

Programação Orientada a Objetos

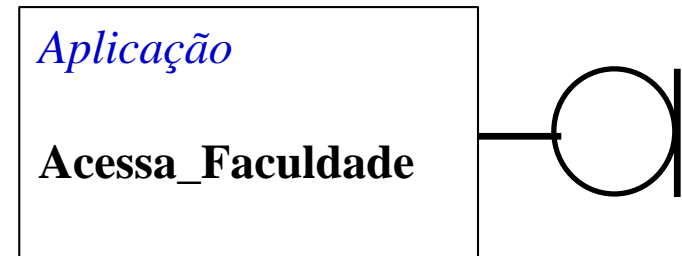
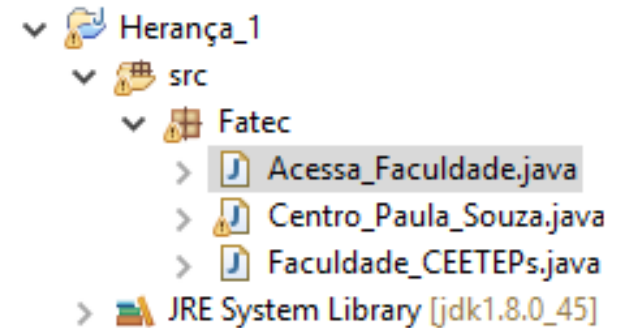
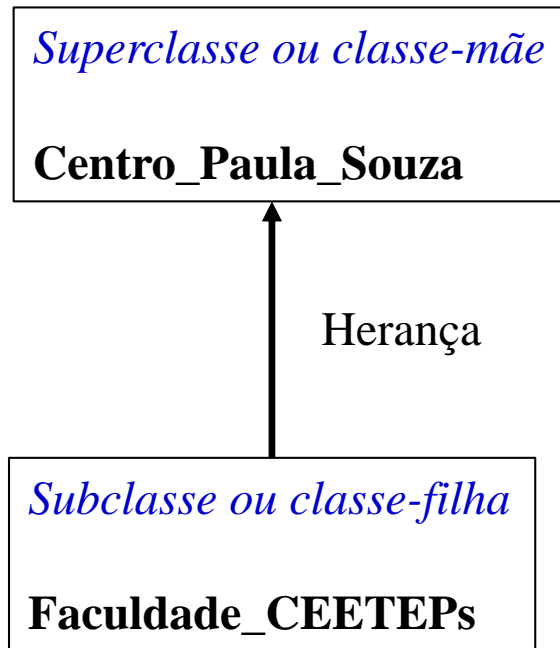
Herança

profº Mauricio Conceição Mario

Conceitos de Orientação a Objetos

- **Herança:** ocorre quando uma classe passa a herdar características (atributos e métodos) definidas em outra classe, especificada como *ancestral* ou *superclasse*. A classe receptora de recursos é denominada subclasse.

Representação de Herança em diagrama de classes



```
Figuras_geom... Centro_Paul... Faculdade_C... ACESSA_Facul... »99+
1 package Fatec;
2
3 public class Centro_Paula_Souza {
4     private String Fatec;
5     private String cidade;
6
7     public Centro_Paula_Souza(String cidade){
8         this.cidade = cidade;
9         System.out.println("método construtor da superclasse \n" + "cidade: " + cidade);
10    }
11
12    public void setFatec(String Fatec){
13        this.Fatec = Fatec;
14    }
15
16    public String getFatec(){
17        return Fatec;
18    }
19
20 }
21
```

```
1 package Fatec;
2
3 public class Faculdade_CEETEPs extends Centro_Paula_Souza{
4     private String faculdade;
5
6     public Faculdade_CEETEPs(){
7         super("Santos"); /*o método super() acessa o método da
8                             classe com hierarquia imediatamente superior na relação de
9                             herança*/
10    }
11
12    public void setFaculdade(String faculdade){
13        this.faculdade = faculdade;
14    }
15
16    public String getFaculdade(){
17        return faculdade;
18    }
19
20 }
```

Figuras_geom... Centro_Paul... Faculdade_C... Acessa_Facul... »99+

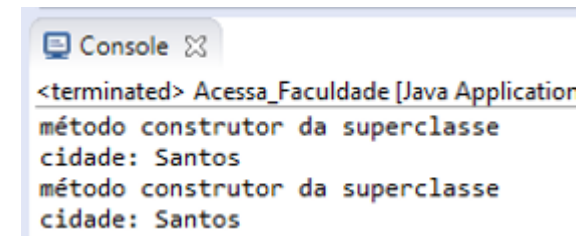
```
1 package Fatec;
2
3 public class Acessa_Faculdade {
4
5     public static void main(String args[]){
6         Faculdade_CEETEPs ceetps = new Faculdade_CEETEPs();
7         /*a variável de instância tem acesso às classes
8          * de relação de herança de hierarquia superior*/
9
10        ceetps.setFatec("Fatec Baixada Santista");
11        System.out.println("Fatec \t"+ ceetps.getFatec());
12    }
13
14 }
15
```

Console

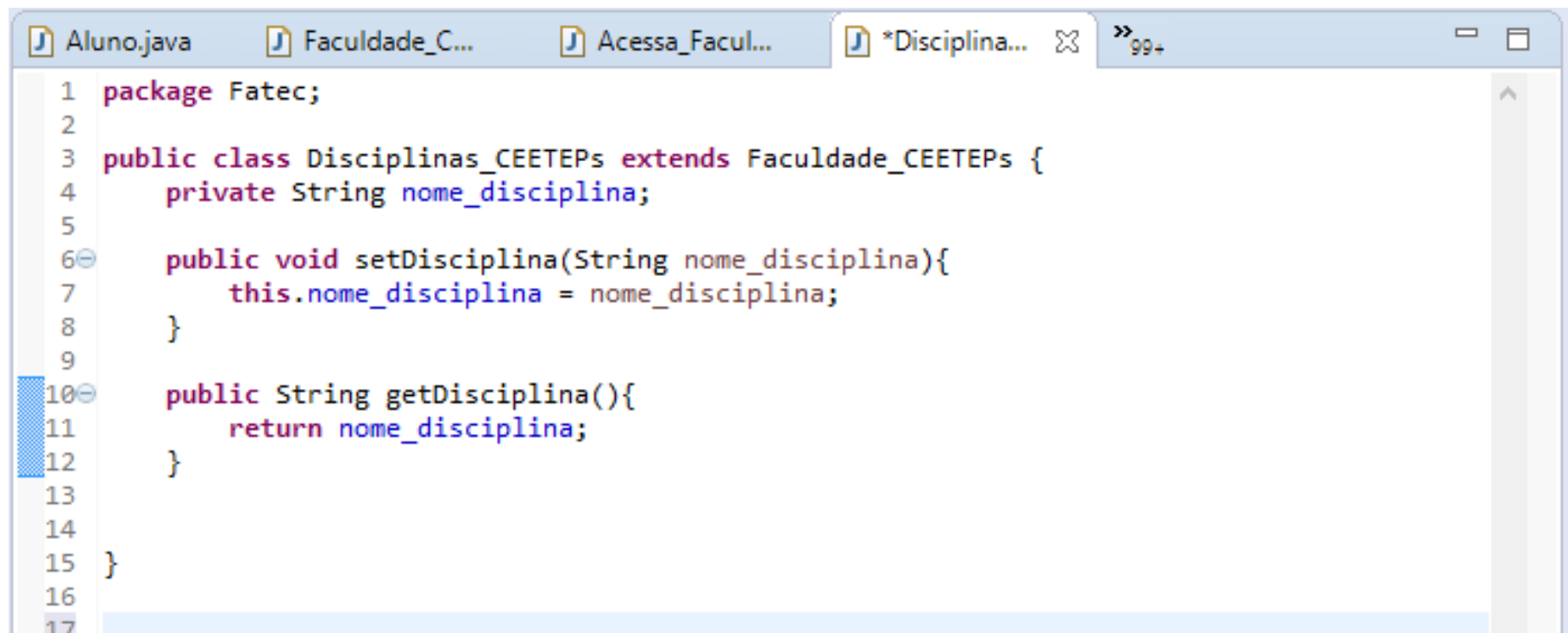
<terminated> Acessa_Faculdade [Java Application] C:\Arquivos de Programas\Java\jdk1.8.0_45\bin\javaw.exe (11 de mar de 2018 1
método construtor da superclasse
cidade: Santos
Fatec Fatec Baixada Santista

Exercícios:

1. Editar, compilar e executar as classes da aplicação relativa às unidades do Centro Paula Souza.
2. Adicionar à aplicação uma classe chamada “Disciplinas_CEETEPs”, com extensão para Faculdade_CEETEPs, com os atributos “nome_disciplina” do tipo string.
3. Prover as informações sobre o novo atributo através da classe “Acessa_Faculdade”.
4. Analisando o exemplo ao fim do arquivo, justificar a execução do método construtor da classe “Centro_Paula_Souza” por 2 vezes:



```
Console
<terminated> Acessa_Faculdade [Java Application]
método construtor da superclasse
cidade: Santos
método construtor da superclasse
cidade: Santos
```



The image shows a screenshot of an IDE window with four tabs: 'Aluno.java', 'Faculdade_C...', 'Acessa_Facul...', and '*Disciplina...'. The active tab is '*Disciplina...'. The code in the editor is as follows:

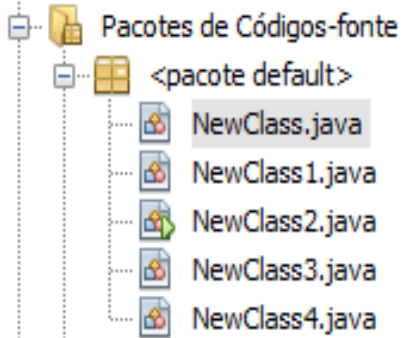
```
1 package Fatec;
2
3 public class Disciplinas_CEETEPs extends Faculdade_CEETEPs {
4     private String nome_disciplina;
5
6     public void setDisciplina(String nome_disciplina){
7         this.nome_disciplina = nome_disciplina;
8     }
9
10    public String getDisciplina(){
11        return nome_disciplina;
12    }
13
14
15 }
16
17
```

Aluno.javaFaculdade_C...Acessa_Facul...Disciplinas_...99+

```
1 package Fatec;
2
3 public class Acessa_Faculdade {
4
5     public static void main(String args[]){
6         Faculdade_CEETEPs ceetps = new Faculdade_CEETEPs();
7         /*a variável de instância tem acesso às classes
8          * de relação de herança de hierarquia superior*/
9         Disciplinas_CEETEPs disciplinas = new Disciplinas_CEETEPs();
10
11         ceetps.setFatec("Fatec Baixada Santista");
12         System.out.println("Fatec \t"+ ceetps.getFatec());
13
14         ceetps.setFaculdade("Administração e Desenvolvimento de Sistemas");
15         System.out.println("Faculdade \t"+ ceetps.getFaculdade());
16
17         disciplinas.setDisciplina("Programação Orientada a Objetos");
18         System.out.println("disciplina \t"+ disciplinas.getDisciplina());
19     }
20 }
21
22
```

Console

<terminated> Acessa_Faculdade [Java Application] C:\Arquivos de Programas\Java\jdk1.8.0_45\bin\javaw.exe (15 de mar de 2018 C
método construtor da superclasse
cidade: Santos
método construtor da superclasse
cidade: Santos
Fatec Fatec Baixada Santista
Faculdade Administração e Desenvolvimento de Sistemas
disciplina Programação Orientada a Objetos



```
/**
 *
 * @author mmario
 */
public class NewClass {
    private String a;

    public NewClass(String a) {
        this.a = a;
        System.out.println("a= " + a);
    }
}
```



```
/**
 *
 * @author mmario
 */
public class NewClass1 extends NewClass {
    public NewClass1() {
        super("aaa");
    }
}
```



```
/**
 *
 * @author mmario
 */
public class NewClass3 extends NewClass1{
    |
}
```



```
/**
 *
 * @author mmario
 */
public class NewClass4 extends NewClass3{
    |
}
```

```
/**
 *
 * @author mmario
 */
public class NewClass2 {
    public static void main(String args []){
        NewClass1 b = new NewClass1();
        NewClass3 c = new NewClass3();
        NewClass4 d = new NewClass4();
    }
}
```

Saída - AnagramGame (run) X

```
run:
a= aaa
a= aaa
a= aaa
CONSTRUÍDO COM SUCESSO (tempo total: 8 segundos)
```