

- Exercícios escritos -

Haskell, Python e Java - Semelhanças e diferenças de cada Linguagem.

Python é uma linguagem de programação de alto nível, interpretada e que abrange diversos paradigmas de programação, sendo eles: A programação imperativa, orientada a objetos, de script e funcional. Utiliza-se de tipagem dinâmica e forte. Além disso, Python assim como a maioria das linguagens modernas, compila para código nativo.

Já o Haskell, é uma linguagem puramente funcional, ou seja, atende apenas ao paradigma de programação funcional. Um dos pontos que estão sendo mais utilizados na linguagem atualmente é o foco no alcance de soluções para problemas matemáticos. Além disso, o código escrito na linguagem é de fácil entendimento, além de ser claro e possuir uma variedade de aplicações.

Java é uma linguagem orientada a objetos, genérica e procedural. Uma de suas maiores vantagens é a portabilidade, pois uma vez escrito o código, ele poderá ser executado em qualquer máquina. Outro ponto forte da linguagem, é a facilidade para criação de programas distribuídos e de multitarefas ou threads. Java é uma linguagem fortemente tipada, ou seja, a declaração do tipo dos dados é obrigatória, não permitindo a alteração dinâmica.

Uma das diferenças entre as três linguagens de programação acima, é justamente, o paradigma de programação utilizado por cada uma. Enquanto Python aborda e aceita diversos paradigmas de programação como imperativa, orientada a objetos, etc, o Haskell utiliza-se apenas do paradigma funcional, enquanto Java do paradigma de orientação a objetos. A utilização de cada linguagem pelo programador deve ser vista como um trunfo, utilizando a melhor opção para resolver determinado problema. Por isso, conhecer diferentes linguagens e paradigmas de programação é de grande importância, dando ao programador maior poder para sempre fazer a melhor escolha.

Slide 39

2°

R: O paradigma utilizado para escrever os programas são, normalmente, definidos de acordo com a necessidade e objetivo de cada um. Por exemplo, um software desenvolvido por uma empresa irá provavelmente utilizar a abordagem de orientação objetos, permitindo uma maior compreensão do código, manutenção e escalabilidade. Por outro lado, no caso de um sistema embarcado ou até mesmo uma maratona de programação onde a programação estruturada é utilizada, os desenvolvedores/competidores estão em busca de um código de maior desempenho, otimizado e implementado no menor tempo possível.

No meu caso, acabo utilizando muito das duas abordagens, programação orientada a objetos para criação de softwares e programação estruturada para criar códigos com maior desempenho e papel bem definido.

3°

R: Um dos pontos em comum que os programadores possuem é a capacidade de escrever código. Porém, a escrita de um código eficiente e que atende bem ao problema proposto, é de fato menos frequente de acontecer. Ao afirmar que "Não existe agora, e nem existirá, uma linguagem de programação na qual seja difícil escrever programas ruins" Larry quer dizer, na minha visão, que não é a linguagem de programação utilizada que irá determinar a qualidade do código, mas sim, o programador. E de fato, se formos pensar, seria praticamente impossível ter uma linguagem que conseguisse guiar o programador para as melhores soluções, tirando a responsabilidade da escrita de um bom código do mesmo.