



CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
Campus Contagem
Atividade Prática - Programação em Java

SÉRIE: 3ª CURSO: INFORMÁTICA

Observe as informações que uma determinada empresa precisa armazenar sobre seus funcionários:

| | A | B | C | D | E | F | G |
|----|-------|----------------------|-----------------------------|--------------|----------|-------|------------|
| 1 | Id | nome | email | senha | cargo | turma | setor |
| 2 | 48742 | Ada Trayte | atrayte5@ox.ac.uk | BOGZLTh | Docente | | |
| 3 | 88693 | Addy Pandie | apandie2l@friendfeed.com | 0IMh6C7XNey | Geral | | Estágio |
| 4 | 18246 | Adel Comberbach | acomberbach22@4shared.com | 5wCOKfmLr | Discente | CAM2 | |
| 5 | 94109 | Aigneis Gilleen | agilleanp@economist.com | 98clcV | Discente | CAM2 | |
| 6 | 85306 | Al Gross | agross8@tumblr.com | uP0f7qZ | Geral | | Estágio |
| 7 | 8691 | Alain Lippingwell | alippingwell1t@jalbum.net | zBxn7xh | Geral | | Biblioteca |
| 8 | 87156 | Alanah Reuss | areuss3@sourceforge.net | LQ27EVRhSR4g | Docente | | |
| 9 | 36115 | Allys Barchrameev | abarchrameev@vistaprint.com | soj54iXthjL2 | Discente | ELE2 | |
| 10 | 96888 | Alyss Print | aprint29@clickbank.net | 7Q0VDsmDI | Discente | INF2 | |
| 11 | 99957 | Amara Bax | abax2o@geocities.jp | iwU8XpcRW | Discente | CAM2 | |
| 12 | 19965 | Andreana Leavy | aleavy15@huffingtonpost.com | 7z1kPg | Geral | | Estágio |
| 13 | 98130 | Andreana Mugford | amugford1h@unicef.org | TgEf0oR | Geral | | Biblioteca |
| 14 | 95816 | Audrey Portail | aportailq@fda.gov | E4uxpZjLG7Gg | Docente | | |
| 15 | 54616 | Ban Girardi | bgirardi11@mayoclinic.com | SmkwQQM5rbfr | Geral | | Secretaria |
| 16 | 20824 | Barbara-anne Hansana | bhansana1k@sphinn.com | DswyrxKpFad | Geral | | Secretaria |
| 17 | 44767 | Barbra Bruckenthal | bbruckenthalg@noaa.gov | Jfill7rbfY | Geral | | Estágio |
| 18 | 29195 | Bibbye Goldbourn | bgoldbourn1v@lulu.com | mSMbUfK | Discente | ELE3 | |
| 19 | 95969 | Brande Heaphy | bheaphy2d@moonfruit.com | Vkw1Su4aY0h | Geral | | Estágio |
| 20 | 95187 | Carlota McNevin | cmcnevin4@vk.com | 5ySSLEMF | Geral | | Estágio |
| 21 | 23939 | Caryl Ledes | clesdes12@mapy.cz | 1kza95 | Docente | | |

Em um programa em Ansi C o mais comum seria criar uma estrutura (struct) com os campos acima e então criar variáveis (ou mesmo um array) com o novo tipo de dado criado (structs criam um novo tipo de dado).

```
struct funcionario{  
    int id;  
    char nome[50];  
    char email[30];  
    char senha[30];  
    char cargo[30];  
    char turma[30];  
    char setor[30];  
};
```

Para preenchê-lo criaríamos funções que buscassem esses dados e fizessem a verificação de cada um deles no programa principal (as vezes na própria função main).

Em Programação Orientada a Objetos (POO) temos o conceito de Classes. Classes podem se comportar exatamente como structs em C. Se considerarmos os arrays de caracteres de C como Strings em outras linguagens como Java teremos:

```

class Funcionario{
    int id;
    String nome;
    String email;
    String senha;
    String cargo;
    String turma;
    String setor;
}

```

Porém classes são tipos de dados avançados capazes de armazenar dados e as funções relacionadas a esses dados. Assim uma Classe Funcionario teria tanto os dados quanto as funções de controle desses dados.

Esse seria um exemplo mais completo da classe Funcionário.

```

1  class Funcionario {
2      int id;
3      String nome, email, senha, cargo, turma, setor;
4
5      public Funcionario(int idNovo, String nomeNovo, String senhaNova) {
6          id = idNovo;
7          nome = nomeNovo;
8          senha = senhaNova;
9      }
10
11     public String getEmail() {
12         return email;
13     }
14
15     public void setEmail(String email) {
16         this.email = email;
17     }
18
19     public String mostraFuncionario() {
20         return "Funcionario [id=" + id + ", nome=" + nome + ", email=" + email +
21             ", senha=" + senha + ", cargo=" + cargo + ", turma=" + turma + ", setor=" + setor + "];"
22     }
23 }

```

Abaixo o programa que usa essa classe. Veja que o código vem bem abaixo (veja o número das linhas).

```

26  public class TestaFuncionario {
27
28      public static void main(String[] args) {
29          Funcionario NovoFuncionario = new Funcionario(1, "Jose", "Senha");
30          NovoFuncionario.setEmail("jose@email.com");
31          System.out.println(NovoFuncionario.mostraFuncionario());
32      }
33
34  }

```

Como a classe Pública se chama TestaFuncionario o nome do arquivo é TestaFuncionario.java.

A linha 29 cria um objeto da classe Funcionário. Para criar um segundo seria preciso repetir essa linha mudando o nome da variável (novoFuncionario2).

Atividade de laboratório:

1) Observe as informações que um professor precisa armazenar de seus alunos:

| Matrícula | Nome | Grupo | SubTurma |
|-------------|------------------------------|-------|----------|
| 20202002629 | Ana Beatriz Costa Viana | 4 | 1 |
| 20202001532 | Ana Laura Gonçalves de Paula | 1 | 1 |
| 20202000348 | Artur Borges Corrêa | 7 | 2 |
| 20202005620 | Brian Nicolas Dias Costa | 9 | 2 |
| 20202001013 | Caio César Nascimento Silva | 7 | 2 |

Crie um programa que é capaz de receber os dados de 3 alunos e ao final imprima essas informações.

Para isso:

a) Crie a classe Aluno para armazenar os dados.

b) Crie uma função construtora (como na linha 5 do código de exemplo) que recebe todos os dados e já os insere no objeto no objeto.

c) Senhas devem ser impressas como * no lugar de cada letra (número de asteriscos igual ao tamanho da senha).

d) Crie a função main e teste com ao menos 3 alunos diferentes com os dados fornecidos pelo teclado e imprimindo suas informações ao final do programa.

e) (Desafio) Crie uma função que busca os dados dos estudantes no arquivo de texto alunos.csv fornecido adiciona-os em um array ou coletânea e ao final imprime-os na tela.