

## MÓDULO 03 – PRÁTICA DE LABORATÓRIO I

### 1 INTRODUÇÃO

Nos tópicos a seguir **estaremos praticando os conceitos visto no módulo 3 de Programação I**. Revise estes conceitos nos seguintes capítulos do livro de apoio de Programação I:

- Capítulo 04 – CLASSES E OBJETOS.

Na sequência, leia atentamente o resumo dos conceitos e os exemplos apresentados neste documento e tire suas dúvidas com o tutor no fórum de dúvidas.

**Atenção:** Para que você possa comparar as suas classes, com as classes desenvolvidas neste material, faça o download deste projeto que está disponível para download no módulo 03.

### 2 CLASSES E OBJETOS

A orientação a objetos traz muitas vantagens para os desenvolvedores de aplicações. As principais vantagens que podemos citar são:

- Estrutura melhor os dados que serão manipulados na aplicação;
- Facilita a reutilização de código;
- Por estruturar melhor o código da aplicação, permite agilidade nas manutenções.

Porém para trabalhar com programação orientada a objetos é imprescindível compreender no mínimo os dois conceitos básicos da orientação a objetos: **as classes e os objetos**.

Podemos definir **as classes como as estruturas que são utilizadas para criar os objetos**. Cada objeto criado por meio de uma classe possuirá a mesma estrutura desta classe.

As classes então, poderão criar os objetos e definir quais são os atributos e métodos (comportamentos) que o objeto irá possuir. Os atributos e métodos do objeto são aqueles que estão definidos na estrutura da classe.

Um atributo é equivalente a uma variável, porém, diferente das variáveis, todos os atributos estarão armazenados em um mesmo local, junto com as demais características do objeto.

Já os métodos, são todas as funcionalidades definidas na classe e habilitadas para o objeto criado poder utilizar.

#### 2.1 EXEMPLO DA ESTRUTURA DE UMA CLASSE

O exemplo abaixo apresenta a estrutura básica de uma classe chamada computador, que poderá criar um ou mais objetos do tipo computador. Um objeto criado por meio desta classe possuirá os atributos e métodos definidos nesta estrutura.

Classe Computador

**//atributos da classe = características dos objetos que serão criados**

modeloProcessador  
capacidadeProcessador  
quantidadeMemoriaRAM  
tamanhoHD

**// métodos da classe = comportamentos dos objetos que serão criados**

Cadastra Computador  
Altera dados Computador  
Exibe dados Computador

As classes são compostas pela seguinte estrutura:

- **Atributos** = os atributos ou variáveis de instância armazenam as propriedades (características) dos objetos;
- **Construtor** = o construtor inicializa os atributos do objeto quando o objeto é instanciado pelo operador **new**;
- **Métodos** = são os serviços oferecidos pelos objetos de uma determinada classe. Os métodos atuam sobre os objetos e são responsáveis pela realização das tarefas.

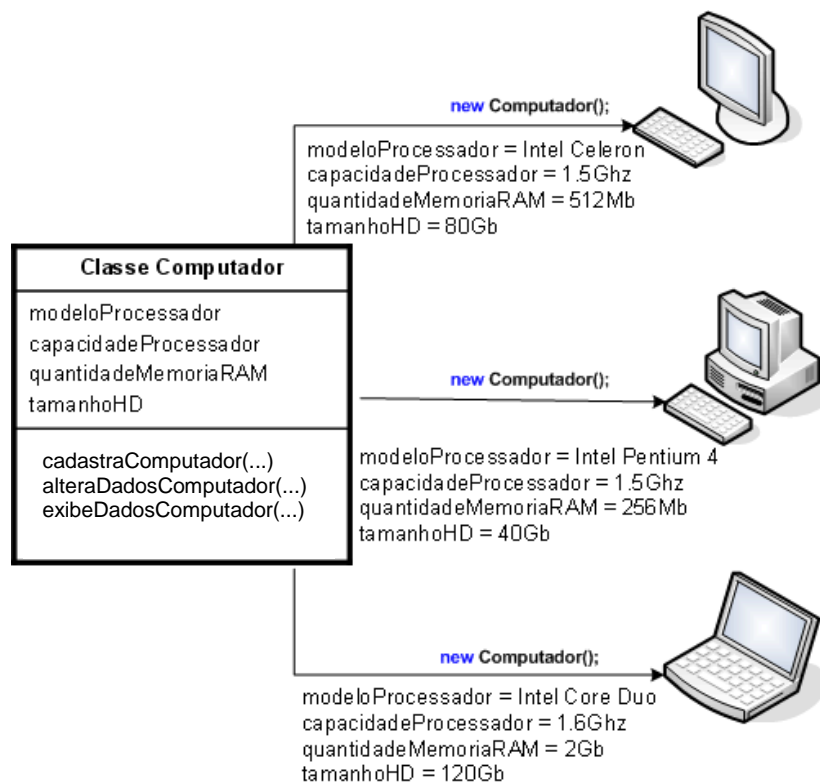
Os atributos de uma classe serão os atributos que irão manter os dados de um objeto criado por meio desta classe. Os métodos por sua vez, são todos os métodos que irão permitir ao objeto realizar operações com os dados armazenados nos atributos. O construtor (ou método construtor) é executado na criação de um objeto, por isso seu nome construtor, pois ele executa as instruções do método durante a construção do objeto. Assim quando um objeto é criado, se a classe possuir um método construtor definido, este método será executado neste momento.

## 2.2 CRIANDO OBJETOS POR MEIO DAS CLASSES

Analisando a figura abaixo podemos verificar que a mesma classe Computador criou três objetos computadores diferentes. Cada objeto computador criado possui propriedades (características) específicas, porém todos possuem os mesmos atributos e as mesmas funcionalidades.

O comando que permite a criação de novos objetos por meio de uma classe definida é o: **new NomeClasse()**.

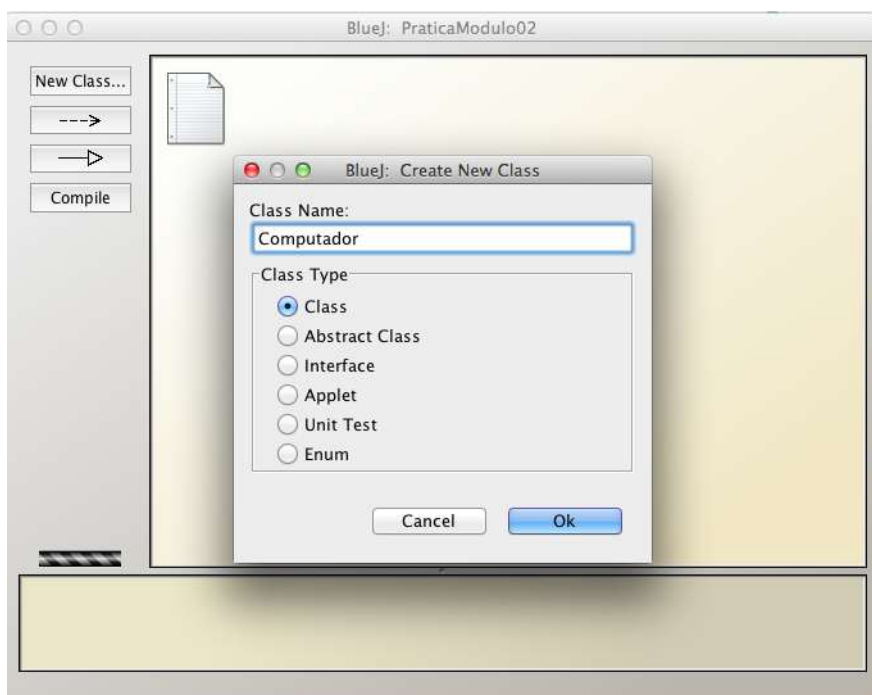
Quando um objeto é criado por meio do operador: **new**, geralmente é fornecido um nome para o objeto. Este nome será utilizado para a chamada dos métodos deste objeto - métodos estes que foram adquiridos da estrutura da classe que o criou.



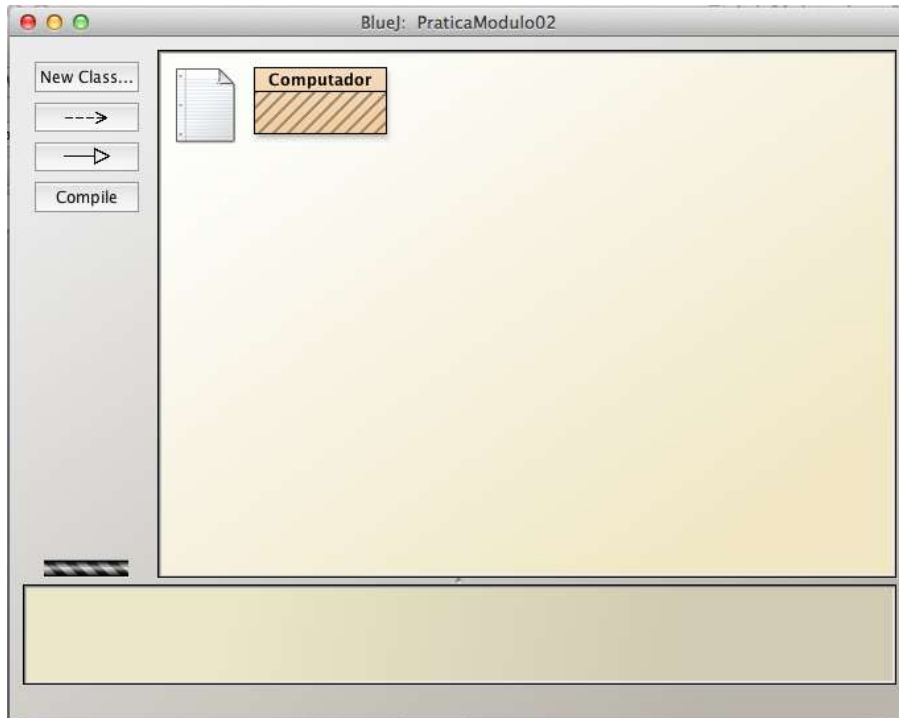
## 2.3 PRIMEIRO EXEMPLO CRIANDO CLASSES EM JAVA

Vamos iniciar a prática verificando como criar a classe: Computador em Java. Para tanto, abra o BlueJ e crie um novo projeto com o nome: PraticaModulo03. **Importante:** caso tenha dúvidas sobre como criar um novo Projeto no BlueJ, revise o material disponibilizado no Módulo 01 de Laboratório I, e/ou verifique o vídeo: **Criando Classes e Objetos no BlueJ**, disponibilizado no módulo 03.

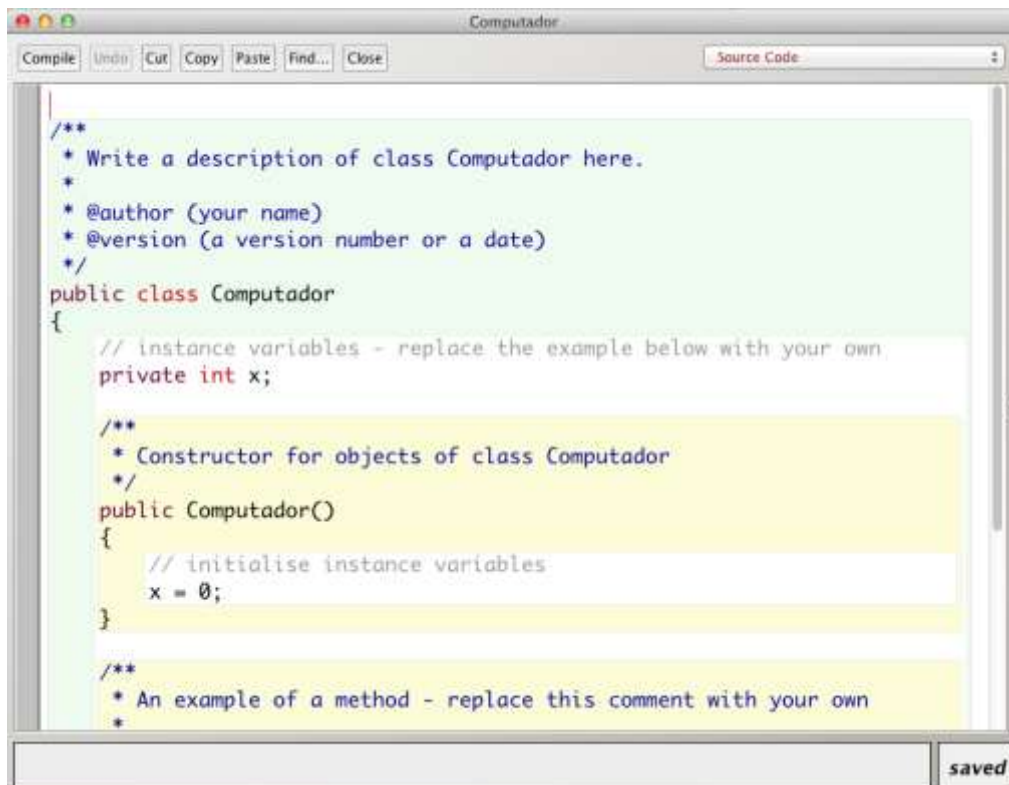
Crie uma classe chamada: **Computador**. Deixe a opção Class marcada e clique em OK.



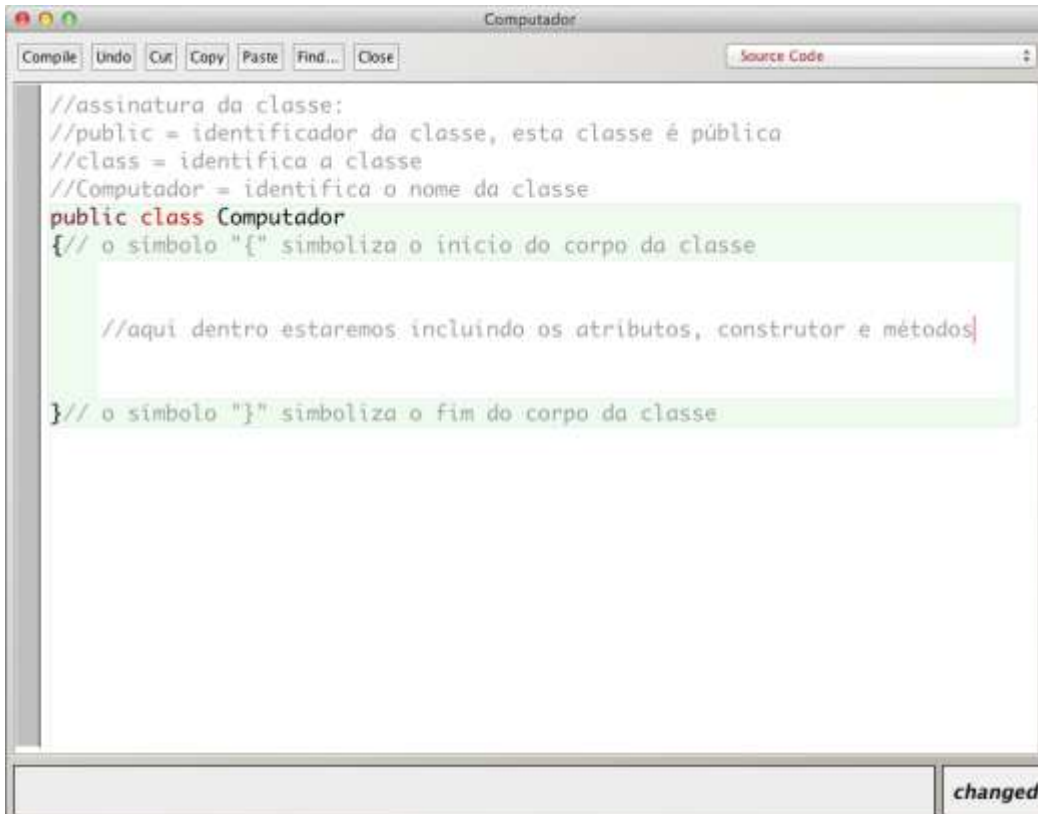
A classe criada será apresentada na tela do BlueJ conforme a figura abaixo. Clique duas vezes em cima da classe para editá-la.



Será aberta uma nova janela com um conteúdo padrão conforme figura abaixo.



Apague todo o conteúdo padrão deixando apenas a estrutura principal da classe, conforme imagem abaixo.



```
//assinatura da classe:
//public = identificador da classe, esta classe é pública
//class = identifica a classe
//Computador = identifica o nome da classe
public class Computador
{
    // o símbolo "{" simboliza o início do corpo da classe

    //aqui dentro estaremos incluindo os atributos, construtor e métodos

}
// o símbolo "}" simboliza o fim do corpo da classe
```

Vamos agora criar os atributos da classe: modeloProcessador, capacidadeProcessador, quantidadeMemoriaRAM, tamanhoHD. Para isso precisamos **declarar os atributos**. Para fazermos a declaração dos atributos utilizamos a seguinte estrutura:

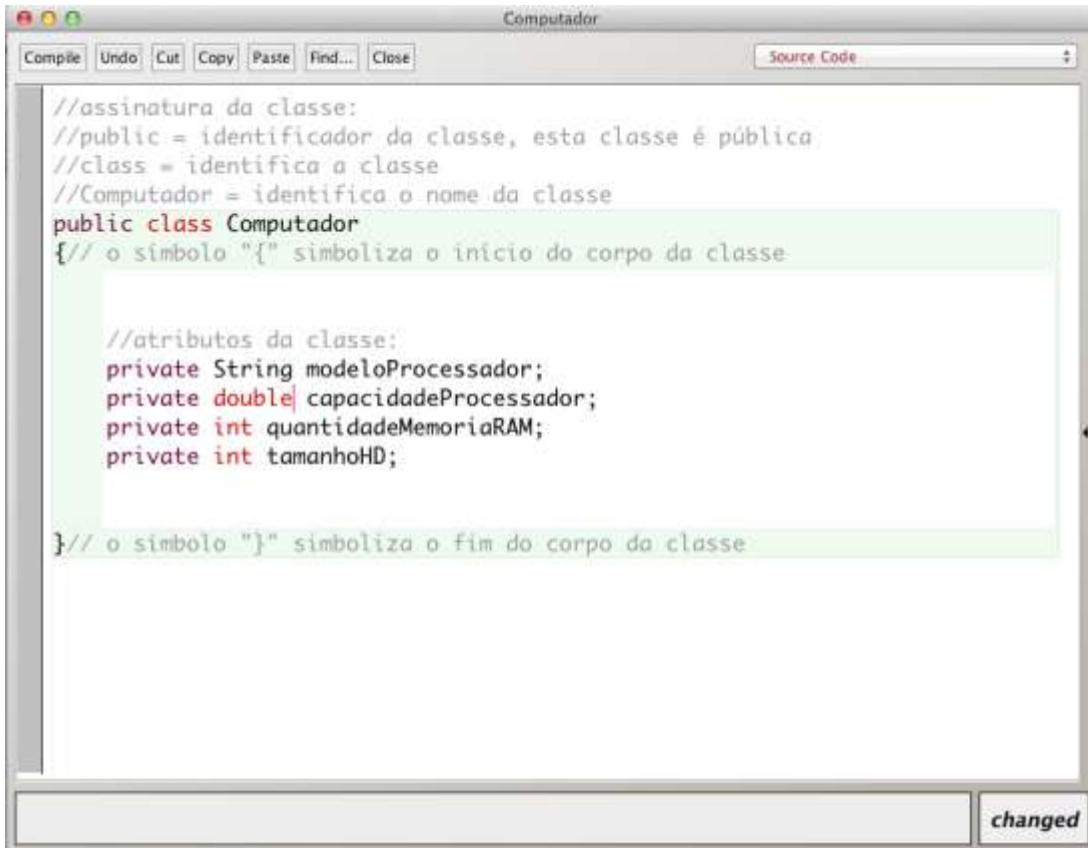
[identificador do atributo] [tipo de dados] [nome do atributo];

Por exemplo:

private String modeloProcessador;

**Importante:** revise o capítulo 04 – Classes e Objetos do livro de apoio de Programação I para verificar os detalhes sobre os atributos.

Vamos então incluir cada atributo da classe Computador no BlueJ, conforme imagem abaixo:

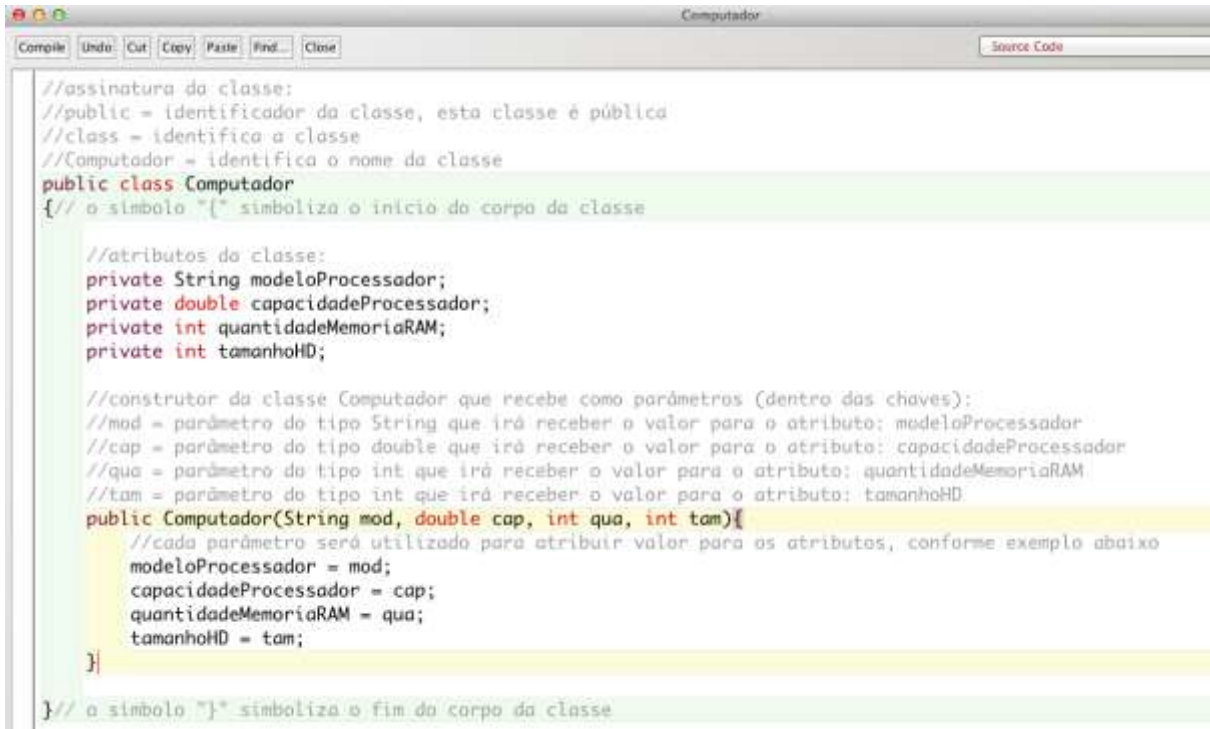


```
//assinatura da classe:  
//public = identificador da classe, esta classe é pública  
//class = identifica a classe  
//Computador = identifica o nome da classe  
public class Computador  
{// o simbolo "{" simboliza o inicio do corpo da classe  
  
    //atributos da classe:  
    private String modeloProcessador;  
    private double capacidadeProcessador;  
    private int quantidadeMemoriaRAM;  
    private int tamanhoHD;  
  
}// o simbolo "}" simboliza o fim do corpo da classe
```

changed

Na sequência, vamos criar o método construtor, para permitir a criação de objetos do tipo: Computador. O método construtor irá receber os valores referente às características de cada objeto, por meio de parâmetros. Estes valores, recebidos pelos parâmetros, serão armazenados nos atributos do objeto que está sendo criado.

**Importante:** revise o capítulo 04 – Classes e Objetos do livro de apoio de Programação I para verificar os detalhes sobre o método construtor e os parâmetros dos métodos das classes.

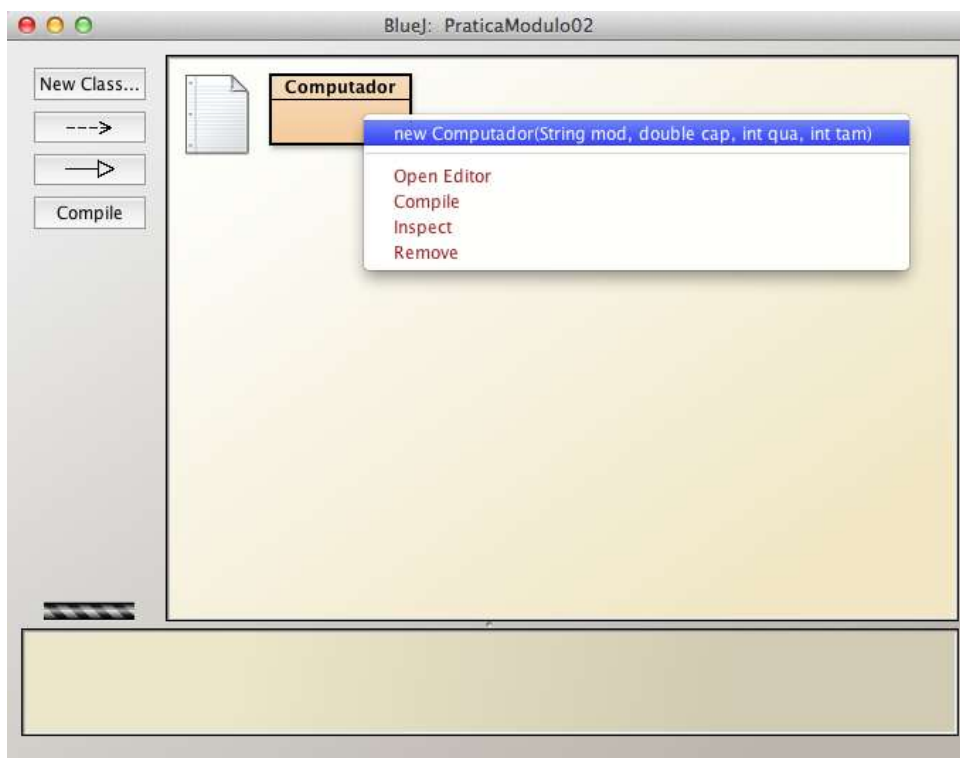


```
//assinatura da classe:
//public = identificador da classe, esta classe é pública
//class = identifica a classe
//Computador = identifica o nome da classe
public class Computador
{
    //atributos da classe:
    private String modeloProcessador;
    private double capacidadeProcessador;
    private int quantidadeMemoriaRAM;
    private int tamanhoHD;

    //construtor da classe Computador que recebe como parâmetros (dentro das chaves):
    //mod = parâmetro do tipo String que irá receber o valor para o atributo: modeloProcessador
    //cap = parâmetro do tipo double que irá receber o valor para o atributo: capacidadeProcessador
    //qua = parâmetro do tipo int que irá receber o valor para o atributo: quantidadeMemoriaRAM
    //tam = parâmetro do tipo int que irá receber o valor para o atributo: tamanhoHD
    public Computador(String mod, double cap, int qua, int tam){
        //cada parâmetro será utilizado para atribuir valor para os atributos, conforme exemplo abaixo
        modeloProcessador = mod;
        capacidadeProcessador = cap;
        quantidadeMemoriaRAM = qua;
        tamanhoHD = tam;
    }
}
```

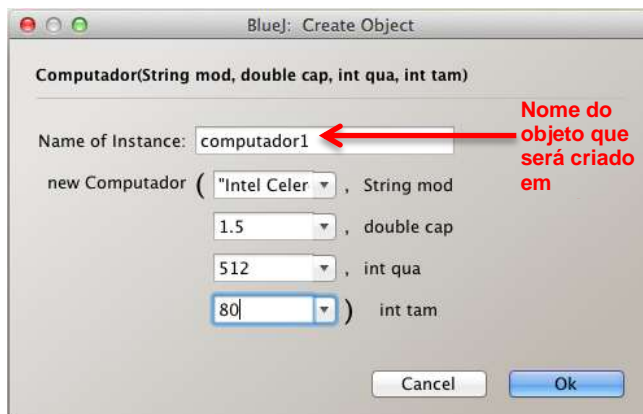
A partir de agora, podemos compilar a classe e utilizá-la para a criação de objetos do tipo: Computador. Vamos então criar os três objetos apresentados na figura do tópico: 2.2 CRIANDO OBJETOS POR MEIO DAS CLASSES.

Para criarmos objetos por meio da classe: Computador devemos clicar com o botão direito em cima da classe e selecionar a opção: new Computador(...), conforme ilustra a figura abaixo:



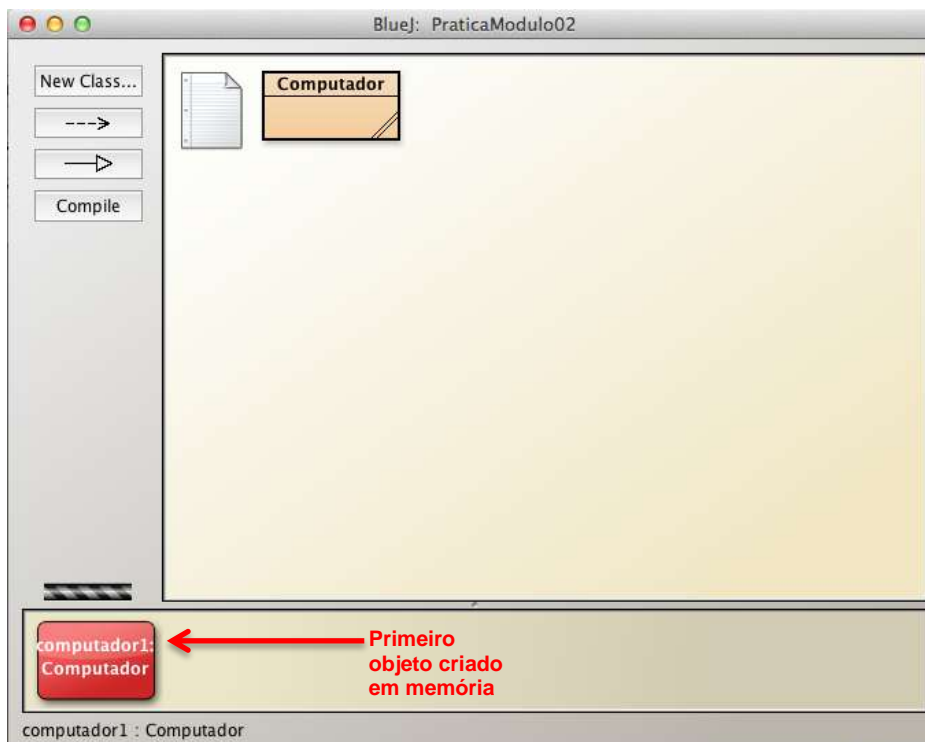
Esta opção indica que estaremos utilizando o operador: **new** para criarmos um novo objeto do tipo Computador. Neste momento, o método construtor que criamos dentro da classe: Computador é chamado para permitir o recebimento dos valores, referente as características de cada objeto.

Após selecionarmos a opção indicada acima, será aberta uma nova janela para que possamos informar os valores, que serão enviados por meio dos parâmetros, para o construtor da classe. Devemos então, informar os valores das características de cada objeto. De acordo com a figura do tópico: 2.2 CRIANDO OBJETOS POR MEIO DAS CLASSES, o primeiro objeto possui as seguintes características:



**Atenção:** os valores do tipo: String devem ser informados entre “aspas duplas”.

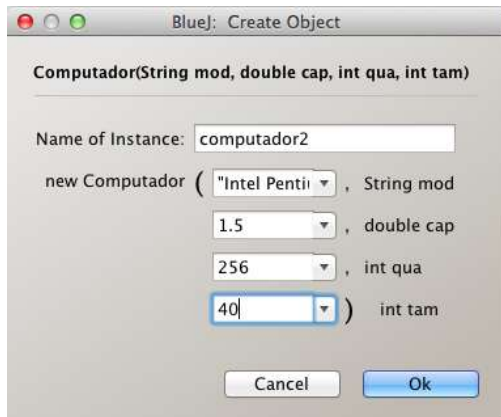
Após informar os valores do primeiro objeto, clique no botão: [OK]. O objeto criado aparecerá na barra inferior da tela do projeto no BlueJ, conforme ilustra a figura abaixo.



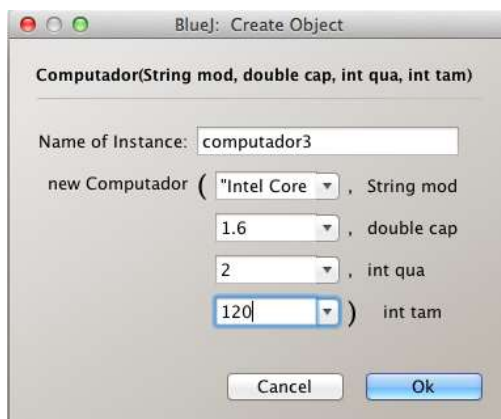


Vamos então criar os outros dois objetos, utilizando a mesma classe: Computador. Para tanto, utilizaremos o mesmo procedimento: clicar com o botão direito em cima da classe e selecionar a opção: new Computador(...). Os valores dos dois novos objetos são:

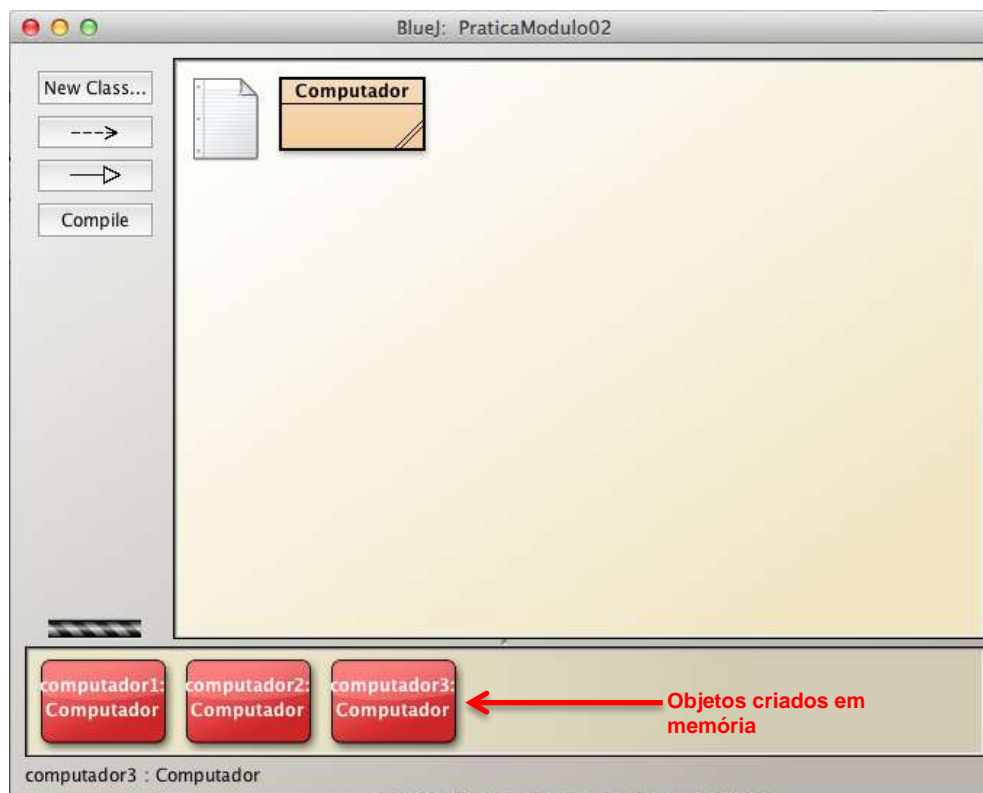
Segundo objeto:



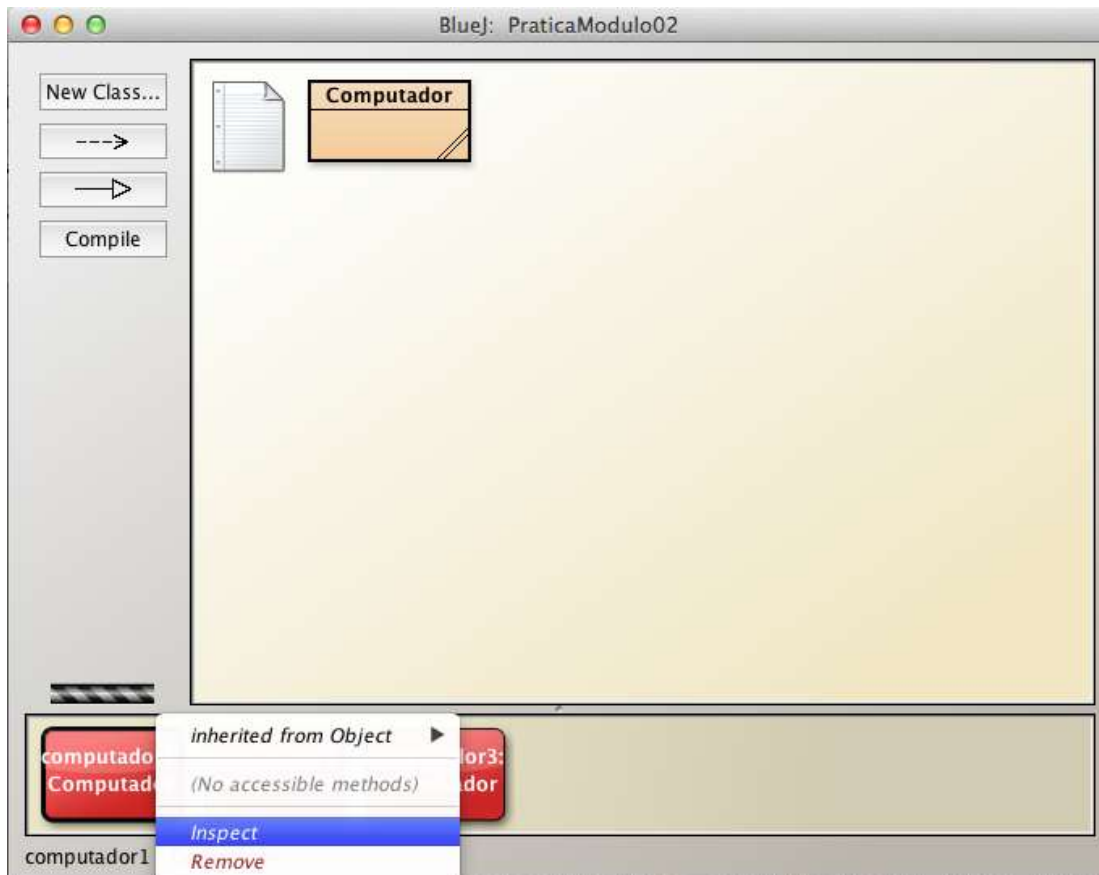
Terceiro objeto:



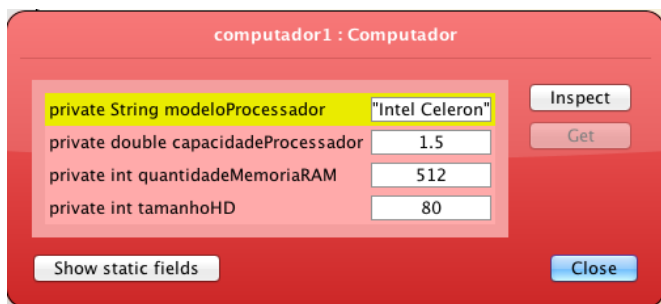
Após a criação dos três objetos a tela do BlueJ irá apresentar os três objetos criados em memória, conforme ilustra a figura abaixo:



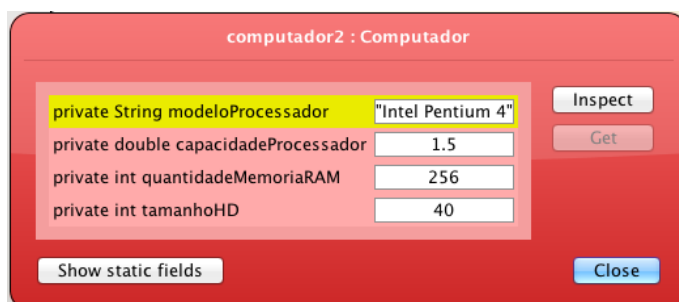
Para verificarmos os dados de cada objeto, e confirmarmos que cada objeto foi criado com base na mesma estrutura de objetos, porém, cada um com um valor específico para as suas propriedades, podemos clicar com o botão direito em cima de cada objeto e selecionar a opção: **Inspect**. Por meio desta opção conseguimos visualizar o valor de cada atributo de cada objeto.



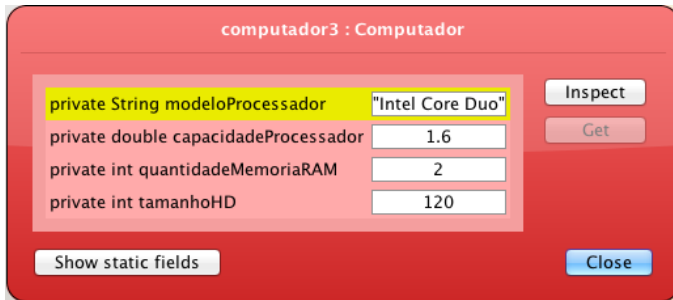
Valores das propriedades do Primeiro objeto (computador1):



Valores das propriedades do Segundo objeto (computador2):



Valores das propriedades do Terceiro objeto (computador3):



computador3 : Computador	
private String modeloProcessador	"Intel Core Duo"
private double capacidadeProcessador	1.6
private int quantidadeMemoriaRAM	2
private int tamanhoHD	120

Buttons: Inspect, Get, Show static fields, Close

Conforme verificamos neste exemplo, por meio de uma estrutura única e padrão, como a classe, podemos criar inúmeros objetos. E cada objeto criado poderá ter valores diferentes para suas propriedades.