

PROGRAMAÇÃO ORIENTADA A OBJETOS – EXERCÍCIOS POO + JAVA

1 Crie uma classe chamada **Invoice** para que uma loja de suprimentos de informática a utilize para representar uma fatura de um item vendido nela. Uma **Invoice** (fatura) deve incluir quatro partes das informações como *variáveis de instância* (atributos) — o número (tipo **String**), a descrição (tipo **String**), a quantidade comprada de um item (tipo **int**) e o preço por item (**double**). Sua classe deve ter um construtor que inicializa as quatro variáveis de instância. Forneça um método *set* e um *get* para cada variável de instância. Além disso, forneça um método chamado **getInvoiceAmount** que calcula o valor de fatura (isto é, multiplica a quantidade pelo preço por item) e depois retorna esse valor como **double**. Se a quantidade não for positiva, ela deve ser configurada como 0. Se o preço por item não for positivo, ele deve ser configurado como 0.0. Escreva um aplicativo de teste para demonstrar as capacidades da classe **Invoice**.

Como criar um projeto no IntelliJ – na página inicial escolha New Project. Na tela New Project aberta, preencha o campo “Project Name” com o nome do projeto – utilize **Exercicio1**. Se quiser, ajuste o caminho da pasta de acordo com sua configuração no campo “Project Location”, selecione “Java” e “IntelliJ”, e confirme que o campo “Project SDK” esteja preenchido com a versão correta do JDK. Pressione [Create].

Um projeto vazio será criado e aberto, conforme as informações fornecidas.

Agora, para criar sua primeira classe Java, expanda a pasta Exercicio1 e clique com o botão direito sobre a pasta “src”. Depois selecione a opção [New > Java Class]. A pasta “src” (*source*) conterá todo o código fonte do projeto. Esta pasta não é um pacote (*package*) e não faz parte do *namespace* das classes.

Ao selecionar uma nova classe Java, o campo “Kind” (tipo do elemento) virá preenchido com “Class” (clcando no combo você verá os diferentes tipos de elemento disponíveis), logo basta informar o nome da sua classe no campo “Name” (nome da classe).

Após, codifique a classe Main (arquivo main.java) para testar a classe **Invoice**.

Para executar o programa, clique com o botão direito do mouse sobre qualquer área do editor e selecionar [Run Main.java]. A saída será apresentada na parte inferior da IDE.

Uma alternativa para executar o projeto é recorrer ao atalho (ponta de seta na cor verde) que fica na barra logo acima do editor. Caso queira apenas compilar o programa (gerar um *build*), sem executá-lo, selecione a opção [Build > Build Project].

ATENÇÃO: seguindo boas práticas de organização de projetos, na criação da classe utilize a URL invertida da organização para compor o *namespace* da classe (Ex.: “br.univali.poo.exercicio1.Invoice”). A classe **Invoice** será criada na estrutura de pacotes “br.univali.poo.exercicio1” e fisicamente a classe criada está pasta “<caminho do seu projeto>\src\br\univali\poo\exercicio1”. Após, programar a classe **Invoice**, usando a mesma lógica acima, crie outra classe para a **InvoiceTest** (que vai ser o Main.java), programe-a e teste. Para executar o programa, clique com o botão direito do mouse sobre qualquer área do editor e selecionar [Run ‘InvoiceTest.main’]. A saída será apresentada na parte inferior da IDE.

2 Crie uma classe chamada **Employee** que inclua três atributos — um primeiro nome (tipo **String**), um sobrenome (tipo **String**) e um salário mensal (**double**). Forneça um construtor que inicializa as três variáveis de instância. Forneça um método *set* e um *get* para cada variável de instância. Se o salário mensal não for positivo, não configure seu valor. Escreva um aplicativo para testar as capacidades da classe **Employee**. Crie dois objetos **Employee** e exiba o salário *anual* de cada objeto. Então dê a cada **Employee** um aumento de 10% e exiba novamente o salário anual de cada **Employee**.

3 Crie uma classe chamada **Date** que inclua três atributos — mês (tipo **int**), dia (tipo **int**) e ano (tipo **int**). Forneça um construtor que inicializa as três variáveis de instância supondo que os valores fornecidos estejam corretos. Ofereça um método *set* e um *get* para cada variável de instância. Apresente um método **displayDate** que exiba mês, dia e ano separados por barras normais (/). Escreva um aplicativo para testar as capacidades da classe **Date**.

PROGRAMAÇÃO ORIENTADA A OBJETOS – EXERCÍCIOS POO + JAVA

4 (*Calculadora de frequência cardíaca alvo*) Ao fazer exercícios físicos, você pode utilizar um monitor de frequência cardíaca para ver se sua frequência permanece dentro de um intervalo seguro sugerido pelos seus treinadores e médicos. Segundo a American Heart Association (AHA) (www.americanheart.org/presenter.jhtml?identifier=4736), a fórmula para calcular a *frequência cardíaca máxima* por minuto é 220 menos a idade em anos. Sua *frequência cardíaca alvo* é um intervalo entre 50-85% da sua frequência cardíaca máxima. [**Observação:** essas fórmulas são estimativas fornecidas pela AHA. As frequências cardíacas máximas e alvo podem variar com base na saúde, capacidade física e sexo da pessoa. **Sempre consulte um médico ou profissional de saúde qualificado antes de começar ou modificar um programa de exercícios físicos.**] Crie uma classe chamada `HeartRates`. Os atributos da classe devem incluir o nome, sobrenome e data de nascimento da pessoa (consistindo em atributos separados para mês, dia e ano de nascimento). Sua classe deve ter um construtor que receba esses dados como parâmetros. Para cada atributo forneça métodos *set* e *get*. A classe também deve incluir um método que calcule e retorne a idade (em anos), um que calcule e retorne a frequência cardíaca máxima e um que calcule e retorne a frequência cardíaca alvo da pessoa. Escreva um aplicativo Java que solicite as informações da pessoa, instancie um objeto da classe `HeartRates` e imprima as informações a partir desse objeto — incluindo nome, sobrenome e data de nascimento da pessoa — calcule e imprima a idade da pessoa (em anos), seu intervalo de frequência cardíaca máxima e sua frequência cardíaca alvo.