

1. Desenvolver um algoritmo que crie um ArrayList para armazenar números inteiros, após isto armazenar 15 números e apresentar os mesmos.

Obs.: não é permitido utilizar nenhuma estrutura de repetição (for, while, do while).

2. Desenvolver um algoritmo que crie um ArrayList para armazenar nomes, deve-se armazenar 9 nomes, apresentar todos os nomes contidos no ArrayList.

Obs.: não é permitido utilizar nenhuma estrutura de repetição (for, while, do while).

3. Desenvolver um algoritmo que crie um ArrayList para armazenar char, deve-se armazenar 15 caracteres, apresentar todos os caracteres armazenados no ArrayList.

Obs.: não é permitido utilizar nenhuma estrutura de repetição (for, while, do while).

4. Desenvolver um algoritmo armazene dez números, apresente-os, remover o número que está na posição 3, apresentar os números armazenados, remover o número que está na posição 4, apresentar os números armazenados e a quantidade de números que restaram no ArrayList.

5. Solicite ao usuário quatro nomes e um outro nome, com esse nome, deve-se verificar se ele está contido o ArrayList ou não e esta informação deve ser apresentada ao usuário.

6. Desenvolver um algoritmo que crie um ArrayList para armazenar números inteiros . Solicite ao usuário quinze números e ao final apresente os números que estão armazenados no ArrayList.

7. Desenvolver um algoritmo e armazene 25 nomes em uma ArrayList e ao final apresentar:

- Todos os nomes no ArrayList;
- Nome com o maior nome;
- Nome com o menor nome;
- Quantidade de pessoas que têm o nome Paulo;
- Quantidade de pessoas que têm o nome Ana;
- Quantidade de pessoas que têm a primeira letra do nome começando com uma vogal.

8. Desenvolver um algoritmo que solicite dez salários, apresentar os seguintes itens:

- Todos os alarmes armazenados;
- A média dos salários;
- A somatória dos salários;
- A soma dos salários que estão armazenados em posições pares;
- A média dos salários que estão armazenados em posições ímpares;
- A quantidade de salários que estão armazenados.

9. Desenvolver um algoritmo que solicite 5 números ao usuário e deve se verificar se o elemento que está contido na primeira posição é igual ao último elemento deste ArrayList.
10. Crie um ArrayList solicitando ao usuário sete números, após isso crie outro ArrayList solicitando ao usuário sete números. Verifique se os elementos que estão armazenados no primeiro são iguais aos elementos que estão armazenados no segundo ArrayList.