Arraylist – 8-11-23

import java.util.ArrayList;

import java.util.Scanner; //MEGLIO USARE INPUTSTREAMREADER

public class Main {

public static void main(String[] args) {

ArrayList<Nodo> listaNodi = new ArrayList<Nodo>();

Scanner scanner = new Scanner(System.in);

while (true) {

System.out.println("Inserisci una parola (o 'exit' per uscire):");

String input = scanner.nextLine();

if (input.equals("exit")) {

break;

}

Nodo nuovoNodo = new Nodo(input);

if (listaNodi.isEmpty()) {

listaNodi.add(nuovoNodo); // Aggiungi in testa se la lista è vuota

} else {

System.out.println("Dove desideri aggiungere la parola?");

System.out.println("1. In testa");

System.out.println("2. In coda");

System.out.println("3. All'interno della lista");

int scelta = scanner.nextInt();

scanner.nextLine(); // Consuma il newline

switch (scelta) {

case 1:

listaNodi.add(0, nuovoNodo); // Aggiungi in testa

break;

case 2:

listaNodi.add(nuovoNodo); // Aggiungi in coda

break;

case 3:

System.out.println("Inserisci l'indice in cui desideri aggiungere la parola:");

int indice = scanner.nextInt();

scanner.nextLine(); // Consuma il newline

if (indice >= 0 && indice < listaNodi.size()) {

listaNodi.add(indice, nuovoNodo); // Aggiungi all'interno della lista

} else {

System.out.println("Indice non valido. La parola verrà aggiunta in coda.");

listaNodi.add(nuovoNodo); // Aggiungi in coda se l'indice non è valido

}

break;

default:

System.out.println("Scelta non valida. La parola verrà aggiunta in coda.");

listaNodi.add(nuovoNodo); // Aggiungi in coda se la scelta non è valida

break;

}

}

System.out.println("Lista di parole:");

for (Nodo nodo : listaNodi) {

System.out.println(nodo.getParola());

}

}

scanner.close();

}

}

class Nodo {

private String parola;

public Nodo(String parola) {

this.parola = parola;

}

public String getParola() {

return parola;

}

}

APPROFONDIMENTO SULLE LISTE:

L'esercizio appena svolto spiega in maniera semplicistica attraverso l'utilizzo della classe nodo come funziona nella realtà l'interfaccia list (O più precisamente LINKEDLIST). Essa è presente all'interno di Java e contiene la classe arraylist che abbiamo analizzato in classe.

per un maggiore approfondimento a casa si lascia l'alunno consultare in autonomia il seguente link

<https://pages.di.unipi.it/romani/DIDATTICA/LSD/LSD/Implementazione/ListeConcatenate/main.html>

Crea un esercizio in Java in cui rappresentare una classe di nome voce al cui interno si trova in private nome e numero di telefono. Crea poi nella classe main il menu in cui l'utente seleziona 1 per aggiungere la voce, 2 per eliminare la voce, 3 per visualizzare la rubrica e 4 per uscire. Crea quindi la rubrica telefonica attraverso una lista di voce in cui inserire i metodi aggiungi voce, elimina voce o visualizza voce.

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.ArrayList;

public class RubricaTelefonica {

public static void main(String[] args) throws IOException {

ArrayList<Voce> rubrica = new ArrayList<>();

BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));

while (true) {

System.out.println("Menu:");

System.out.println("1. Aggiungi Voce");

System.out.println("2. Elimina Voce");

System.out.println("3. Visualizza Rubrica");

System.out.println("4. Esci");

System.out.print("Seleziona un'opzione: ");

String scelta = reader.readLine();

switch (scelta) {

case "1":

System.out.print("Inserisci il nome: ");

String nome = reader.readLine();

System.out.print("Inserisci il numero di telefono: ");

String numero = reader.readLine();

Voce nuovaVoce = new Voce(nome, numero);

rubrica.add(nuovaVoce);

System.out.println("Voce aggiunta alla rubrica.");

break;

case "2":

System.out.print("Inserisci il nome da eliminare: ");

String nomeDaEliminare = reader.readLine();

boolean rimossa = false;

for (int i = 0; i < rubrica.size(); i++) {

if (rubrica.get(i).getNome().equals(nomeDaEliminare)) {

rubrica.remove(i);

rimossa = true;

System.out.println("Voce rimossa dalla rubrica.");

break;

}

}

if (!rimossa) {

System.out.println("Nessuna voce con quel nome trovata nella rubrica.");

}

break;

case "3":

System.out.println("Rubrica Telefonica:");

for (Voce voce : rubrica) {

System.out.println(voce);

}

break;

case "4":

System.out.println("Uscita dalla rubrica telefonica.");

reader.close();

System.exit(0);

default:

System.out.println("Opzione non valida. Riprova.");

}

}

}

}

class Voce {

private String nome;

private String numero;

public Voce(String nome, String numero) {

this.nome = nome;

this.numero = numero;

}

public String getNome() {

return nome;

}

@Override

public String toString() {

return "Nome: " + nome + ", Numero: " + numero;

}

}