Equipe

Gustavo Taufembach Bett

Thiago Dimon Miranda

Turma de terça-feira

Ktlint: Linter para formatação de código Kotlin



Ktlint é uma ferramenta linter para garantir a consistência e a qualidade do código Kotlin. Seu objetivo é realizar o processo de análise estática de código para identificar possíveis erros de formatação com base nas convenções oficiais da linguagem ou personalização de regras.

Instalação e Configuração do Ktlint

Passos para Instalação

 Adicionar a dependência no arquivo build.gradle.kts ou no build.gradle

Snippet de Código

```
plugins {
  id("org.jlleitschuh.gradle.ktlint") version <current_version>
}

repositories {
  // Required to download KtLint
  mavenCentral()
}
```

Instalação e Configuração do Ktlint

Passos para Instalação

- Adicionar a dependência no arquivo build.gradle.kts ou no build.gradle
- Configure as regras conforme a necessidade no arquivo .editorconfig.

Snippet de Código

```
root = true

[*.{kt,kts}]

indent_size = 2

max_line_length = 80

insert_final_newline = true

ktlint_standard_no-wildcard-imports = true
```

Instalação e Configuração do Ktlint

Passos para Instalação

- Adicionar a dependência no build.gradle.kts ou build.gradle.
- Configure as regras conforme a necessidade no arquivo .editorconfig.
- 3. Executar via linha de comando os scripts para formatação.

Snippet de Código

```
// Para verificar problemas de formatação./gradlew ktlintCheck// Para corrgir problemas automaticamente./gradlew ktlintFormat
```

Integração com o Ambiente de Desenvolvimento

A integração com o ambiente de desenvolvimento pode se dar de várias formas, permitindo que as verificações e formatação de código possam ocorrer em tempo real ou antes de um commit.

Plugin para IDEs

Disponíveis para IntelliJ IDEA e Android Studio, basta instalar o plugin do ktlint para verificação em tempo real.

Inspeções e Quick Fixes

Aproveite as inspeções e quick fixes do ktlint no IDE para identificar e corrigir problemas de estilo de forma rápida e fácil.

Git Hooks

Para garantir que nenhum código fora do padrão seja commitado, o Ktlint pode ser adicionado como um hook do Git.

Além disso, você pode configurar atalhos de teclado para formatar o código rapidamente, como Alt+Shift+F.

Estilização Automática e Integração CI/CD

1

2

Formatação Automática

Execute "./gradlew ktlintFormat" para corrigir erros de estilo.

Integração CI/CD

Garante formatação consistente em todas as etapas do desenvolvimento.

3

Ignorar Arquivos

Use ".editorconfig" para configurar exceções.

A configuração de regras personalizáveis e exceções, permite uma estilização e adaptação conforme a necessidade do projeto.



Guias de Estilo: A Base do ktlint



Convenções do Kotlin

- **Indentação:** Usa 4 espaços em vez de tabs;
- Comprimento de linhas: limite de 100 caracteres;
- Espaçamento: Usa um único espaço após palavras-chaves;
- Uso de chaves: Sempre obrigatórias para estruturas de controle;
- Imports: Sem imports com *.



Google Kotlin

Semelhante ao padrão oficial, mas com algumas diferenças:

- No uso de var sempre que possível utilizar val;
- Organização dos imports em ordem alfabética;
- Espaço em branco após comentários de linha única.

ktlint_standard_google = true



Regras Customizadas

Aqui você pode customizar as regras conforme preferir e a necessidade do projeto.



Regras Configuráveis: Indentação e Comprimento da Linha

Ktlint permite configurar regras como `indent` (indentação) e `max-line-length` (comprimento da linha) para melhorar a legibilidade do código.

Indentação

Configure o tamanho da indentação (e.g., 4 espaços) e alinhamento de parâmetros. Exemplo: `indent_size = 4` no .editorconfig.

Comprimento Máximo

Defina o limite de caracteres por linha (e.g., 120) e a quebra automática. Exemplo: `max_line_length = 120` no .editorconfig.

Made with Gamma

```
☑ Main.kt

     m pom.xml (KtLinter)
                                         ② .editorconfig
        ◎ [*.{kt,kts}]
           indent_size = 4
           indent_style = space
[INFO] Executing tasks
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:4:13: Argument should be on a separate line (unless all arguments can fit a single line) (standar
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:4:121: Exceeded max line length (120) (standard:max-line-length)
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:4:240: Missing newline before ")" (standard:argument-list-wrapping)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:6:1: Unexpected indentation (8) (should be 4) (standard:indent)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:8:13: Parameter should start on a newline (standard:parameter-list-wrapping)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:8:13: Missing newline after "(" (standard:wrapping)
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:8:13: Newline expected after opening parenthesis (standard:function-signature)
[INFO]
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/Ktlinter/src/main/kotlin/Main.kt:9:1: Unexpected indentation (12) (should be 8) (standard:indent)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/Ktlinter/src/main/kotlin/Main.kt:9:13: Parameter should start on a newline (standard:function-signature)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:9:23: Missing newline before ")" (standard:wrapping)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:9:24: Missing newline before ")" (standard:parameter-list-wrapping)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:9:24: Newline expected before closing parenthesis (standard:function-signature)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:9:24: Missing trailing comma before ")" (standard:trailing-comma-on-declaration-site)
[INFO]
          [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:13:1: File must end with a newline (\n) (standard:final-newline)
[INFO]
          [ktlint] 23:05:34.895 [main] WARN com.pinterest.ktlint.cli.internal.KtlintCommandLine -- Lint has found errors than can be autocorrected using 'ktlint --format'
[INFO]
          [ktlint]
[INFO]
          [ktlint] Summary error count (descending) by rule:
[INFO]
          [ktlint] standard:function-signature: 3
[INFO]
          [ktlint] standard:argument-list-wrapping: 2
[INFO]
          [ktlint] standard:indent: 2
[INFO]
          [ktlint] standard:parameter-list-wrapping: 2
[INFO]
          [ktlint] standard:wrapping: 2
[INFO]
          [ktlint] standard:final-newline: 1
[INFO]
          [ktlint] standard:max-line-length: 1
[INFO]
          [ktlint] standard:trailing-comma-on-declaration-site: 1
```

Com ktlint —format

Regras Configuráveis: Importações e Nomenclatura

Ktlint oferece regras para controlar importações e nomenclatura, promovendo clareza e organização no código.



Importações Coringa

(AB

AB Nomenclatura

Evite `import foo.*`: use importações explícitas para clareza.

Use `snake_case` para variáveis e `camelCase` para funções.

Código mais organizado, fácil de entender e colaborativo, reduzindo ambiguidades.

```
m pom.xml (KtLinter)
                                     root = true
     3 ◎ [*.kt,*.kts]
          naming.variable-naming = snake_case
          naming.function-naming = camelCase
           no-wildcard-imports = enabled
[INFU] NO SOURCES TO COMPILE
      --- surefire:2.22.2:test (default-test) @ KtLinter ---
[INFO]
[INFO]
[INFO] --- jar:3.4.1:jar (default-jar) @ KtLinter ---
[INFO] Building jar: C:\Users\Gustavo\Desktop\projetos\KtLinter\target\KtLinter-1.0-SNAPSHOT.jar
[INFO]
[INFO] --- antrun:3.1.0:run (ktlint) @ KtLinter ---
[INFO] Executing tasks
        [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:3:1: Wildcard import (standard:no-wildcard-imports)
[INFO]
        [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:8:9: Function name should start with a lowercase letter (except factory methods) and use camel ca
[INFO]
        [ktlint] Summary error count (descending) by rule:
[INFO]
        [ktlint] standard:function-naming: 1
[INFO]
[INFO] -----
```

Código

```
import java.util.* // X Erro: Importação coringa proibida

fun main() {
   val minhaVariavel = "Erro" // X Erro: Deve estar em snake_case

   fun Minha_Funcao() { // X Erro: Deve estar em camelCase
        println("Erro de nomenclatura")
   }
}
```

Regras Configuráveis: Indentação em String Templates

A regra string-template-indent assegura indentação consistente em strings multi-linha, otimizando a legibilidade de strings complexas, especialmente em códigos ou dados extensos.

Indentação Consistente

Uniformiza a indentação em strings multi-linha.

Legibilidade Aprimorada

Facilita a leitura de strings complexas.

Configuração Simplificada

Ative com: string-template-indent = enabled

Em resumo, ktlint é vital para a qualidade do código Kotlin. Use a documentação, ajuste as configurações e aplique o ktlint para um código mais limpo, consistente e de fácil manutenção.

```
m pom.xml (KtLinter)
                           @ .editorconfig
           root = true

    [*.kt,*.kts]

[INFO]
[INFO] --- jar:3.4.1:jar (default-jar) @ KtLinter ---
[INFO] Building jar: C:\Users\Gustavo\Desktop\projetos\KtLinter\target\KtLinter-1.0-SNAPSHOT.jar
[INFO] --- antrun:3.1.0:run (ktlint) @ KtLinter ---
[INFO]
         [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:5:24: A multiline expression should start on a new line (standard:multiline-expression-wrapping)
[INFO]
         [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:5:24: Expected newline before multiline string template (standard:string-template-indent)
[INFO]
         [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:9:1: Unexpected indent of multiline string closing quotes (standard:indent)
[INFO]
         [ktlint] C:/Users/Gustavo/Desktop/projetos/KtLinter/src/main/kotlin/Main.kt:9:29: Expected newline after multiline string template (standard:string-template-indent)
[INFO]
         [ktlint] 23:41:23.593 [main] WARN com.pinterest.ktlint.cli.internal.KtlintCommandLine -- Lint has found errors than can be autocorrected using 'ktlint --format'
[INFO]
[INFO]
         [ktlint] Summary error count (descending) by rule:
[INFO]
         [ktlint] standard:string-template-indent: 2
[INFO]
         [ktlint] standard:indent: 1
         [ktlint] standard:multiline-expression-wrapping: 1
```

Código