

Classe Correntista FormataData usando Calendar

Criar um projeto -> Aula04

Criar um pacote chamado entity.

Criar uma classe Correntista contendo os atributos (código, nome, saldo, email, dataCadastro).

```
package entity;

import java.text.SimpleDateFormat;
import java.util.Calendar;

public class Correntista {
    private Integer codigo;
    private String nome;
    private Double saldo=0.;
    private String email;
    private Calendar dataCadastro;
```

Na hora de tipar a dataCadastro usar a importação de Calendar. Para importar clicar em CTRL + SHIFT + O.

```
    public Correntista() {
    }

    public Correntista(Integer codigo, String nome, Double
saldo, String email, Calendar dataCadastro) {
        super();
        this.codigo = codigo;
        this.nome = nome;
        this.saldo = saldo;
        this.email = email;
        this.dataCadastro = dataCadastro;
    }

    @Override
    public String toString() {
        return "Correntista [codigo=" + codigo + ", nome=" +
nome + ", saldo=" + saldo + ", email=" + email + ",
```

```
dataCadastro=" + new SimpleDateFormat("dd/MM/yyyy").format  
(dataCadastro.getTime()) + "];  
}
```

No toString formatará como será apresentada a saída utilizando o SimpleDateFormat, colocando na ordem brasileira de data ("dd/MM/yyyy"). A formatação do mês é preciso que seja com letras maiúsculas pois senão será interpretado como minutos (hora).

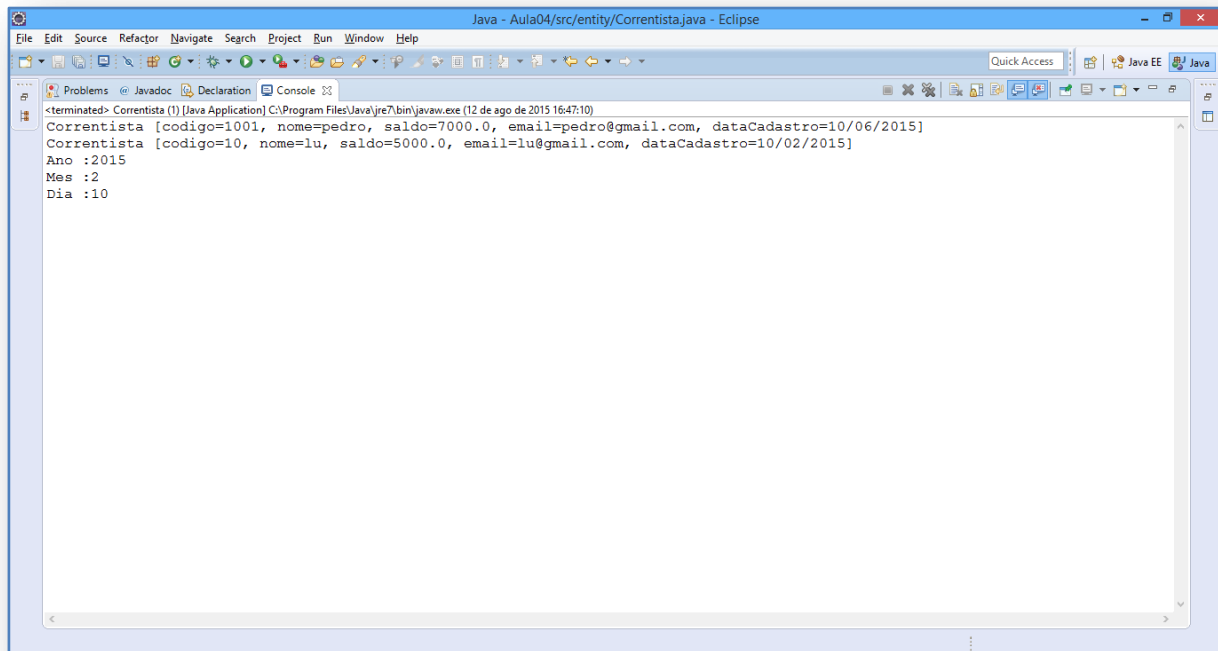
```
public Integer getCodigo() {  
    return codigo;  
}  
public void setCodigo(Integer codigo) {  
    this.codigo = codigo;  
}  
public String getNome() {  
    return nome;  
}  
public void setNome(String nome) {  
    this.nome = nome;  
}  
public Double getSaldo() {  
    return saldo;  
}  
public void setSaldo(Double saldo) {  
    this.saldo = saldo;  
}  
public String getEmail() {  
    return email;  
}  
public void setEmail(String email) {  
    this.email = email;  
}  
public Calendar getDataCadastro() {  
    return dataCadastro;  
}  
public void setDataCadastro(Calendar dataCadastro) {  
    this.dataCadastro = dataCadastro;  
}
```

No teste da main, criar o objeto de Correntista e setar os dados dele. O Calendar começa contando os meses a partir de 0 (zero). Ex: jan=0, fev=1. Chamar a classe Calendar e depois setar a data. Pedir a impressão no console.

```
public static void main(String[] args) {  
  
    Correntista c = new Correntista();  
    c.setCodigo(1001);  
    c.setNome("pedro");  
    c.setEmail("pedro@gmail.com");  
    c.setSaldo(7000.);  
        Calendar cal = Calendar.getInstance();  
        cal.set(2015,05,10);  
    c.setDataCadastro(cal);  
    System.out.println(c);  
}
```

Criar um objeto x para a classe Calendar e depois dar espaço de memória. Setar a nova data. Criar um Correntista novo e passar os dados e pedir as impressões.

```
Calendar x = Calendar.getInstance();  
x.set(2015, 01 ,10);  
Correntista c2 = new  
Correntista(10,"lu",5000., "lu@gmail.com", x );  
System.out.println(c2);  
  
System.out.println("Ano :" +  
c2.getDataCadastro().get(Calendar.YEAR));  
System.out.println("Mes :" + (1 +  
c2.getDataCadastro().get(Calendar.MONTH)) );  
System.out.println("Dia :" +  
c2.getDataCadastro().get(Calendar.DAY_OF_MONTH));  
}  
}
```



```
Java - Aula04/src/entity/Correntista.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java EE Java
Problems Javadoc Declaration Console
<terminated> Correntista (1) [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (12 de ago de 2015 16:47:10)
Correntista [codigo=1001, nome=pedro, saldo=7000.0, email=pedro@gmail.com, dataCadastro=10/06/2015]
Correntista [codigo=10, nome=lu, saldo=5000.0, email=lu@gmail.com, dataCadastro=10/02/2015]
Ano :2015
Mes :2
Dia :10
```