

Mal cheiros detectados

1 Expressões booleanas negativas

2 Ifs aninhados

1- Expressões booleanas negativas

Antes:

```
public void registraUsuario2(String nome)
    throws UsuarioJaRegistradoException, UsuarioComNomeVazioException,
    UsuarioInexistenteException {
    if (nome != null) {
        if (!nome.isEmpty()) {
            Usuario usuario = new Usuario(nome);
            if (!_usuarios.contains(usuario)) {
                _usuarios.add(usuario);
            } else
                throw new UsuarioJaRegistradoException("--->Já existe usuário com o nome \""
                    + nome + "\"! Use outro nome!");
        } else
            throw new UsuarioComNomeVazioException("--->Não pode registrar usuario com nome vazio!");
    } else
        throw new UsuarioInexistenteException("--->Não pode registrar usuario inexistente!");
}
```

Depois:

```
public void registraUsuario2(String nome)
    throws UsuarioJaRegistradoException, UsuarioComNomeVazioException,
    UsuarioInexistenteException {
    if (nome != null) {
        if (nome.isEmpty()) {
            throw new UsuarioComNomeVazioException("--->Não pode registrar usuario com nome vazio!");
        } else{
            Usuario usuario = new Usuario(nome);
            if (_usuarios.contains(usuario)) {
                throw new UsuarioJaRegistradoException("--->Já existe usuário com o nome \""
                    + nome + "\"! Use outro nome!");
            } else
                _usuarios.add(usuario);
        }
    } else
        throw new UsuarioInexistenteException("--->Não pode registrar usuario inexistente!");
}
```

```
▲ tarefa.principal.test.BibliotecaTestContextoInicial [Runner: JUnit 4] (0,000 s)
  ✔ whenAdicionoLivroNuloEntaoAdicionarLivroInexistenteExceptionEhLancada (0,000 s)
  ✔ whenAdicionoUsuarioJaExistenteEntaoUsuarioJaRegistradoExceptionEhLancada (0,000 s)
  ✔ whenAdicionoUmLivroEntaoIncrementaTamListaLivrosDisponiveisDeUm (0,000 s)
  ✔ whenAdicionoUsuarioInexistenteEntaoUsuarioInexistenteExceptionEhLancada (0,000 s)
  ✔ whenAdicionoUmUsuarioEntaoIncrementaTamListaUsuariosDeUm (0,000 s)
  ✔ whenAdicionoOutroLivroComMesmoTituloEAutorEntaoIncrementaTamListaLivrosDisponiveisDeUm (0,000 s)
  ✔ whenAdicionoUsuarioComNomeVazioEntaoUsuarioComNomeVazioExceptionEhLancada (0,000 s)
```

2- Ifs aninhados

Antes:

```
public void registraUsuario2(String nome)
    throws UsuarioJaRegistradoException, UsuarioComNomeVazioException,
        UsuarioInexistenteException {
    if (nome != null) {
        if (nome.isEmpty()) {
            throw new UsuarioComNomeVazioException("--->Não pode registrar usuario com nome vazio!");
        } else{
            Usuario usuario = new Usuario(nome);
            if (_usuarios.contains(usuario)) {
                throw new UsuarioJaRegistradoException("--->Já existe usuário com o nome \""
                    + nome + "\"! Use outro nome!");
            } else
                _usuarios.add(usuario);
        }
    } else
        throw new UsuarioInexistenteException("--->Não pode registrar usuario inexistente!");
}
```

Depois:

```
public void registraUsuario2(String nome)
    throws UsuarioJaRegistradoException, UsuarioComNomeVazioException, UsuarioInexistenteException {
    if (nome == null)
        throw new UsuarioInexistenteException("--->Não pode registrar usuario inexistente!");
    if (nome.isEmpty())
        throw new UsuarioComNomeVazioException("--->Não pode registrar usuario com nome vazio!");
    Usuario usuario = new Usuario(nome);
    if (_usuarios.contains(usuario))
        throw new UsuarioJaRegistradoException(
            "--->Já existe usuário com o nome \"" + nome + "\"! Use outro nome!");
    _usuarios.add(usuario);
}
```

```
▲ tarefa.principal.test.BibliotecaTestContextoInicial [Runner: JUnit 4] (0,001 s)
  ✔ whenAdicionoLivroNuloEntaoAdicionarLivroInexistenteExceptionEhLancada (0,000 s)
  ✔ whenAdicionoUsuarioJaExistenteEntaoUsuarioJaRegistradoExceptionEhLancada (0,001 s)
  ✔ whenAdicionoUmLivroEntaoIncrementaTamListaLivrosDisponiveisDeUm (0,000 s)
  ✔ whenAdicionoUsuarioInexistenteEntaoUsuarioInexistenteExceptionEhLancada (0,000 s)
  ✔ whenAdicionoUmUsuarioEntaoIncrementaTamListaUsuariosDeUm (0,000 s)
  ✔ whenAdicionoOutroLivroComMesmoTituloEAutorEntaoIncrementaTamListaLivrosDisponiveisDeUm (0,000 s)
  ✔ whenAdicionoUsuarioComNomeVazioEntaoUsuarioComNomeVazioExceptionEhLancada (0,000 s)
```

- Resultado Final da Refatoração

Antes:

```
public void registraUsuario2(String nome)
    throws UsuarioJaRegistradoException, UsuarioComNomeVazioException,
    UsuarioInexistenteException {
    if (nome != null) {
        if (!nome.isEmpty()) {
            Usuario usuario = new Usuario(nome);
            if (!_usuarios.contains(usuario)) {
                _usuarios.add(usuario);
            } else
                throw new UsuarioJaRegistradoException("--->Já existe usuário com o nome \""
                    + nome + "\"! Use outro nome!");
        } else
            throw new UsuarioComNomeVazioException("--->Não pode registrar usuario com nome vazio!");
    } else
        throw new UsuarioInexistenteException("--->Não pode registrar usuario inexistente!");
}
```

Depois:

```
public void registraUsuario2(String nome)
    throws UsuarioJaRegistradoException, UsuarioComNomeVazioException, UsuarioInexistenteException {
    if (nome == null)
        throw new UsuarioInexistenteException("--->Não pode registrar usuario inexistente!");
    if (nome.isEmpty())
        throw new UsuarioComNomeVazioException("--->Não pode registrar usuario com nome vazio!");
    Usuario usuario = new Usuario(nome);
    if (_usuarios.contains(usuario))
        throw new UsuarioJaRegistradoException(
            "--->Já existe usuário com o nome \"" + nome + "\"! Use outro nome!");
    _usuarios.add(usuario);
}
```