

Programa IT Academy – Processo Seletivo – Edição #15

Nome do participante

Eduardo Amengual Garcia

Problema

Dados relativos a bolsas de estudo do governo.

Etapa 2 – Enunciado

Nesta etapa, você vai escrever um programa de computador. Para isso deve ser feita a leitura do arquivo .csv enviado junto com este enunciado. Neste arquivo você encontra dados sobre bolsas de estudo no Brasil. Você deve implementar as seguintes funcionalidades...

Explicação da solução:

Após tentativas iniciais acessar e manipular o arquivo .csv usando JSON e Javascript, percebeu-se que este, não era o método mais eficiente de se desenvolver a solução. Logo posteriormente a isso, optei por trabalhar usando Java juntamente com a IDE Apache NetBeans.

1º Passo: Criar a classe Bolsa que quando estanciada, gera um objeto com atributos equivalentes a uma linha da tabela do arquivo .csv.

2º Passo: Criar a classe Main, para acessar o arquivo .csv , fazer operações e manipular dados.

3° Passo: Acessar o arquivo .csv utilizando uma String com o valor do diretório onde o arquivo reside, juntamente com a classe `BufferedReader` (ajuda a ler textos de uma entrada de dados) e a classe `FileReader` (lê dados de um arquivo). Após isso, criou-se um loop `while` para varrer todas as linhas do arquivo. Dentro do loop, foi criado um vetor para armazenar os dados de uma linha do arquivo .csv, totalizando 11 variáveis que, posteriormente são usados como atributos de uma instância de um objeto da classe `Bolsa`, objeto esse que é adicionado em um `ArrayList` e, todo esse ciclo se repete até que seja feito com todas as linhas do arquivo .csv.

4° Passo: Uma vez com um `ArrayList` contendo todas as linhas do arquivo .csv em forma de objetos, passou-se a ser criado um menu para linha de comando com estrutura `switch case` onde a resposta do usuário seria lida através da instancição de um objeto da classe `Scanner`.

```
--- exec-maven-plugin:3.0.0:exec (default-cli) @ Parte2 ---
Programa IT Academy - Processo Seletivo #15

1 - Consultar Bolsa zero/ano
2 - Codificar nomes
3 - Consultar Media anual
4 - Ranking de valores da bolsa
5 - Finalizar programa

Digite sua opccao:
|
```

Imagem 1: Menu principal

5° Passo: Trabalhar em cima da primeira funcionalidade (Consultar bolsa zero/ano). Foi criado um menu para ler a opção do usuário, bem semelhante ao menu principal (Imagem 1) e posteriormente, a resposta do usuário é lida com o objeto `Scanner`. Uma vez com a resposta do usuário, é feito um `if else`, para verificar se a resposta do usuário condiz com alguma opção do menu. Depois é feito um laço de repetição `for each` para varrer o `ArrayList` dos objetos e é feita uma verificação com os atributos do objeto e por fim, os dados são impressos na tela. Vale ressaltar que para imprimir os dados de 2016, houve algum problema

do qual eu não pude identificar, portanto, imprimi o valor manualmente, visto que o bolsista zero do ano de 2016 é o primeiro objeto do ArrayList.

```
Digite sua opcao:
1

Opção 1 - Consultar bolsa zero/ano

- 2013
- 2014
- 2015
- 2016

Selecione o ano desejado:
```

Imagem 2: Menu funcionalidade 1

```
Selecione o ano desejado:
2013

NOME: JOSE QUERGINALDO BEZERRA
CPF: ***.283.554-**
ENTIDADE DE ENSINO: UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
VALOR BOLSA: 1400.0
```

Imagem 3: Resposta 1 da Funcionalidade 1

```
Selecione o ano desejado:
2014

NOME: CAROLINE ALCANTARA DUARTE
CPF: ***.877.525-**
ENTIDADE DE ENSINO: UNIVERSIDADE DO ESTADO DA BAHIA
VALOR BOLSA: 1100.0
```

Imagem 4: Resposta 2 da Funcionalidade 1

```
Selecione o ano desejado:
2015

NOME: CICERO SARAIVA SOBRINHO
CPF: ***.202.853-**
ENTIDADE DE ENSINO: UNIVERSIDADE DA INTEGRACAO INTERNACIONAL DA LUSOFONIA AFRO-BRASILEIRA
VALOR BOLSA: 1300.0
```

Imagem 5: Resposta 3 da Funcionalidade 1

```
Selecione o ano desejado:
2016

NOME: ALEXANDRE RIBEIRO NETO
CPF: ***.195.647-**
ENTIDADE DE ENSINO: UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
VALOR BOLSA: 765.0
```

Imagem 6: Resposta 4 da Funcionalidade 1

6º Passo: Trabalhar em cima da segunda funcionalidade (Codificar nomes). Infelizmente não consegui finalizar essa funcionalidade. Consegui apenas formular a primeira parte da solução, onde a ideia é de ler a String que o usuário digitar e transformá-la para uma String em caixa alta usando um método do objeto String em Java `.toUpperCase()`. Posteriormente fazer um laço `for each` para varrer todos os objetos do `ArrayList` seguido de um `If else` com parâmetros para comparar o atributo `NM_BOLSISTA` com a String digitada pelo usuário.

7º Passo: Trabalhar em cima da terceira funcionalidade (Consultar média anual). Para essa funcionalidade, criei mais um menu para ler a opção do usuário com o objeto `Scanner`. Após isso, criei um novo `ArrayList` para adicionar todos os objetos que possuam o atributo `AN_REFERENCIA` igual ao do ano digitado pelo usuário. Essa ação é feita de maneira simples com um `for each` no `ArrayList` geral dos objetos, seguido de um `If` para fazer a verificação e então adicionar o objeto no segundo `ArrayList` além de pegar o atributo `AN_REFERENCIA` e adicionar à uma variável total, para ir acumulando os valores. Para finalizar, fiz uma conta

básica onde se divide o valor total pelo tamanho do ArrayList e por fim, imprimir os resultados na tela.

```
Opção 3 - Consultar média anual

- 2013
- 2014
- 2015
- 2016

2013
Selecione o ano desejado:
A média dos valores das bolsas de 2013 é de: 935
```

Imagem 7: Resposta 1 da funcionalidade 3

```
Opção 3 - Consultar média anual

- 2013
- 2014
- 2015
- 2016

2014
Selecione o ano desejado:
A média dos valores das bolsas de 2014 é de: 936
```

Imagem 8: Resposta 2 da funcionalidade 3

```
Opção 3 - Consultar média anual

- 2013
- 2014
- 2015
- 2016

2015
Selecione o ano desejado:
A média dos valores das bolsas de 2015 é de: 938
```

Imagem 9: Resposta 3 da funcionalidade 4

```
Opção 3 - Consultar média anual

- 2013
- 2014
- 2015
- 2016

2016
Selecione o ano desejado:
A média dos valores das bolsas de 2016 é de: 966
```

Imagem 10: Resposta 4 da funcionalidade 3

8º Passo: Trabalhar em cima da funcionalidade 4 (Ranking de valores da bolsa). Para essa funcionalidade optei por fazer um array X de valores Double e preenchê-lo com todos os valores dos atributos VL_BOLSISTA_PAGAMENTO de todos os objetos do ArrayList utilizando um laço de repetição for. Posteriormente criei outro Array y de valores Double porém, com apenas 3 espaços. Então, utilizando um for each no array X é possível verificar se valor de array X é maior ou menor que o valor da primeira posição de array Y, caso seja maior, o valor de array tomará essa posição e o valor de array Y irá para a segunda posição do array Y, e assim seguidamente até que a verificação do

array X seja completada para conseguir os 3 maiores valores de bolsa. Uma vez com os 3 valores mais altos, basta fazer uma verificação no arraylist de objetos utilizando um for each seguindo de um If para verificar os valores e posteriormente imprimi-los. Infelizmente consegui apenas encontrar os 3 valores mais altos, logo, não consegui completar totalmente a funcionalidade 4.

```
Opção 4 - Ranking dos valores de bolsa

Colocação tipo A: os 3 alunos com os valores de bolsa mais altos

Nome:  MARIA ELZA BOCATTI ROSSINI
Valor bolsa: 1300.0

Nome:  JOSE GLEBSON VIEIRA
Valor bolsa: 1400.0

Nome:  LIAMARA SCORTEGAGNA
Valor bolsa: 1500.0
```

Imagem 11: Resposta da funcionalidade 4

Autoavaliação: Creio que poderia ter terminado todas as funcionalidades se não tivesse empenhado tanto tempo tentando realizar o exercício com Javascript. Vale dizer que sem dúvidas, minha maior dificuldade foi conseguir ter “acesso” aos dados do arquivo .csv para então manipular os dados. Porém no que diz respeito a lógica das operações e funcionalidades, fiquei bem satisfeito com meu desempenho.