

FIAP GRADUAÇÃO

Programação Orientada a Objetos - POO



Antes de explicar vamos entender o porquê da utilização:

- Padronização no desenvolvimento
 - Facilita o entendimento do código e com isso a manutenção e continuidade
 - Software/sistema é criado e depois é melhorado ou aumentado?
 - Sim e com certeza, ou seja...
 - Se para fazer uma ponte é necessário engenharia e aprovações do projeto o porquê um projeto de software não?
 - Com base na pergunta temos a necessidade de aprender
- Programação Orientada a Objetos (POO)

Antes de explicar vamos entender o porquê da utilização:

- Padronização no desenvolvimento
 - Facilita o entendimento do código e com isso a manutenção e continuidade
 - Software/sistema é criado e depois é melhorado ou aumentado?
 - Sim e com certeza, ou seja...
 - Se para fazer uma ponte é necessário engenharia e aprovações do projeto o porquê um projeto de software não?
 - Com base na pergunta temos a necessidade de aprender
- Programação Orientada a Objetos (POO)

- Programação
 - Linguagem de programação é algo padronizado com uma sintaxe para determinado fim, vale observar que a interface de desenvolvimento (IDE) pode ser escolhida, dependendo da linguagem de programação, por exemplo o Java pode ser utilizado o Eclipse, NetBeans ou outras IDEs.
 - Programar é codificar algo da necessidade do nosso mundo para um programa, sendo para fins de automações ou até mesmo por exemplo na criação de uma loja virtual ao invés ou em complemento de uma loja física.
- Agora começa o primeiro conceito da POO: Objeto
 - Representa uma entidade física, conceitual ou de software (vamos trabalhar bastante nesse tópico).
- Programação Orientada a Objetos
 - Conhecido como um paradigma de programação, feito para a comunicação dos estados da programação para facilitar a padronização.

(Atributos: Material, tipo, tamanho e cor)

**Atributos, Características
ou
Propriedades**



material: *tecido*
tipo: *passeio*
tamanho: *41*
Cor principal: *preto*



material: *plástico*
tipo: *futebol de salão*
tamanho: *42*
Cor principal: *amarelo*



material: *couro*
tipo: *social*
tamanho: *40*
Cor principal: *preto*

Quando se fala em abstração da orientação a objetos é em resumo entender o que o sistema precisa fazer para criar um objeto que é o objeto por exemplo apresentado anteriormente que é o calçado e como fazer com que seja um objeto com atributos e métodos no software/sistema.

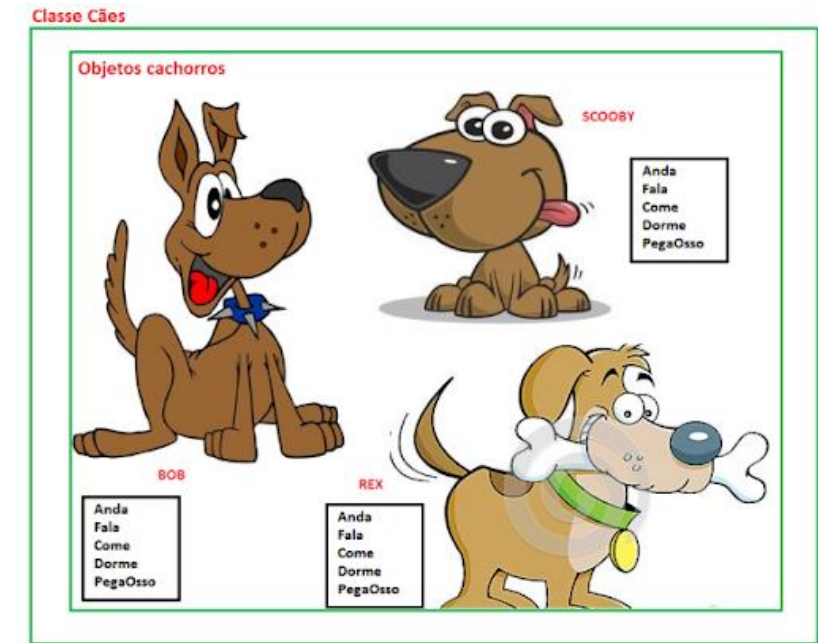
No momento de criar a classe deve ser considerado apenas o necessário dos aspectos do objeto.

A classe que será criada na programação em POO para refletir o objeto mapeado é também conhecido como objetos ou instâncias de classes, ou seja, estará disponível, quando chamado por outra classe irá instanciar (criar) um objeto, baseado no exemplo do sapato pode instanciar/criar um tênis, chuteira ou sapato social.

As abstrações são representadas pelas classes.

Uma classe deve conter apenas os elementos necessários para resolver um aspecto bem definido do sistema.

A classe é uma descrição nomeada para um grupo de entidades (chamadas de objetos ou instâncias de classe) que têm as mesmas características.



Fonte: https://www.google.com/url?sa=i&url=http%3A%2Ffabrica.ms.senac.br%2F2015%2F03%2Fclasses-objetos-atributos-e-metodos-em-java%2F&psig=AOvVaw02RCySacm4suNH_VlvRud&ust=1640823151161000&source=images&cd=vfe&ved=0CA5QJRqfwotCKD&4Nzch_UCFQAAAAAABAD

- Em resumo um objeto é uma instância de uma classe, mas para entender isso se torna necessário lembrar do calçado, então na programação é feito um código que é por exemplo um arquivo do tipo classe que no caso do Java é um arquivo .java que tem as características que são os atributos na programação (cor, tamanho, etc) e os métodos que são as ações, por exemplo em um caso de ação de um calçado no sistema pode ser cadastrar calçado, alterar tamanho ou cor.

Classe

Curso

Propriedades / atributos

Nome
Local
Dias oferecidos
Carga horária
Hora de Início
Hora de Término



Comportamentos / métodos / funções

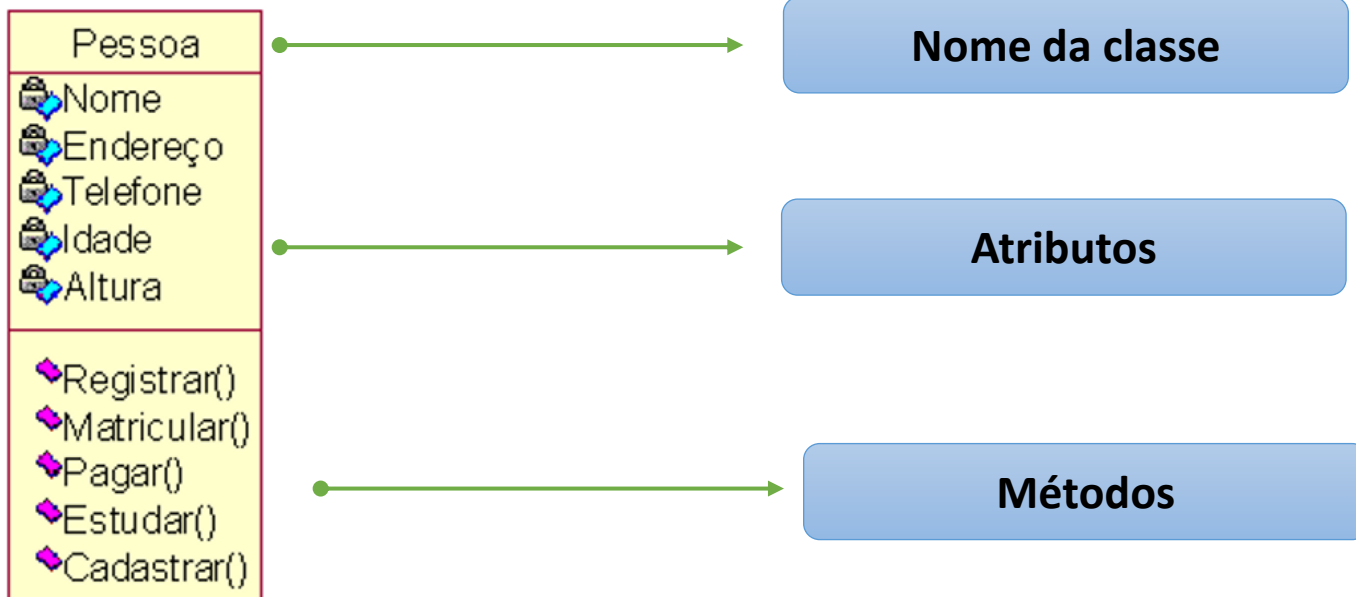
Adicionar um aluno
Excluir um aluno
Obter lista de alunos
Verificar se está cheio

Objetos:

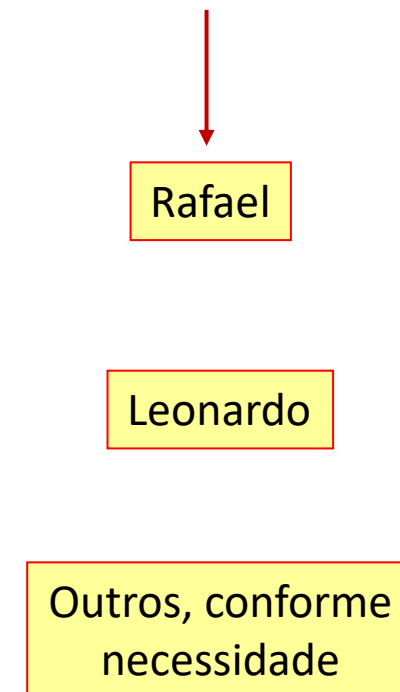
- analiseDeSistemas
- desenvolvimentoWeb
- engenhariaDaComputacao

- Com um diagrama de classes (UML) é possível representar uma classe.
- Para representar uma classe é necessário ter um retângulo com três divisões, conforme a primeira imagem abaixo à esquerda em amarelo:

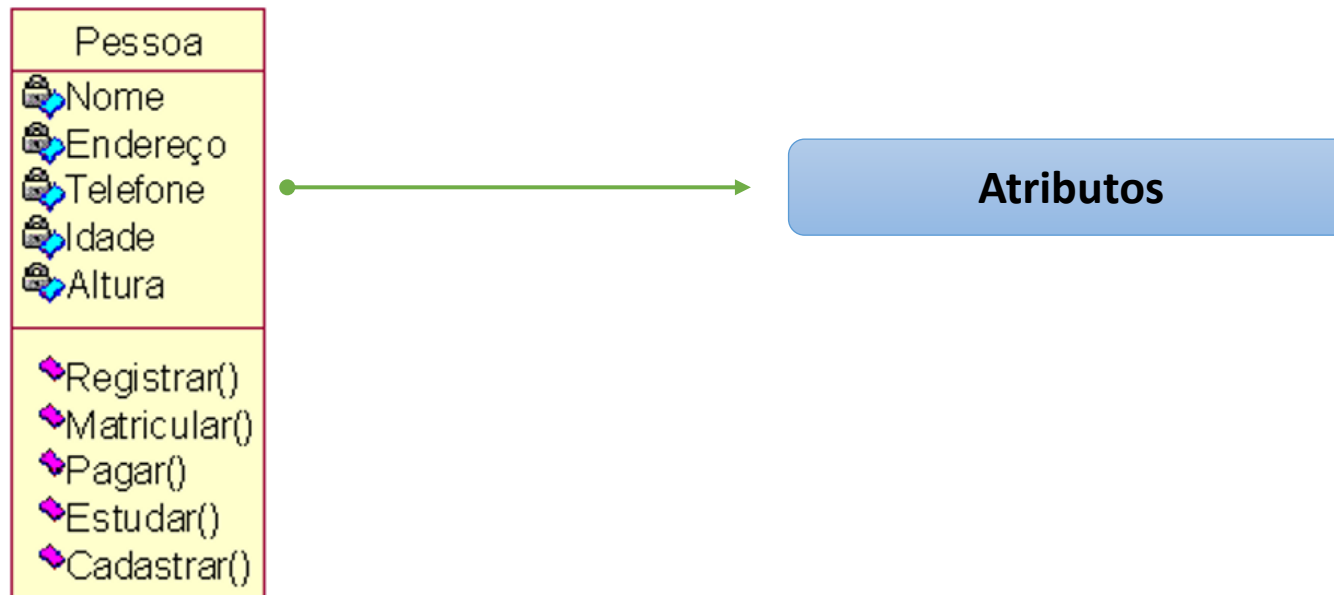
Classe



Possíveis objetos da classe

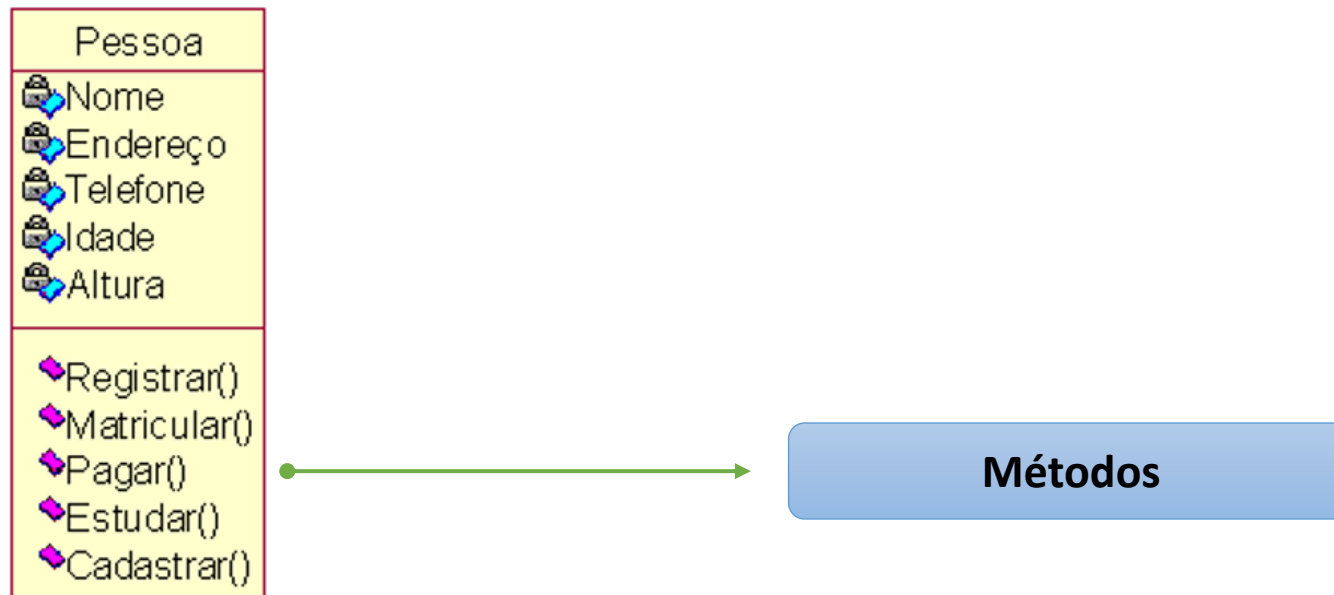


- São as características do objeto, como apresentado abaixo para o objeto pessoa nesse caso pode ser nome, endereço, telefone, idade e altura.



- Estão ligados as características, então isso pode ser por exemplo tamanho que na programação é número que está ligado em java a double ou se for um número inteiro int, também pode ser o nome que na programação em Java é String.
 - Existem outros tipos de atributos que geram a necessidade de outros tipos de características na definição durante o desenvolvimento da programação em Java, que durante o ano iremos utilizar cada um, sendo eles:
 - boolean
 - byte
 - char
 - short
 - int
 - float
 - long
 - double

- Os métodos que também são conhecidos como funções, são as ações que essa classe pode ter que irá fazer alguma ação, no caso do exemplo abaixo a classe pessoa pode realizar as seguintes ações:
 - Registrar, Matricular, Pagar, Estudar e Cadastrar.



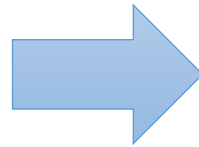
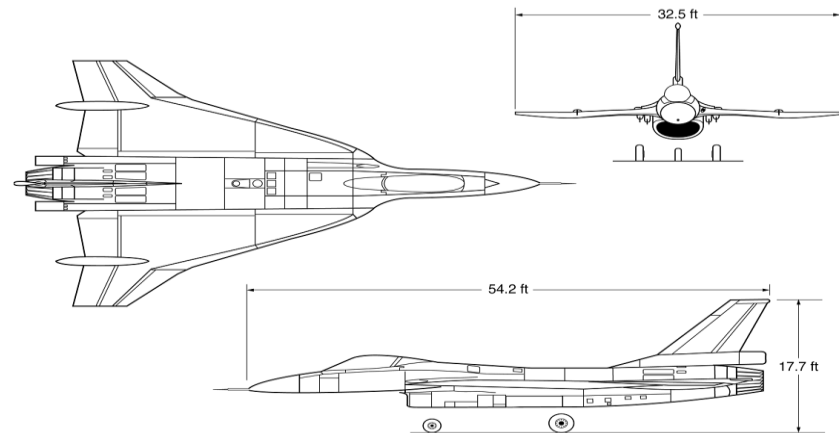
DEFINIÇÃO – ORIENTAÇÃO À OBJETOS: Objeto e classe



Fonte: [Link](#)

O QUE É UM MODELO VISUAL?

Um modelo é simplificação da realidade.



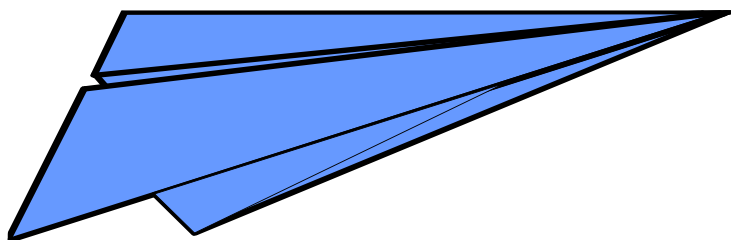
- Nasce do princípio que não é possível entender e fazer um sistema conforme o cliente ou usuário pensou ou precisa.
- Durante o desenvolvimento do modelo visual que parte desse modelo é a documentação e o UML são feitas as validações de quem gerou a necessidade do sistema e então se fazem os ajustes e são alinhadas as expectativas versus as realidades do que poderá ser entregue.
- Lembre-se que antes de iniciar a construção de um prédio o arquiteto e/ou engenheiro valida a planta, pois na validação irá saber se atende a necessidade e expectativa da demanda.

A IMPORTÂNCIA DE SE FAZER A MODELAGEM VISUAL

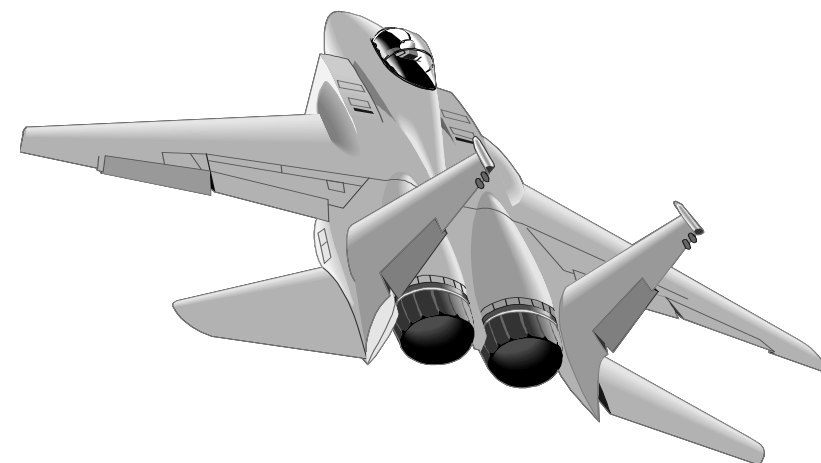
FIAP

Menos Importante

Mais Importante



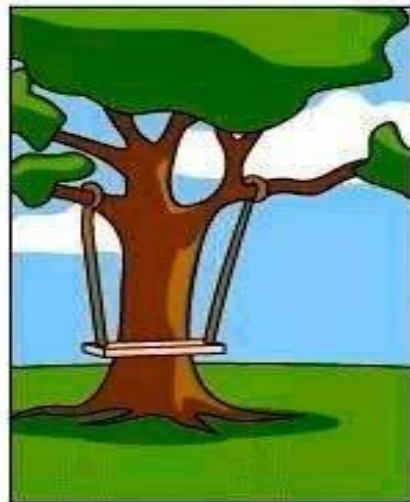
Avião de papel



Avião à Jato



Como o cliente explicou...



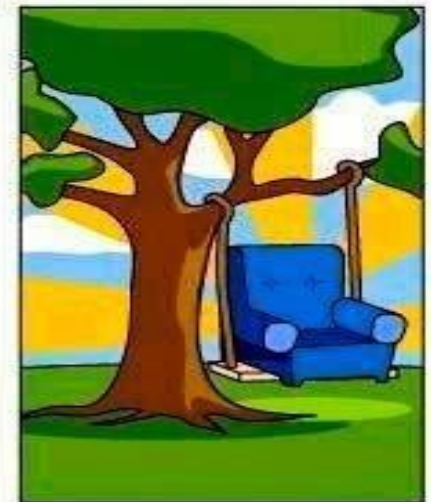
Como o líder de projeto entendeu...



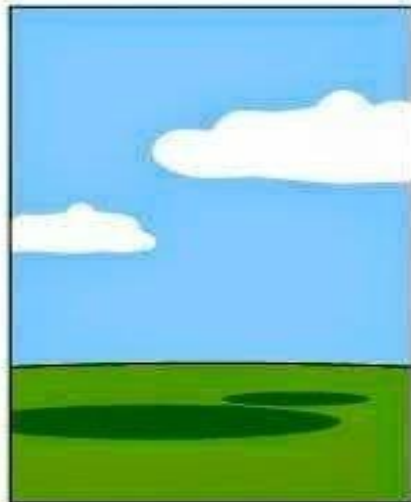
Como o analista projetou...



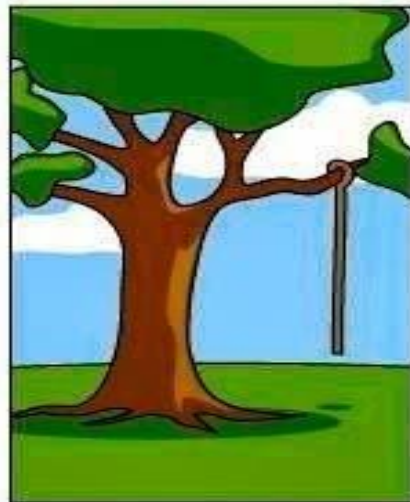
Como o programador construiu...



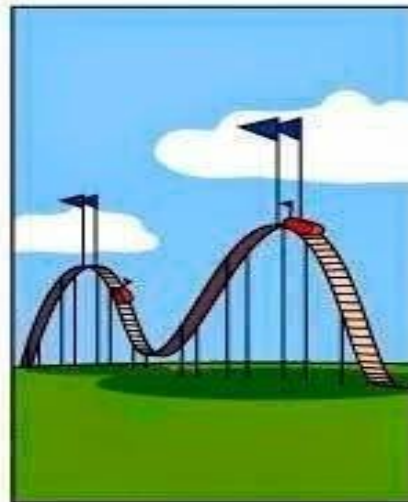
Como o Consultor de Negócios descreveu...



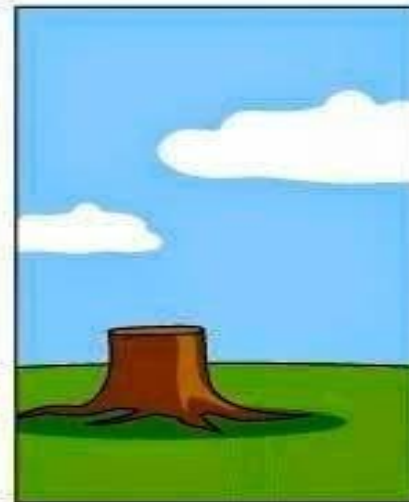
Como o projeto foi documentado...



Que funcionalidades foram instaladas...



Como o cliente foi cobrado...



Como foi mantido...



O que o cliente realmente queria...

