

JET
BRAINS

—

チームのコード品質を 底上げする JetBrains Qodana と GitHub Actions の統合

—

堀岡 勝 (Masaru Horioka)

JetBrains 日本担当

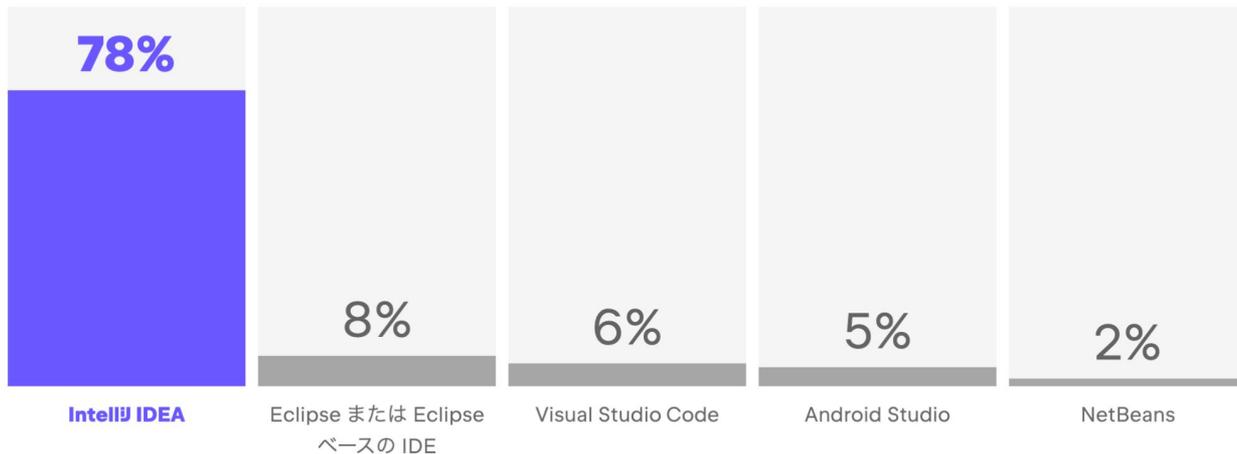
本日本話する内容

1. JetBrains 製品と静的コード解析
2. Qodana 概要
3. Qodana と GitHub Actions の統合
4. まとめ

JetBrains 製品やテクノロジー
お使いですか？

2023 開発者エコシステムの現状より(抜粋)

Java 開発ではどの IDE またはエディター
を最もよく使用していますか？



回答者を代表するような母集団を確保できるようにあらゆる対策を講じましたが、JetBrains 製品のユーザーがアンケートに回答する可能性が高かったため、結果に若干の偏りが生じている可能性があります。

SNYK JVM Ecosystem Report 2021 (抜粋)

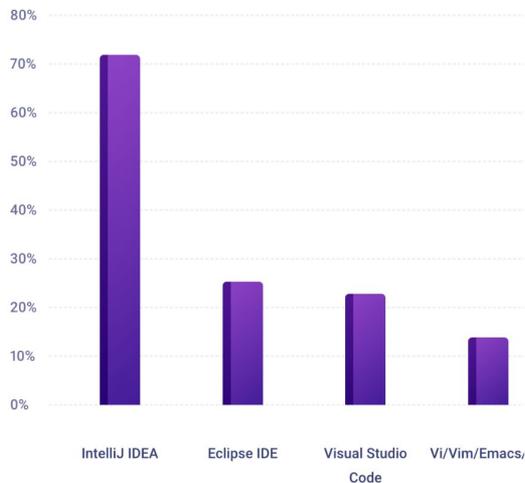
IntelliJ IDEA most dominant IDE in the Java Ecosystem

Over 70% of JVM developers use IntelliJ IDEA

For years, we've seen the dominance of JetBrains IntelliJ IDEA in the Java ecosystem. This year is not any different. IntelliJ IDEA is the most widely used within the JVM community. Also, the adoption of Visual Studio Code and Apache Netbeans grew tremendously.

We found out that more than half of the JVM community sees fit for multiple IDEs.

Most popular IDEs



有償の JetBrains IDE が支持される理由

お客様の声

「このコードの問題

IntelliJ | PhpStorm | PyCharm | Rider ...

(などのJetBrains 製品の静的コード解析で)

でチェックしていれば見つかったのに」

```
129 // checking if the pet name already exist for the owner
130 if (StringUtils.hasText(petName)) {
131     Pet existingPet = owner.getPet(petName.toLowerCase(), ignoreNew: false);
132     if (existingPet != null && existingPet.getId() != pet.getId()) {
133         result.rejectValue(field: "name", err
134     }
135 }
136
137 LocalDate currentDate = LocalDate.now();
138
```

© org.springframework.samples.petclinic.model.BaseEntity
public Integer getId()
spring-petclinic

- [High] Qodana: リクエストマッピング内のパス変数 'petId' を解決できません :69
- [High] Qodana: Number オブジェクトが 'equals()' ではなく '!=' を使用して比較されています
 - 'equals()' の代わりに '==' を使用した数値比較
 - [High] Number オブジェクトが 'equals()' ではなく '!=' を使用して比較されています :132**
 - > @PathVariable の宣言と実際の使用法の不一致
 - > 文法

```
pet(petname.toLowerCase(), ignoreNew: false),
```

```
& existingPet.getId() != pet.getId()) {
```

```
field: "name", error:
```

```
© org.springframework.samples.  
petclinic.model.BaseEntity
```

```
public Integer getId()
```

```
spring-petclinic
```

```
late now():
```

```
ingPet != null && existingPet.getId() != pet.getId()) {
```

Number オブジェクトが 'equals()' ではなく '!=' を使用して比較されています

'!=' をnullセーフな '!equals()' に置換  その他のアクション... 

JetBrains IDE によるクイックフィックス

```
if (existingPet != null && !Objects.equals(existingPet.getId(), pet.getId()))  
    result.rejectValue(field: "name", errorCode: "duplicate", defaultMessage: "a  
}
```

```
if (existingPet != null && !existingPet.getId().equals(pet.getId()))  
    result.rejectValue(field: "name", errorCode: "duplicate", defaultl
```

課題

1. 開発メンバー全員が JetBrains IDE を使っているわけではない
2. (JetBrains IDE を使っている)指摘を見逃している

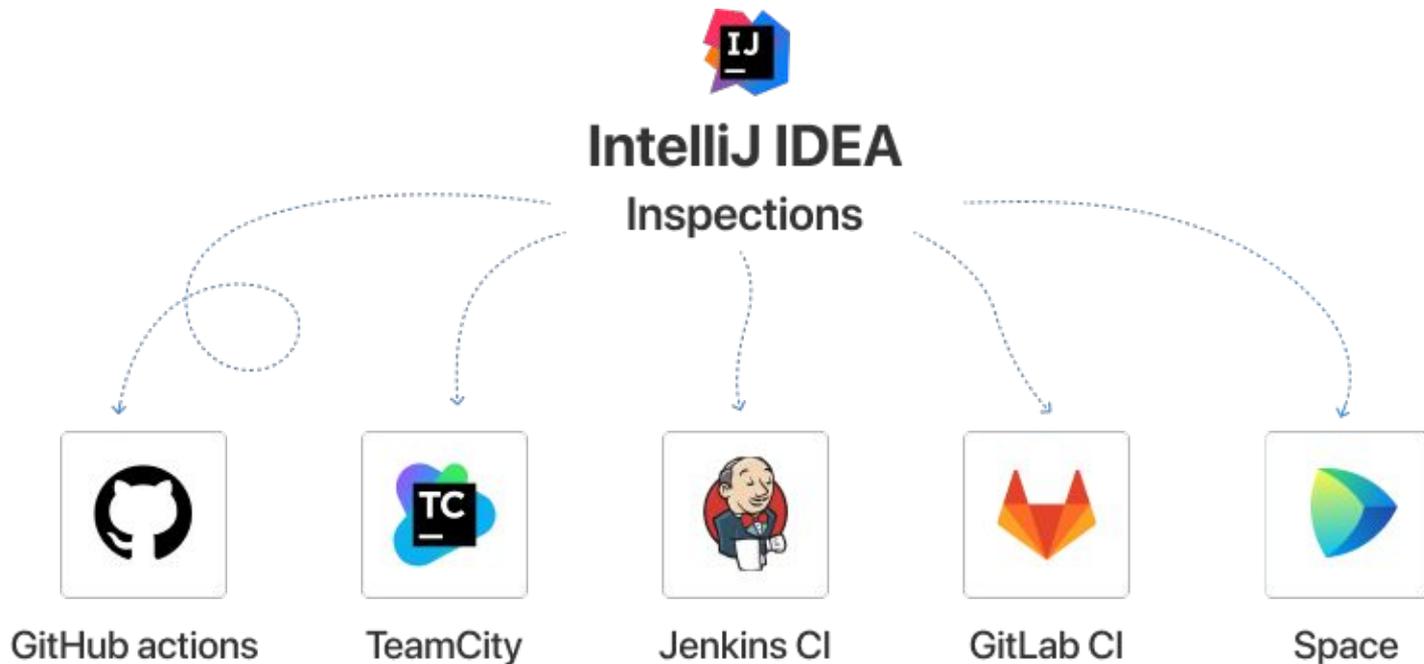


Qodana

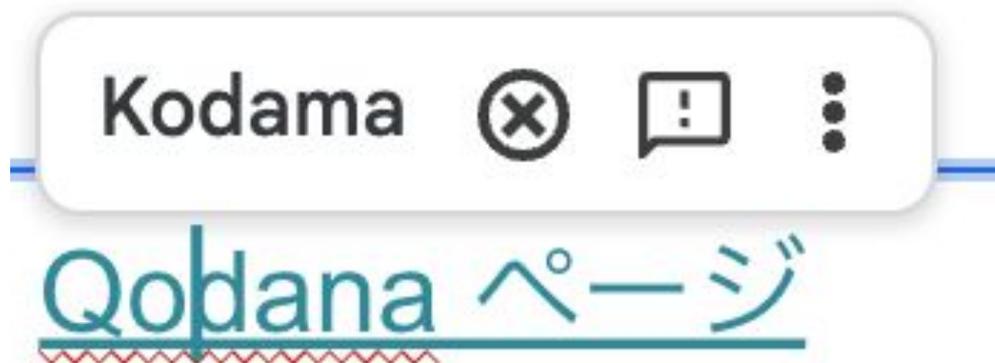
by JetBrains

概要

JetBrains IDE の静的コード解析機能をベースとする 様々な CI と統合可能なコード品質プラットフォーム



Qodana と書いて「こだな」と呼びます



約 2 年のプレビューを経て 2023 年 7 月に正式リリース



The Qodana Blog

The code quality platform for your favorite CI tool

Follow: [X](#) [RSS](#)

[Get Qodana](#)

[すべて](#) [ニュース](#) [Opinion](#)

[Code Review](#)

[Coding](#)

[Collaboration](#)

[News](#)

[Releases](#)

[Sales and Promotions](#)

Qodana が最高クラスの JetBrains IDE との統合機能を搭載して正式リリース



Ryuji Owan

2023年8月4日

Read this post in other languages:

[English](#), [Deutsch](#), [Español](#), [Français](#), [한국어](#), [简体中文](#), [Português do Brasil](#)

JetBrains は常に開発者が創造力と思考を刺激されながら楽しく作業できるツールの提供に努めてきました。JetBrains IDE はコードを理解し、それを改善するための有益な提案を行うように設計されています。このようなヒントがエディター内で提供されるのは大変便利です。しかし、CI を中心とする最近のワークフローではビルドパイプラインに信頼できるクオリティゲートが必要となります。Qodana はそれを念頭に制作されました。

Qodana 2023.2 の新機能

#1 JetBrains IDE 2023.2 で Qodana のサーバーサイド解析を完全に統合

#2 コードカバレッジのサポート

#3 Checkmarx データに基づく脆弱性チェッカー

#4 クイックフィックス (実地的)

2023.2 リリース以外の Qodana の機能

Qodana 限定のインスペクションを含む 2,500 個以上のコードインスペクション

対話型のインスペクションレポートとダッシュボード

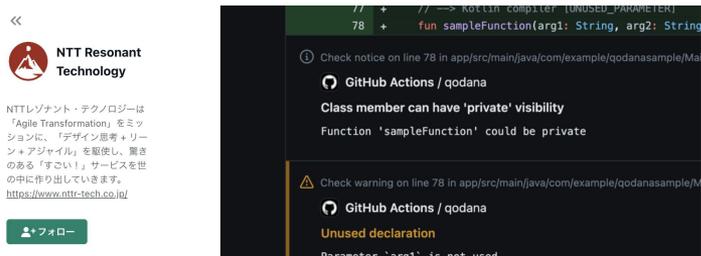
クラウドベースのレポート概要

技術的負債を管理するためのベースライン

サードパーティライセンス監査

インスペクション作成ツール

プレビュー期間中から日本のお客様にもご活用いただいているようです



[Qodana] [Android静的解析] GitHubのPull Requestに対してAndroid StudioのCode Inspectionsを自動実行する

♡ 5

LINE Engineering [Blog](#) [Interview](#) [Culture](#) [Open Source](#) [Careers](#) [Research](#)

LINE株式会社は、2023年10月1日にLINEヤフー株式会社になりました。LINEヤフー株式会社の新しいブログはこちらです。

Qodana(IntelliJ IDEAのCode Inspection)のCIへの組み込み ~ Kotlinのコード品質を高めるために ~

 Ryosuke Hosebe 2022-12-22
LINE Engineer

[f](#) [m](#) [x](#) [e](#)

TECH BLOG

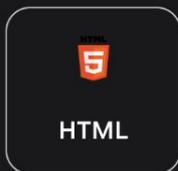
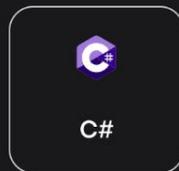
2023-08-30

ZOZOTOWN AndroidチームにおけるQodanaを活用したビルドワーニングへの取り組み

[Android](#) [GitHub Actions](#) [Java](#) [Kotlin](#)



JetBrains IDE がサポートする様々な言語やフレームワークに対応



∞
その他の
CI/CD システム

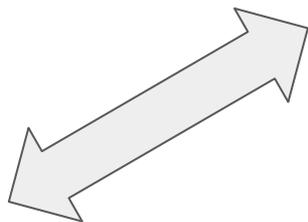
製品プランと価格

プラン	Community	Ultimate	Ultimate Plus
言語とテクノロジー	JVM、Android、Python	Community + PHP、ウェブ (JS、TSなど)、Go、.NET	Ultimateと同じ
フレームワーク		JVM: Spring、Jakarta EE 等 Python: Django、Flask 等 PHP: Drupal、WordPress 等	Ultimateと同じ
追加機能		セキュリティ解析	Ultimate + 依存関係セキュリティ チェック ライセンス監査
価格	無料	月額 ¥720/人 より	月額 ¥2160/人 より 今なら ¥1080/人

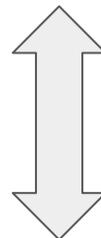
Github Actions との統合

Qodana の構成要素

1. CI で動作するコンテナベースの静的コード解析ツール



2. 管理コンソールとダッシュボード
Web アプリ/サービス (Qodana Cloud)



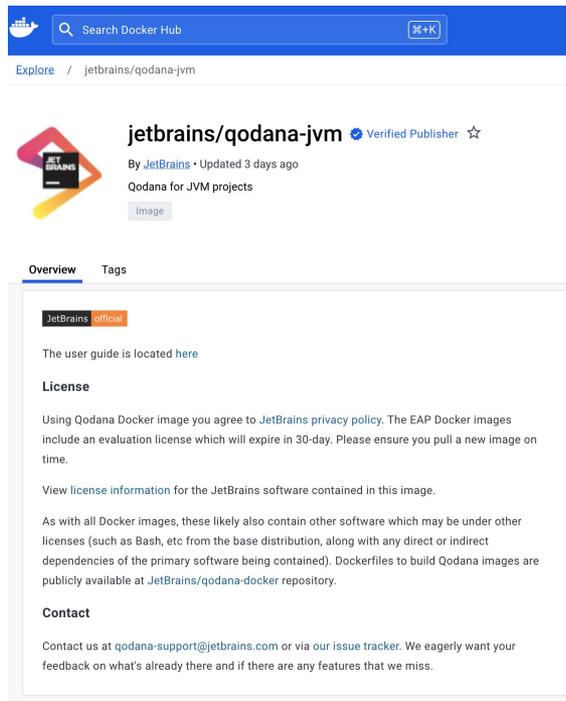
3. JetBrains IDEプラグインとVS Code拡張



コンテナベースの静的コード解析ツール

解析を実行するための Docker イメージ

GitHub 連携のための Action



Search Docker Hub

Explore / jetbrains/qodana-jvm

jetbrains/qodana-jvm Verified Publisher ☆

By JetBrains • Updated 3 days ago

Qodana for JVM projects

Image

Overview Tags

JetBrains **official**

The user guide is located [here](#)

License

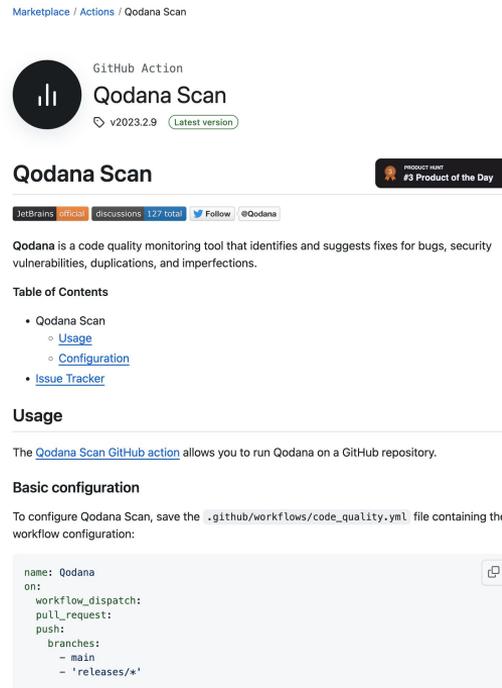
Using Qodana Docker image you agree to [JetBrains privacy policy](#). The EAP Docker images include an evaluation license which will expire in 30-day. Please ensure you pull a new image on time.

View [license information](#) for the JetBrains software contained in this image.

As with all Docker images, these likely also contain other software which may be under other licenses (such as Bash, etc from the base distribution, along with any direct or indirect dependencies of the primary software being contained). Dockerfiles to build Qodana images are publicly available at [JetBrains/qodana-docker](#) repository.

Contact

Contact us at qodana-support@jetbrains.com or via our [issue tracker](#). We eagerly want your feedback on what's already there and if there are any features that we miss.



Marketplace / Actions / Qodana Scan

GitHub Action

Qodana Scan

v2023.2.9 Latest version

Qodana Scan Product of the Day

JetBrains **official** discussions 127 total Follow @Qodana

Qodana is a code quality monitoring tool that identifies and suggests fixes for bugs, security vulnerabilities, duplications, and imperfections.

Table of Contents

- Qodana Scan
 - [Usage](#)
 - [Configuration](#)
- [Issue Tracker](#)

Usage

The [Qodana Scan GitHub action](#) allows you to run Qodana on a GitHub repository.

Basic configuration

To configure Qodana Scan, save the `.github/workflows/code_quality.yml` file containing the workflow configuration:

```
name: Qodana
on:
  workflow_dispatch:
  pull_request:
  push:
    branches:
      - main
      - 'releases/*'
```

qodana.yaml 設定ファイル

プロジェクトルートに配置し、チェック内容を指定 (IDE に設定に相当)

Qodana Scan 実行
のプロファイル

機能追加のための
JetBrains IDE プラグ
イン (例: 日本語言語
パック)

言語向け Linter の指
定
追加、除外ルール

```
5   version: "1.0"
6   #Specify inspection profile for code analysis
7   profile:
8     name: qodana.recommended
9   projectJDK: 17 #(Applied in CI/CD pipeline)
10  #bootstrap: sh ./prepare-qodana.sh
11  plugins:
12    - id: com.intellij.java
13  linter: jetbrains/qodana-jvm:latest
14  include:
15    - name: ClassComplexity
16    - name: CheckDependencyLicenses
17  #exclude:
18    # - name: <SomeDisabledInspectionId>
19    #   paths:
20    #     - <path/where/not/run/inspection>
21
```

.github/workflowsのワークフローファイルの例

コードをチェックアウト
(clone)

```
steps:  
  - uses: actions/checkout@v3  
    with:  
      ref: ${{ github.event.pull_request.head.sha }} # to  
      fetch-depth: 0 # a full history is required for pul
```

Qodana Scan 実行
のアクション

```
  - name: 'Qodana Scan'  
    uses: JetBrains/qodana-action@v2023.2
```

呼び出しパラメータ

```
    with:  
      args: --property=project.open.type=Maven, --baseline,
```

Qodana Cloud への
トークン

```
    env:  
      QODANA_TOKEN: ${{ secrets.QODANA_TOKEN }}|
```

Actions の実行結果で指摘を表示

<> Code 🔗 Pull requests 🔄 Actions 📁 Projects 🛡 Security 📄 Insights ⚙ Settings

← Qodana

🟢 make qodana work flow separated from maven build #1 Re-run a

🏠 Summary

GitHub Actions / Qodana for JVM
completed 45 minutes ago in 0s

Jobs

- 🟢 qodana
- Qodana for JVM

Run details

- 🕒 Usage
- 📄 Workflow file

51 new problems found by Qodana for JVM

Qodana for JVM

24 new problems were found

Inspection name	Severity	Problems
@NotNull/@Nullable 問題	Warning	8
未解決のビュー参照	Warning	6
Javadoc 宣言の問題	Warning	4
プレーンテキストとして指定されたリンク	Warning	3
@PathVariable の宣言と実際の使用法の不一致	Warning	2
'equals()' の代わりに '==' を使用した数値比較	Warning	1

⚠ Check warning on line 60 in src/main/java/org/springframework/samples/petclinic/owner/PetValidator.java 📄

github-actions / Qodana for JVM

@NotNull/@Nullable 問題

アノテーションのないパラメーターが @NotNullApi パラメーターをオーバーライドしています

⚠ Check warning on line 52 in src/main/java/org/springframework/samples/petclinic/owner/PetTypeFormatter.java 📄

github-actions / Qodana for JVM

@NotNull/@Nullable 問題

アノテーションのないメソッドが @NotNullApi アノテーションが付けられたメソッドをオーバーライドしています

⚠ Check warning on line 132 in src/main/java/org/springframework/samples/petclinic/owner/PetController.java 📄

github-actions / Qodana for JVM

'equals()' の代わりに '==' を使用した数値比較

Number オブジェクトが 'equals()' ではなく '!=' を使用して比較されています

プルリクエストに対するコメントとして指摘表示

 Open add problem by adding blank if body #1
mhorioka wants to merge 14 commits into `main` from `qodana` 

 **github-actions** bot commented 8 minutes ago

Qodana for JVM

1 new problem were found

Inspection name	Severity	Problems
本体が空のステートメント	◆ Warning	1

 Qodana analysis was run in the pull request mode: only the changed files were checked
 [View the detailed Qodana report](#)

▶ Contact Qodana team



それ以外の GitHub 連携方法はマニュアル([日本語](#) | [英語](#))にて解説

- クイックフィックス(自動修正)を行い、自動的にプッシュ
- Code Scanning との統合
- など

Qodana Cloud 管理コンソールとダッシュボード



Qodana Cloud for JetBrains Test Masaru Fake Company

JetBrains Test Team

2 projects, 1 member

QC

J



intellij-samples-qodana-java

 refs/heads/main 1h ago

 **242** problems

Coverage not enabled

spring-petclinic-qodana

 refs/heads/qodana 48m ago

 **24** problems

Coverage not enabled



spring-petclinic-qodana

af5b34ac 49m ago



ACTUAL PROBLEMS

24

BASELINE

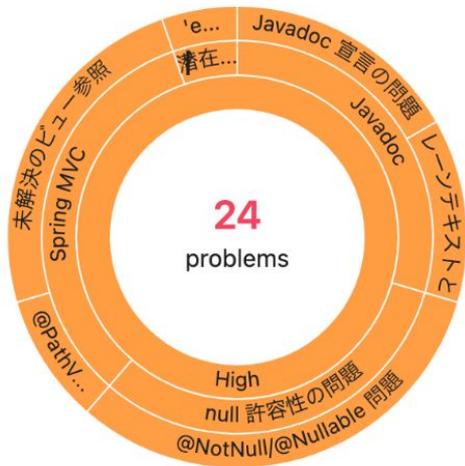
27 problems

CONFIGURATION

754 inspections

LICENSE AUDIT

Not passed



Files and folders: All ▾

Severity: All ▾

Category: All ▾

Type: All ▾

Tags ▾

Problems 24

Files 8

Search



Group by: File ▾

PetController.java

- Number オブジェクトが 'equals()' ではなく '!=' を使用して比較されています 潜在的なバグ 'equals()' の代わりに '==' を使用した数値比較

JAVA

[src/main/java/org/springframework/samples/petclinic/owner/PetController.java](#)

```
130         if (StringUtils.hasText(petName)) {
131             Pet existingPet = owner.getPet(petName.toLowerCase(), false);
132             if (existingPet != null && existingPet.getId() != pet.getId()) {
133                 result.rejectValue("name", "duplicate", "already exists");
134             }

```

[Open file in IntelliJ IDEA](#)

More actions

[Copy link](#)

`equals()` ではなく `==` または `!=` を使用して `Number` の等価性をテストしているコードを報告します。

自動ボックスングを使用すると、2つのプリミティブではなくラッパー型の2つのインスタンスを比較するという間違いを犯しやすくなります。たとえば、`int` ではなく `Integer` を使用しているケースが挙げられます。

例:

```
void foo(Integer a, Integer b) {
    final boolean bool = a == b;
}
```

[Open external resource](#)

`a` が非 null であることがわかっている場合は、「安全でない」クイックフィックスを適用して次のような結果を得ても問題ありません。

```
void foo(Integer a, Integer b) {
    final boolean bool = a.equals(b);
}
```

Inspection ID: NumberEquality

[Show less](#)

PetController.java

Number オブジェクトが 'equals()' ではなく '!=' を使用して比較されています 潜在的なバグ 'equals()' の代わりに '=' を使用した数値比較

JAVA

src/main/java/org/springframework/samples/petclinic/owner/PetController.java

```

130         if (StringUtils.hasText(petName)) {
131             Pet existingPet = owner.getPet(petName.toLowerCase(), false);
132             if (existingPet != null && existingPet.getId() != pet.getId()) {
133                 result.rejectValue("name", "duplicate", "already exists");
134             }
    
```

Open file in IntelliJ IDEA More actions

Copy link

IntelliJ IDEA VCS Visual Studio Code

equals() ではなく == または != を使用して比較されているコードを報告します。自動ボックスを使用すると、2つの異なる型 (たとえば Integer と Integer) の 2 つのインスタンスを比較するという間違いを犯しやすくなります。たとえば、int ではなく Integer を使用している場合、例:

```

void foo(Integer a, Integer b) {
    final boolean bool = a = b;
}
    
```

a が非 null であることがわかっている場合は、「安全でない」クイックフィックスを適用して次のような結果を得ても問題ありません。

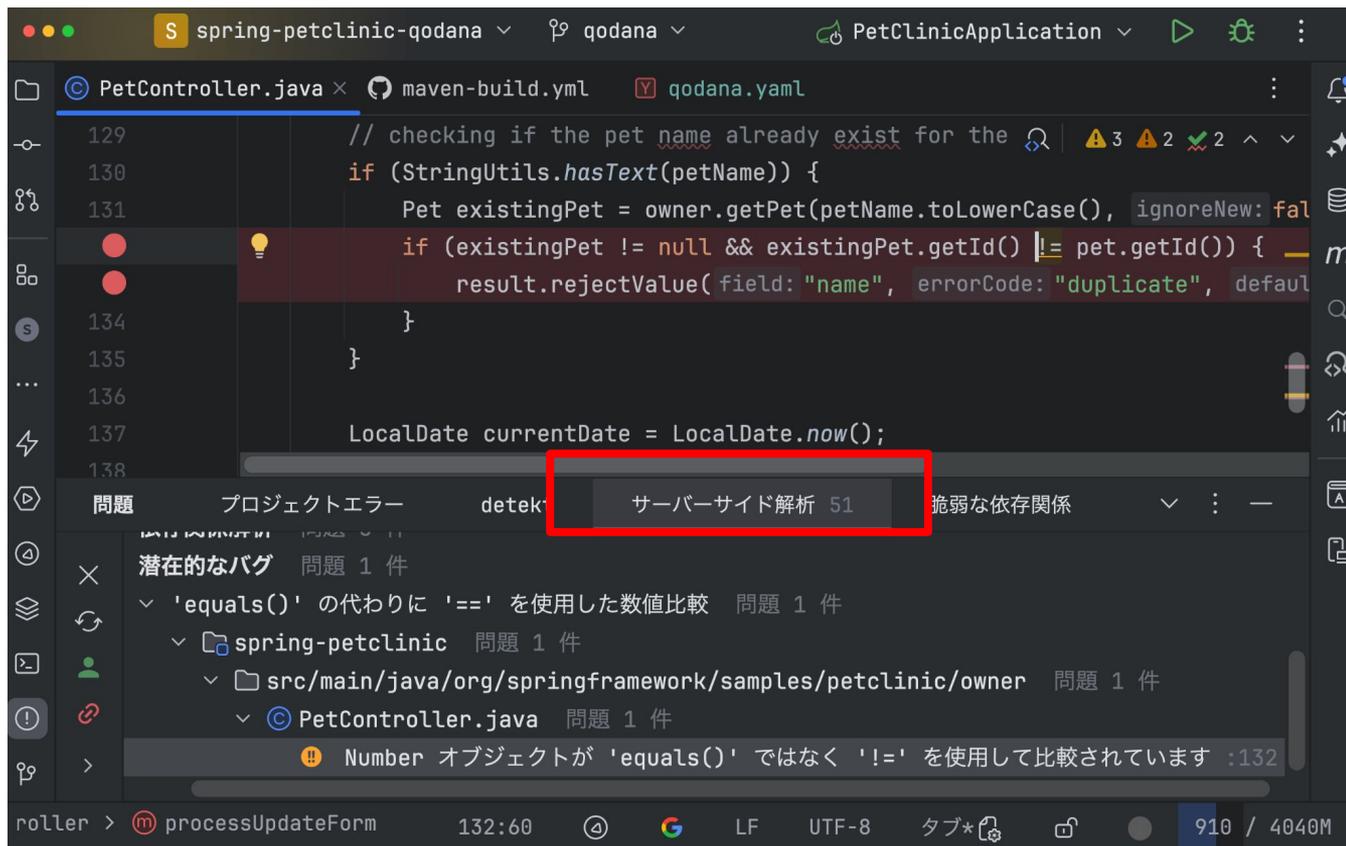
```

void foo(Integer a, Integer b) {
    final boolean bool = a.equals(b);
}
    
```

Inspection ID: NumberEquality

Show less

JetBrains IDE Qodana プラグイン(標準バンドル)



まとめ

JetBrains の静的解析ツール Qodana と GitHub に連携について紹介しました。ぜひお試しください！

- OSS の解析結果を見る: [Qodana Cloud](#)
- 自社コードの解析結果を見る: [JetBrains IDE から Qodana をローカル実行](#) (Qodanaライセンス不要)
- GitHub Actions との統合: 無料版や有償版の 60 日トライアルをお試しください

ブースでは JetBrains グッズや簡単なアンケートにお答えいただくと割引クーポンを配布しています。

オンラインの方は[こちら \(回答期限は12/15まで\)](https://forms.gle/LM7EvMwRmPA6WKe27) => <https://forms.gle/LM7EvMwRmPA6WKe27>

QRコードからアンケートにアクセス 



参考資料

JetBrains 日本語ウェブサイト [Qodana ページ](#)

Qodana マニュアル [英語](#) | [日本語](#)

JetBrains 日本語ブログ [Qodana関連](#)

Qodana/GitHub Actions/IDE 連携解説ビデオ [Unlock Code Quality with the Qodana IDE Plugin](#)

[Qodana CLI](#) , [Qodana Scan](#) (GitHub Action)

OSSの解析結果例 [Qodana Cloud](#)

各社様開発者ブログで Qodana を取り上げていただいている例

- NTTレゾナント・テクノロジー様 <https://note.com/nttrtech/n/n1bf364bb2d86>
- ZOZO 様 <https://techblog.zozo.com/entry/zozotown-android-improve-build-warning>
- LINE ヤフー様 <https://engineering.linecorp.com/ja/blog/qodona-introduction>

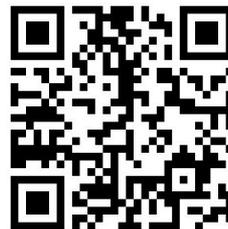
ありがとうございました

フィードバック、お問い合わせ、技術サポート:

日本語窓口: sales.jp@jetbrains.com

X/Twitter: [@masaruhr](https://twitter.com/masaruhr)

QRコードからクーポンが当たるアンケートにアクセス 



オンラインの方はこちら (回答期限は12/15まで)=> <https://forms.gle/LM7EvMwRmPA6WKe27>