



Q1.1. In the entity given below, please select the entity, characteristic, and behavior names that are of low-quality by clicking on them once (This will highlight the name in green implying the name is of low-quality).

An employee working in an organization.
(YTD = Year To Date)

Employee
Tax ID: Number First Name: Text Last Name: Text Employee Type: Text Start Date: Text End Date: Text Hours Worked: Number Rate: Number Deductions: Number YTD: Number String Input: List of Texts
Initialize Employee Calculate Tax Calculate Net Start Date To End Date Build Employee Summary Build Pay slip Calculate Pay

Q1.2. In the entity given below (extracted from a project named "Couchbase"), please select the entity, characteristic, and behavior names that are of low-quality by clicking on them once (This will highlight the name in green implying the name is of low-quality). This entity is extracted from the project named **"Couchbase"**.

Encoder that encodes Query Requests into lower level Http Requests as well as decoding Http Objects into Couchbase Responses.

Query Handler
Logger: Couchbase Logger Response Header: Http Response Response Content: Byte Buf Query Row Observable: Replay Subject Query Error Observable: Replay Subject Query Status Observable: Replay Subject Query Info Observable: Replay Subject Query Parsing State: Number
Bytes Before Handle Generic Query Response Parse Query Response Transition To Next Token Find End Parse Query Signature Parse Query Rows

Q1.3. In the entity given below, please select the entity, characteristic, and behavior names that are of low-quality by clicking on them once (This will highlight the name in green implying the name is of low-quality). This entity is extracted from the project named **"Egit"**.

A future http response

Http Future
Obj Ref: Atomic Reference Latch: Count Down Latch Timeout: Number Status: Operation Status Op: Http Operation Eglit Logger: Egit Logger

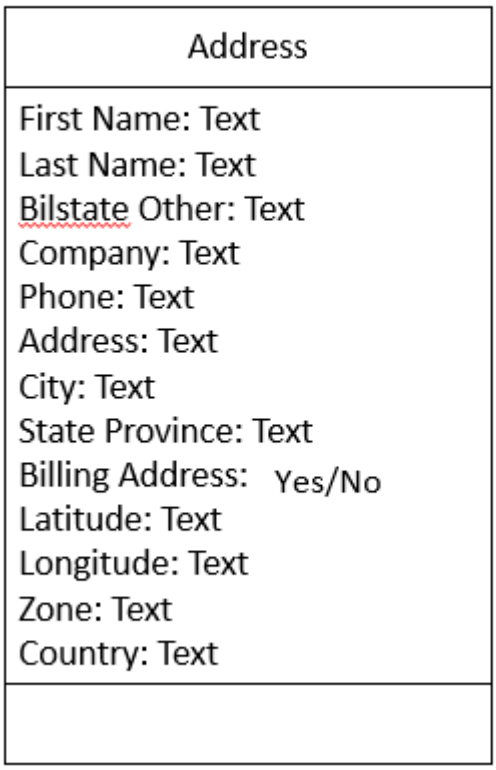
Q1.4. In the entity given below, please select the entity, characteristic, and behavior names that are of low-quality by clicking on them once (This will highlight the name in green implying the name is of low-quality). This entity is extracted from the project named **"Eclipse"**.

A label on a software interface (UI = User Interface, URI = Uniform Resource Identifier)

UI Label
Logger: Eclipse Logger Topic: Text Icon URI: Text Label: Text Tooltip: Text Busy: Text

Q1.5. In the entity given below, please select the entity, characteristic, and behavior names that are of low-quality by clicking on them once (This will highlight the name in green implying the name is of low-quality).

Address of a customer on an e-commerce software



Javadoc-Related Problems

. Please watch the following video on "Code Descriptions" before attempting the questions. The transcript of the video is available at https://raw.githubusercontent.com/sgun571/Code-Review/master/Javadocs1.

A Javadoc may have any tone: a tone of instruction, condition, statement, etc. and as long as the Javadoc describes what a block of code does, the tone does not matter.

. As you have already seen, a code description can consist of sentences, phrases or both. Now we are going to look into the **grammatical** and **typographical errors** in *code descriptions*. Please watch the following video before attempting the rest of the questions. The transcript of the video is available at https://raw.githubusercontent.com/sgun571/Code-Review/master/JavadocsGrammarTypo.

Q2.1.

Please click on the locations where you see a grammatical or typographical error in the description given below:

```
/**
 * Common interface for all DCP requests.
 *
 * Note that they can flow in both directions. For example, {@link ConnectionType#PRODUCER}
 * connection, will means that requests will flow from server to client.
 *
 * @author Sergey Avseyev
 * @since 1.1.0
 */
```

Q2.2. Please click on the locations where you see a grammatical or typographical error in the description given below:

```
/**
 * Initialize the viewer with the information received.
 *
 * @param treeID
 *          The unique ID of the tree that is returned by
 *          {@link TmfStatisticsViewer#getTreeID(String)}
 * @param traces
 *          The list of the traces to add in the tree.
 * @since 2.0
 */
```

Q2.3. Please click on the locations where you see a grammatical or typographical error in the description given below:

```
/**
 * Gets the folder that will be use as the parent of tabs that will hold the
 * viewer.
 *
 * In order to be able to add new tabs in this view, the parent of the
 * viewer control has to be this composite.
 *
 * @return the folder composite to use as the parent for the viewer control
 *         to create.
 */
```

Q2.4. Please click on the locations where you see a grammatical or typographical error in the description given below:

```
/**
 * Checks if statistic update is ongoing. If it is ongoing the new time
 * range is stored as pending
 *
 * @param timeRange
 *            - new time range
 * @return true if statistic update is ongoing else false
 */
```

Q2.5. Please click on the locations where you see a grammatical or typographical error in the description given below:

```
/**
 * This interface must be implemented to be a commit message provider.
 * A commit message provider provides the complete or a fragment of a
 * commit message. This message will be added to the text field in the
 * {@link CommitDialog}.
 * <br/>
 * Primarily use is the integration of mylyns commit templates.
 *
 * @see CommitDialog
 *
 * @author Thorsten Kamann <thorsten@itemis.de>
 * @since 0.10
 */
```

Q2.6. Please click on the locations where you see a grammatical or typographical error in the description given below:

```
/**
 * Counts the number of wait requested. It avoids removing the waiting
 * cursor, since there may be multiple requests running at the same time.
 */
```

Experiment Description

. Thank you for taking part in this research. Finally, we would like you to fill out this short demographics questionnaire which can help us to study the effect that the level of programming expertise has on the code review outcomes.

Programming expertise

Q0. Please enter your participant number below:

Q1. What is your current profession?

Q2. How will you rate your **programming expertise** according to the definitions given below?

- **I have no programming knowledge or experience.**
- **Novice**:
  - I have little or no experience
  - I want unambiguous rules to accomplish my tasks
  - I am able to handle small, isolated tasks
- **Advanced Beginner:**
  - I have gained some experience
  - I can work more independently than a novice
  - I know general principles in a limited context, but do not have a holistic understanding
- **Competent:**

- I have a holistic understanding of the problem domain
- I base my work on deliberate planning and extensive past experience
- I can apply general maxims (e.g. design patterns) easily to specific contexts

Proficient:

- I have a vast amount of experience that I can intuitively apply to new contexts
- I can easily differentiate between irrelevant and important details
- I constantly reflect on what I have done and revise my own approach to perform better in the future

Expert:

- I am a major source of knowledge and information for others
- I primarily work from my intuition