

Linguagem de Programação 2

Aula 1 - Exercícios

- 1 – Escreva o código fonte da classe Jogador.
 - Incluir três atributos nome, idade e e-mail;
 - Incluir dois métodos construtores;
 - Incluir os métodos de acesso get e set aos atributos;
 - Incluir o método toString.
- 2 – Implementar o método equals da classe Jogador. Considerar todos atributos da classe Jogador;
- 3 – Apresentar a implementação do método hashCode da classe Jogador. Considerar todos atributos da classe Jogador;
- 4 – Descrever a finalidade do código fonte do método equals dos seguintes itens:

```
public boolean equals(Object obj) {  
  
    // item 1 ⇒ Compara se o objeto é igual ao outro, se sim, ele retorna true se não ele vai  
    // para o próximo if.  
    if (this == obj)  
        return true;  
  
    // item 2 ⇒ Compara se o objeto comparado é nulo, se sim, ele retorna false devido ao fato  
    // de um objeto ser nulo e outro não, então não são iguais. Agora, se ele não for nulo, ele vai  
    // para o próximo if  
    if (obj == null)  
        return false;  
  
    // item 3 ⇒ Compara se os dois objetos pertencem a classes diferentes, se sim, retorna  
    // false se não ele cai na próxima condição  
    if (getClass() != obj.getClass())  
        return false;  
    // item 4 ⇒ O objeto é convertido para um objeto do tipo Aluno e será analisado o valor  
    // dentro do objeto  
  
    Aluno other = (Aluno) obj;  
  
    // item 5 ⇒ Se a idade for nula e o objeto tiver a idade com o valor diferente de nulo, vai ser  
    // retornado falso. Se não, se a idade do primeiro objeto será comparada com a idade do  
    // segundo objeto, se forem diferentes, vai retornar falso e acontece o mesmo com o nome do  
    // objeto. Se forem idênticos, será impresso true.  
    if (idade == null) {  
        if (other.idade != null)  
            return false;  
    } else if (!idade.equals(other.idade))
```

```

        return false;
    if (nome == null) {
        if (other.nome != null)
            return false;
    } else if (!nome.equals(other.nome))
        return false;

    return true;
}

```

5 – Escreva um programa Java que receba palavras digitadas pelo usuário através do teclado, armazenar apenas as palavras diferentes em uma coleção, até o usuário digitar a palavra fim. Em seguida, pela segunda vez, solicite novamente ao usuário para digitar várias palavras até digitar a palavra fim, informe quais são as palavras que foram digitadas na segunda vez que encontram-se armazenadas na coleção.

Exemplo:

Digite palavras ou fim para encerrar:

Carro
Pedra
Chave
Bola
Caderno
fim

Novamente digite palavras ou fim para encerrar:

Carro – Localizado
Navio – Não Localizado
Relógio – Não Localizado
Bola – Localizado
fim