

Aula 08 - Array List e Sistemas em Java

Nessa aula vamos começar a desenvolver pequenos sistemas em Java com tudo o que aprendemos até o momento. Entretanto, antes disso, temos que aprender mais um conceito importante, o Array List.

1. ArrayList

O ArrayList é uma Collection no JAVA que além de ser um array, oferece a possibilidade de tipar seus itens e também oferece vários métodos para manipulação do seu conteúdo.

Declaração e instância do ArrayList:

```
ArrayList<Aluno> listaAlunos = new ArrayList<>();
```

Alguns dos métodos/funcionalidades mais importantes do ArrayList, são:

Adicionando itens em um ArrayList

Método: add

```
listaAlunos.add(aluno);
```

Removendo itens de um ArrayList

Método: remove

```
listaAlunos.remove(index);
```

Percorrendo os itens de um ArrayList

```
for(Aluno a: listaAlunos) {  
    System.out.printf("ID: %d - Nome: %s - RA: %s \n", a.getId(),  
a.getNome(), a.getRa());  
}
```

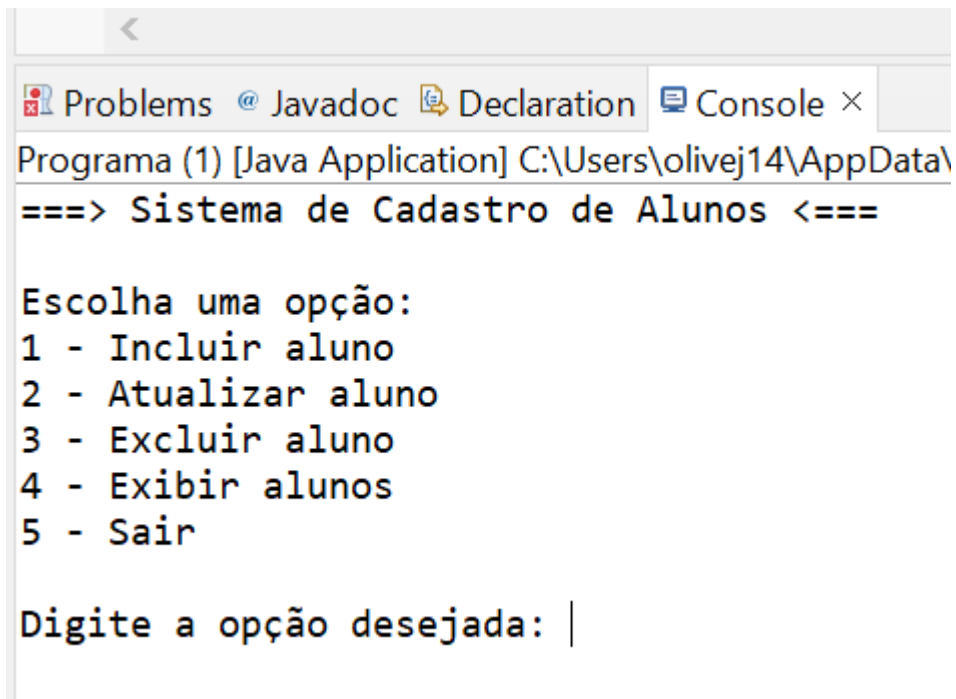
Acessando itens de um ArrayList

```
Aluno aluno = listaAlunos.get(index);
```

2. Sistema em Java

Agora que entendemos como funciona o ArrayList, vamos fazer um sistema em Java que permita cadastrar alunos. O sistema deve ter as funcionalidades de inclusão, alteração, exclusão e seleção, todos dentro de um ArrayList.

Além disso o sistema deve ter o seguinte layout para interação com o usuário:



The screenshot shows an IDE window with a tab labeled 'Console'. The console output is as follows:

```
Programa (1) [Java Application] C:\Users\olivej14\AppData\
===> Sistema de Cadastro de Alunos <===

Escolha uma opção:
1 - Incluir aluno
2 - Atualizar aluno
3 - Excluir aluno
4 - Exibir alunos
5 - Sair

Digite a opção desejada: |
```

Para facilitar o vosso entendimento, segue o código fonte desse sistema que desenvolvi:

Aluno.java

```
package Aula08;

public class Aluno {
    private int id;
    private String nome;
    private String ra;

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }
}
```

```

    public String getRa() {
        return ra;
    }

    public void setRa(String ra) {
        this.ra = ra;
    }

    Aluno() {

    }

    Aluno(String nome, String ra) {
        this.nome = nome;
        this.ra = ra;
    }
}

```

Programa.java

```

package Aula08;

import java.io.IOException;
import java.util.ArrayList;
import java.util.Scanner;

public class Programa {

    public static void main(String[] args) throws IOException{
        Scanner ler = new Scanner(System.in);

        ArrayList<Aluno> listaAlunos = new ArrayList<>();

        int opcao;
        int id;
        String nome;
        String ra;

        do {
            System.out.printf("==> Sistema de Cadastro de Alunos
<==\n\n");

```

```
System.out.printf("Escolha uma opção:\n");
System.out.printf("1 - Incluir aluno \n" +
                  "2 - Atualizar aluno \n" +
                  "3 - Excluir aluno \n" +
                  "4 - Exibir alunos \n" +
                  "5 - Sair\n\n");
System.out.printf("Digite a opção desejada: ");
opcao = ler.nextInt();

if (opcao == 1) {
    Aluno aluno = new Aluno();

    id = listaAlunos.size() + 1;
    aluno.setId(id);

    System.out.printf("\nDigite o nome do aluno: ");
    nome = ler.next();
    aluno.setNome(nome);

    System.out.printf("Digite o ra do aluno: ");
    ra = ler.next();
    aluno.setRa(ra);

    listaAlunos.add(aluno);

    System.out.printf("Aluno incluído com sucesso!");
    System.in.read();
}
else if (opcao == 2) {
    for (Aluno a: listaAlunos) {
        System.out.printf("ID: %d - Nome: %s - RA: %s \n",
a.getId(), a.getNome(), a.getRa());
    }

    System.out.printf("\nDigite o id do aluno que deseja
atualizar: ");
    id = ler.nextInt();

    System.out.printf("Digite o novo nome do aluno: ");
    nome = ler.next();

    System.out.printf("Digite o novo ra do aluno: ");
    ra = ler.next();
```

```

        Aluno aluno = listaAlunos.get(id-1);

        aluno.setNome(nome);
        aluno.setRa(ra);

        System.out.printf("Aluno atualizado com sucesso!");
        System.in.read();
    }
    else if (opcao == 3) {
        for(Aluno a: listaAlunos) {
            System.out.printf("ID: %d - Nome: %s - RA: %s \n",
a.getId(), a.getNome(), a.getRa());
        }

        System.out.printf("\nDigite o id do aluno para
exclusão: ");

        id = ler.nextInt();

        listaAlunos.remove(id-1);

        System.out.printf("Aluno excluído com sucesso!");
        System.in.read();
    }
    else if (opcao == 4) {
        for(Aluno a: listaAlunos) {
            System.out.printf("ID: %d - Nome: %s - RA: %s \n",
a.getId(), a.getNome(), a.getRa());
        }

        System.in.read();
    }
}while(opcao != 5);

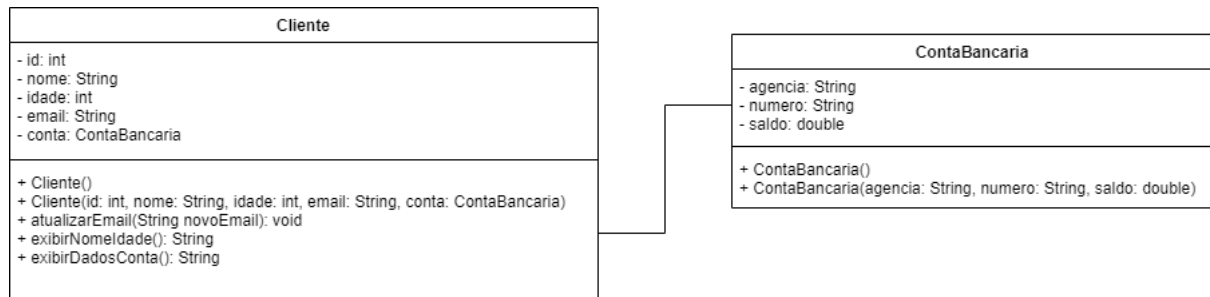
}
}

```

3. Exercícios

Baseado na explicação acima, vamos criar um sistema para os exercícios abaixo. O sistema deve ter as opções de Inclusão, Alteração, Exclusão e Exibição.

31. Crie um sistema com as classes conforme o Diagrama de Classe (UML) abaixo. Crie Getters e Setters para todos os atributos das classes. Crie um programa que utilize essas classes para cadastrar 5 clientes em uma lista de clientes e pergunte para cada cliente se ele tem ou não conta bancária. Caso o cliente tenha, permita ele cadastrar os dados da conta bancária. Ao final, exibir todos os clientes e suas respectivas contas bancárias, se houver.



ContaBancaria.java

```

package Ex31;

public class ContaBancaria {
    private String agencia;
    private String numero;
    private double saldo;

    ContaBancaria() {

    }

    ContaBancaria(String agencia, String numero, double saldo) {
        this.agencia = agencia;
        this.numero = numero;
        this.saldo = saldo;
    }

    public String getAgencia() {
        return agencia;
    }

    public void setAgencia(String agencia) {
        this.agencia = agencia;
    }
}
  
```

```

    public String getNumero() {
        return numero;
    }

    public void setNumero(String numero) {
        this.numero = numero;
    }

    public double getSaldo() {
        return saldo;
    }

    public void setSaldo(double saldo) {
        this.saldo = saldo;
    }
}

```

Cliente.java

```

package Ex31;

public class Cliente {

    private int id;
    private String nome;
    private int idade;
    private String email;
    private ContaBancaria conta;

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }
}

```

```
}

public String getNome() {
    return nome;
}

public void setNome(String nome) {
    this.nome = nome;
}

public int getIdade() {
    return idade;
}

public void setIdade(int idade) {
    this.idade = idade;
}

public String getEmail() {
    return email;
}

public void setEmail(String email) {
    this.email = email;
}

public ContaBancaria getConta() {
    return conta;
}

public void setConta(ContaBancaria conta) {
    this.conta = conta;
}

Cliente(){

}

Cliente(int id,String nome,int idade,String email,ContaBancaria
conta){
    this.id=id;
    this.nome=nome;
    this.idade=idade;
    this.email=email;
    this.conta=conta;
}

public void atualizarEmail(String novoEmail){
    this.email=novoEmail;
}
```



```

    public String exibirCliente() {
        return("ID: "+this.id+ " - Nome: " + this.nome + " - Idade: " +
this.idade + " - Email: " + this.email);
    }

    public String exibirDadosConta() {
        return("Agencia: "+this.conta.getAgencia()+" - Numero:
"+this.conta.getNumero()+" - Saldo: "+this.conta.getSaldo());
    }
}

```

Programa.java

```

package Ex31;

import java.io.IOException;
import java.util.ArrayList;
import java.util.Scanner;

public class Programa {

    public static void main(String[] args) throws IOException{
        Scanner ler = new Scanner(System.in);

        ArrayList<Cliente> listaClientes = new ArrayList<>();

        int opcao;
        int id;
        Cliente cli;
        int indexCliente;
        int id_cliente = 0;
        String possuiConta;

        do {
            System.out.printf("==> Sistema de Cadastro de Clientes
<==\n\n");
            System.out.printf("Escolha uma opção:\n");
            System.out.printf("1 - Incluir \n" +
                                "2 - Atualizar \n" +
                                "3 - Excluir \n" +
                                "4 - Exibir \n" +
                                "5 - Sair\n\n");

            System.out.printf("Digite a opção desejada: ");

```

```
opcao = ler.nextInt();

if (opcao == 1) {
    Cliente cliente = new Cliente();

    id_cliente++;
    cliente.setId(id_cliente);

    System.out.print("Digite o seu nome: ");
    cliente.setNome(ler.next());

    System.out.print("Digite a seu idade: ");
    cliente.setIdade(ler.nextInt());

    System.out.print("Digite o seu e-mail: ");
    cliente.setEmail(ler.next());

    System.out.print("Possui conta bancária? S/N");
    possuiConta = ler.next().toUpperCase();

    if(possuiConta.equals("S")) {
        ContaBancaria conta = new ContaBancaria();

        System.out.print("Digite a agencia: ");
        conta.setAgencia(ler.next());

        System.out.print("Digite o numero: ");
        conta.setNumero(ler.next());

        conta.setSaldo(0);

        cliente.setConta(conta);
    }
    else {
        cliente.setConta(null);
    }

    listaClientes.add(cliente);

    System.out.printf("Cliente incluído com sucesso!");
    System.in.read();
}
else if(opcao == 2) {
```

```

        for(Cliente c: listaClientes) {
            System.out.println(c.exibirCliente());
        }

        System.out.print("Digite o ID do cliente que você deseja
atualizar: ");
        id = ler.nextInt();

        indexCliente = -1;
        for(Cliente c: listaClientes) {
            if (c.getId() == id) {
                indexCliente = listaClientes.indexOf(c);
                break;
            }
        }

        if (indexCliente != -1) {
            cli = listaClientes.get(indexCliente);

            System.out.print("Digite o seu novo nome: ");
            cli.setNome(ler.next());

            System.out.print("Digite a sua nova idade: ");
            cli.setIdade(ler.nextInt());

            System.out.print("Digite o seu novo e-mail: ");
            cli.setEmail(ler.next());

            System.out.printf("Cliente atualizado com
sucesso!");
        }
        else {
            System.out.printf("Cliente não encontrado!");
        }
        System.in.read();
    }
    else if(opcao == 3) {
        for(Cliente c: listaClientes) {
            System.out.println(c.exibirCliente());
        }

        System.out.print("Digite o ID do cliente que você deseja
excluir: ");

```

```

        id = ler.nextInt();

        indexCliente = -1;
        for(Cliente c: listaClientes) {
            if (c.getId() == id) {
                indexCliente = listaClientes.indexOf(c);
                break;
            }
        }

        if (indexCliente != -1) {
            listaClientes.remove(indexCliente);

            System.out.printf("Cliente excluído com sucesso!");
        }
        else {
            System.out.printf("Cliente não encontrado!");
        }
        System.in.read();
    }
    else if(opcao == 4) {
        for(Cliente c: listaClientes) {
            System.out.println(c.exibirCliente());

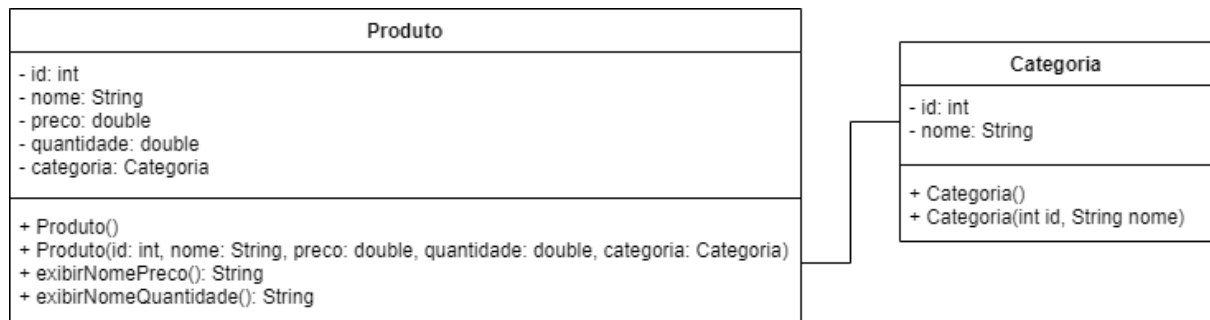
            if (c.getConta() != null)
                System.out.println(c.exibirDadosConta());
        }

        System.in.read();
    }

}while( (opcao >= 1) && (opcao <= 4) );
}
}

```

32. Crie um sistema com as classes conforme o Diagrama de Classe (UML) abaixo. Crie Getters e Setters para todos os atributos das classes. Crie um programa que utilize essas classes para cadastrar 5 produtos em uma lista de produtos e pergunte para cada produto se ele tem ou não uma categoria. Caso o produto tenha, permita ele cadastrar os dados da categoria. Ao final, exibir todos os produtos e suas respectivas categorias, se houver.



Categoria.java

```
package Ex32;

public class Categoria {
    private int id;
    private String nome;

    Categoria() {
    }

    Categoria(int id, String nome) {
        this.id = id;
        this.nome = nome;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }
}
```

Produto.java

```
package Ex32;
```

```
public class Produto {  
    private int id;  
    private String nome;  
    private double preco;  
    private double quantidade;  
    private Categoria categoria;  
  
    Produto(){  
    }  
  
    Produto(int id, String nome, double preco, double quantidade,  
Categoria categoria){  
        this.id = id;  
        this.nome = nome;  
        this.preco = preco;  
        this.quantidade = quantidade;  
        this.categoria = categoria;  
    }  
  
    public int getId() {  
        return id;  
    }  
  
    public void setId(int id) {  
        this.id = id;  
    }  
  
    public String getNome() {  
        return nome;  
    }  
  
    public void setNome(String nome) {  
        this.nome = nome;  
    }  
  
    public double getPreco() {  
        return preco;  
    }  
  
    public void setPreco(double preco) {  
        this.preco = preco;  
    }  
}
```

```

    public double getQuantidade() {
        return quantidade;
    }

    public void setQuantidade(double quantidade) {
        this.quantidade = quantidade;
    }

    public Categoria getCategoria() {
        return categoria;
    }

    public void setCategoria(Categoria categoria) {
        this.categoria = categoria;
    }

    public String exibirInfoCategoria() {
        if (this.categoria != null)
            return "Categoria: " + this.categoria.getNome();
        else
            return "Produto sem categoria";
    }

    public String exibirInfoProduto() {
        return "Id: " + this.id + " | Produto: " + this.nome + " |  

Quantidade: " + this.quantidade + " | Preço: " + this.preco;
    }
}

```

Programa.java

```

package Ex32;

import java.util.ArrayList;
import java.util.Scanner;

public class Programa {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        ArrayList<Produto> listaProdutos = new ArrayList<>();

        int opcao;
    }
}

```

```

    int id;
    Produto p;
    double quantidade;
    double precoProduto;
    int indexProduto;
    int id_produto = 0;
    String possuiCategoria;
    String novaCategoria;

    do {
        System.out.println("==> Sistema de Cadastro de Produtos
<==\n\n");
        System.out.println("Escolha uma opção:\n");
        System.out.printf("1 - Incluir \n" +
            "2 - Atualizar \n" +
            "3 - Excluir \n" +
            "4 - Exibir \n" +
            "5 - Sair\n\n");

        System.out.printf("Digite a opção desejada: ");
        opcao = ler.nextInt();

        if (opcao == 1) {
            Produto produto = new Produto();

            id_produto++;
            produto.setId(id_produto);

            System.out.println("Digite o nome do produto: ");
            produto.setNome(ler.next());

            System.out.println("Digite o preço do produto: ");
            produto.setPreco(ler.nextDouble());

            System.out.println("Digite a quantidade do produto: ");
            produto.setQuantidade(ler.nextDouble());

            System.out.println("O produto possui categoria? S/N:
");
            possuiCategoria = ler.next().toUpperCase();

            if (possuiCategoria.equals("S")) {
                Categoria categoria = new Categoria();

```



```

        System.out.println("Digite a categoria do produto:
");

        categoria.setNome(ler.next());

        categoria.setId(id_produto);

        produto.setCategoria(categoria);
    }

    listaProdutos.add(produto);

    System.out.printf("Produto incluído com sucesso!");
    ler.nextLine();

} else if (opcao == 2) {
    for (Produto prod : listaProdutos) {
        System.out.println(prod.exibirInfoProduto());
    }

    System.out.print("Digite o ID do produto que você
deseja atualizar: ");
    id = ler.nextInt();

    indexProduto = -1;
    for (int i = 0; i < listaProdutos.size(); i++) {
        if (listaProdutos.get(i).getId() == id) {
            indexProduto = i;
            break;
        }
    }

    if (indexProduto != -1) {
        p = listaProdutos.get(indexProduto);

        System.out.println("Digite o novo nome do produto:
");

        p.setNome(ler.next());

        System.out.println("Digite o novo preço: ");
        p.setPreco(ler.nextDouble());

        System.out.println("Digite a nova categoria: ");

```

```

        novaCategoria = ler.next();
        if (!novaCategoria.isEmpty()) {
            Categoria categoria = new Categoria();
            categoria.setId(id);
            categoria.setNome(novaCategoria);
            p.setCategoria(categoria);
        }

        System.out.println("Produto atualizado com
sucesso!");

        ler.nextLine();
    } else {
        System.out.printf("Produto não encontrado!");
        ler.nextLine();
    }

} else if (opcao == 3) {
    for (Produto prod : listaProdutos) {
        System.out.println(prod.exibirInfoProduto());
    }

    System.out.print("Digite o ID do produto que você
deseja excluir: ");
    id = ler.nextInt();

    indexProduto = -1;
    for (int i = 0; i < listaProdutos.size(); i++) {
        if (listaProdutos.get(i).getId() == id) {
            indexProduto = i;
            break;
        }
    }

    if (indexProduto != -1) {
        listaProdutos.remove(indexProduto);

        System.out.printf("Produto excluído com sucesso!");
        ler.nextLine();
    } else {
        System.out.printf("Produto não encontrado!");
        ler.nextLine();
    }
}

```

```
        } else if (opcao == 4) {  
            for (Produto prod : listaProdutos) {  
                System.out.println(prod.exibirInfoProduto());  
                System.out.println(prod.exibirInfoCategoria());  
            }  
        }  
  
    } while ((opcao >= 1) && (opcao <=4));  
  
}  
}
```