

repetition benefits

```
export PATH=/opt/homebrew/opt/ruby/bin:/opt/homebrew/opt/openjdk/bin:/opt/homebrew/bin:/opt/homebrew/sb
```

このコマンドは、シェルの環境変数 `PATH` に複数のディレクトリを追加しています。これにより、指定されたディレクトリ内の実行可能ファイルがシェルから直接実行できるようになります。各ディレクトリは、異なるソフトウェアやツールの実行ファイルが含まれている可能性があります。例えば、`/opt/homebrew/opt/ruby/bin` には Ruby の実行ファイルが、`/Users/lzwjava/platform-tools` には Android プラットフォームツールが含まれているかもしれません。

この場合、変数の使用を避けてください:

```
"$HOME/Library/Python/3.9/bin"
```

```
export PATH=/opt/homebrew/opt/ruby/bin:/opt/homebrew/opt/openjdk/bin:/opt/homebrew/bin:/opt/homebrew/sb
```

このコマンドは、`PATH` 環境変数を設定しています。`PATH` は、シェルがコマンドを実行する際に検索するディレクトリのリストです。この設定により、指定されたディレクトリ内の実行ファイルがシェルから直接実行できるようになります。

2 番目のバージョンは、`/Users/lzwjava` ディレクトリが複数回出現するため、少し繰り返しが多いように見えますが、よりクリーンで読みやすくなっています。

プログラミングにおいて、繰り返しは実際には有益である場合があります。すべてを変数や関数で抽象化したり簡略化したりする必要はありません。この場合、ホームディレクトリ `/Users/lzwjava` は頻繁に変更されることはないため、それを繰り返すことは全く問題ありません。

これは、人の名前を参照する方法と似ています。「彼」や「彼女」、「彼ら」といった代名詞を多用しすぎると、誰のことを話しているのかが不明確になります。名前自体は変わる可能性が低く、直接それらを使用することで明確さを加えることができます。

抽象化と単純化は重要ですが、場合によっては不必要な複雑さを招くことがあります。時には、繰り返しの方がシンプルで理解しやすいこともあります。