



## RPG0014 – Iniciando o caminho pelo Java

### Objetivos da prática

- Utilizar herança e polimorfismo na definição de entidades.
- Utilizar persistência de objetos em arquivos binários.
- Implementar uma interface cadastral em modo texto.
- Utilizar o controle de exceções da plataforma Java.
- No final do exercício, o aluno terá implementado um sistema cadastral em Java, utilizando recursos da programação orientada a objetos e a persistência em arquivos binários.

### Análise e conclusão

- **Quais são as vantagens e desvantagens do uso de herança?** Algumas vantagens são: reutilização de atributos e métodos, polimorfismo, uma maior abstração, etc. Algumas desvantagens são: é mais complexa, é difícil fazer a manutenção, não é suportado por todos os sistemas de bancos de dados, entre outros.
- **Por que a interface Serializable é necessária ao efetuar a persistencia em arquivos binários?** Porque ela sinaliza ao ambiente de execução Java que uma classe pode ser serializada, ou seja, que seus objetos podem ser convertidos em uma sequência de bytes. Essa sequência de bytes pode ser salva em um arquivo binário ou transmitida pela rede e, posteriormente, reconstruída como objetos.
- **Como o paradigma funcional é utilizado pela API stream no Java?** O paradigma funcional é utilizado principalmente através de expressões lambda e operações de alto nível que permitem manipular dados de maneira funcional.
- **Quando trabalhamos com Java, qual padrão de desenvolvimento é adotado na persistência de dados em arquivos?** Padrão DAO

## Análise e conclusão

- **O que são elementos estáticos e qual o motivo para o método `main` adotar esse modificador?** São elementos que normalmente estão associados à classe como um todo. Se o método `main` não adotasse esse modificador a JVM teria que criar uma instância da classe antes de chamar o método `main`, ou seja, se não houver o modificador no método `main`, o método não poderá se associar à uma classe inteira, apenas a instâncias específicas.
- **Para que serve a classe `Scanner`?** É usada para ler dados de entrada a partir de várias fontes, como o teclado, arquivos ou strings. Fornece métodos para analisar e converter tipos primitivos e strings a partir da entrada.
- **Como o uso de classes de repositório impactou na organização do código?** Pode impactar de várias formas, principalmente de forma positiva. Como por exemplo: facilidade de manutenção, reutilização do código, melhora na leitura, flexibilidade na implementação, etc.