|  |  |
| --- | --- |
| **Curso:** Mobile Android 1 | **Unidade e módulo:** POO + JAVA |
| **Matéria da aula:** POO | **Duração da aula:** 3h30min. |
| **Professor:** | **Professor:** |
| **Especialista:** | |
| **Assistente:** | **Assistente:** |
| **Link do módulo no Github:** [Unidade\_01\_Programação\_Orientada\_a\_Objetos](https://github.com/digital-house/clasesMobile/tree/master/Unidad_01_Programacion_Orientada_a_Objetos) | |

|  |
| --- |
| **OBJETIVOS** |
| * Os alunos deverão entender por que surgiu o paradigma de POO e quais diferenças existem em relação ao paradigma anterior. * Os alunos deverão conseguir modelar uma situação problemática usando diagramas de classe. |

|  |
| --- |
| **CONTEÚDOS** |
| 1. Programação Orientada a Objetos: por que surgiu e para que serve? 2. Abstração, classes, objetos, responsabilidades, encapsulamento. 3. UML, diagrama de classes, métodos, atributos 4. Relacionamentos de associação e de uso. 5. Herança e polimorfismo. |

**DESCRIÇÃO DA AULA:**

|  |  |
| --- | --- |
| **Início da aula** | **Duração** |
| **Como os objetivos serão apresentados:**  Os objetivos são apresentados de maneira explícita. O objetivo mínimo desta aula é que os alunos consigam identificar os pilares da programação orientada a objetos e, diante de uma problemática em um contexto, modelar um sistema de objetos que permita apresentar uma solução. | **5 minutos** |
| **Atividade ou pergunta motivadora:**  O que é programação? Comparar a programação com uma receita de cozinha. | **3 minutos** |

|  |  |
| --- | --- |
| **Desenvolvimento da aula** | **Duração** |
| **Introdução a conteúdo novo** |  |
| **Matéria vista na aula anterior:**  Apresentação. |  |
| **Matéria:**  Introdução à Programação Orientada a Objetos |  |
| **Resumo:**  O que é programação? O que é a POO? Como surgiu? O que é um objeto?   * Apresentar o conceito de programação e convidar os alunos a participarem compartilhando suas ideias e opiniões. Comparar a programação com seguir uma receita, usando o exemplo de preparar um ovo frito. * Falar sobre a “crise do software” e o surgimento da POO como solução. * Explicar o que é um objeto e o significado de responsabilidade, atributos, abstração e contexto | **60 minutos** |
| **Perguntas para estimular a reflexão:**   * O que é um objeto? * Existem coisas que não são objetos? (Filosofia; tudo que pode ser definido é um objeto.) |  |
| **Perguntas para verificar a compreensão:**   * O que faz uma cadeira ser uma cadeira? * Quando vemos uma cadeira de um tipo que nunca tínhamos visto antes, como sabemos que é uma cadeira? * Então, por que chamamos vários objetos diferentes de “cadeira”? (Uma “cadeira” é definida a partir da sua responsabilidade.) * Se virmos duas cadeiras iguais, como sabemos que são duas cadeiras, e não uma só mudando de lugar muito rapidamente? (Noção de classe como molde.) * Qual é o propósito de uma cadeira? (Noção de contexto.) |  |
| **Introdução a conteúdo novo** |  |
| **Matéria:**   * UML - Diagrama de classes e relacionamentos |  |
| **Resumo:**   * Mostrar no quadro como representar uma classe (nome, atributos, métodos), a visibilidade de seus componentes, o estado de um objeto (encapsulamento) e os relacionamentos de associação e de uso (parâmetros). | **45min** |
| Usar o seguinte documento, que pré-estabelece os elementos de UML a serem usados ao longo das aulas  Link para o documento:   * [Diagrama de classe - DH Standard UML](https://docs.google.com/document/d/174QIdct9PvScfHwJBFIks79joKfjKzZfsyeEOEdruig/edit)   Para introduzir a linguagem UML, usar como exemplo uma “pessoa”. Mostrar que, para representar uma classe, desenhamos uma caixa dividida em duas partes; na primeira, colocamos os atributos e, na segunda, as responsabilidades.  Perguntar aos alunos: O que modelamos sobre uma pessoa? Quais responsabilidades ela tem?  Vários elementos vão surgir, mas devemos chegar à conclusão de que não conseguimos modelar uma pessoa se não tivermos um contexto.  **Contexto**: Queremos modelar um contribuinte da Receita Federal. Quais responsabilidades um contribuinte tem? E quais atributos?  Focar no atributo “domicílio”. Que tipo de atributo é o “domicílio”?  A maioria dos alunos responderá que “domicílio” pode ser modelado com um *string*. É importante que os alunos entendam que um domicílio deve ser modelado como um objeto separado. Perguntar: O que acontece se quisermos enviar uma notificação jurídica ao contribuinte? Ter o domicílio como *string* é útil para nós?  Quando os alunos entenderem a necessidade de ter um domicílio, explicar o relacionamento do tipo “tem”.  **Exemplo de relacionamento de uso:**  É possível modelar o uso de uma xícara com uma cafeteira.  Uma cafeteira “tem” ou “usa” uma xícara de café?  A cafeteira tem a responsabilidade de preparar o café e recebe como parâmetro uma xícara, em que despeja o conteúdo.  Depois, usar como possível exemplo: Se a cafeteira tivesse copinhos de plástico dentro dela e servisse neles o café, que tipo de relacionamento seria? Nesse caso, seria um relacionamento do tipo “tem”.  Dar exemplos de uso, como o de um reprodutor de DVDs e um DVD. Comentar que o relacionamento do tipo “tem” não faria sentido, porque seria como se o reprodutor viesse de fábrica com DVDs dentro dele e, para ver um filme diferente, precisássemos comprar um reprodutor novo.  Acrescentar a “visibilidade” aos modelos executados. Explicar a diferença entre “público” e “privado” e como é sua notação. |  |
| **Perguntas para verificar a compreensão:** |  |
| **Prática curta** |  |
| Ler o enunciado abaixo para os alunos. Pedir para realizar um modelo UML:  O Banco Santander solicita a modelagem de um novo sistema. Esse sistema será usado para registrar as contas e os clientes do banco.  **Clientes:** os clientes serão identificados por um número de cliente, um sobrenome, um número de RG e um CPF.  **Contas:** as contas do banco permitem fazer depósitos, sacar dinheiro e consultar o saldo. Cada conta está associada a um cliente.  [Enunciado Banco Santander](https://docs.google.com/document/d/1VnOMSnjRMAwVkkTg8aPqFNJBQ5U2BirppKHJJ6xrY48/edit)  Uma vez terminado o exercício, abrir uma discussão com a turma e analisar o modelo. Repassar conceitos vistos no princípio. Para que os alunos entendam o conceito de “encapsulamento”, analisar o atributo “saldo”. O que aconteceria se o atributo saldo fosse público? O que aconteceria se o banco implementasse, na responsabilidade “depositar”, um sistema que informasse à Receita Federal sobre depósitos de valores superiores a 1 milhão de dólares? O que aconteceria se fosse cobrado um imposto sobre cada saque? | **30 minutos** |

|  |  |
| --- | --- |
| **Introdução a conteúdo novo** |  |
| **Matéria:**  Herança e polimorfismo. |  |
| **Resumo:**   * Apresentar a seguinte extensão ao enunciado do banco:   **Conta poupança:** além do saldo, as contas poupança têm uma taxa de juros. Neste tipo de conta, é possível realizar três operações:  ○ Depositar dinheiro: o cliente pode depositar a quantia de dinheiro que quiser.  ○ Sacar dinheiro: o cliente pode sacar dinheiro desde que não supere seu saldo.  ○ Recolher juros: o cliente pode recolher os juros mensais aplicados pela sua conta poupança.  **Conta corrente:** além do saldo, as contas correntes têm um limite autorizado de cheque especial. Neste tipo de conta, é possível realizar três operações:  ○ Depositar dinheiro: o cliente pode depositar a quantia de dinheiro que quiser.  ○ Depositar cheques: o cliente pode depositar cheques. Um cheque tem um valor, um banco emissor e uma data de pagamento.  ○ Sacar dinheiro: o cliente pode sacar dinheiro e, caso não tenha saldo suficiente, usar seu cheque especial.  Perguntar aos alunos: O que eles fariam para adaptar o modelo a essa nova especificação?  Usando esse modelo como exemplo, explicar herança e polimorfismo.  Para reforçar os conceitos de herança e polimorfismo, é possível trabalhar com estes exemplos:   1. Animais -> Os cachorros e gatos comem e fazem barulho. 2. Pessoas -> Enviar a todos os alunos a mensagem “Trabalhar”. Como eles implementariam isso? Do que isso depende? Todos entenderam? | **40 minutos** |
| **Perguntas para estimular a reflexão:**   * Os alunos conseguem pensar em outros relacionamentos de herança? |  |

|  |  |
| --- | --- |
| **Prática curta (Entre 5 e 7 minutos por exercício. Aproximadamente 20 minutos. Especificar as atividades obrigatórias e as complementares.)** |  |
| Apresentar o exercício do Tamagotchi. A ideia não é terminá-lo em sala, mas sim que os alunos pensem sobre o exercício até a próxima aula. (Exercício no SGE.) | **20 minutos** |

|  |  |
| --- | --- |
| **Encerramento** |  |
| Usar o quadro para reforçar os conceitos vistos. | **7 minutos** |