package stq;

import java.util.\*;

public class Stack\_Queues\_Trees {

private static LinkedList <String> stack = new LinkedList<>();

private static LinkedList <String> queue = new LinkedList<>();

private static Scanner input = new Scanner(System.in);

public static void Stack(){

boolean match = true;

while(match){

println("\tSTACK \n\n");

println("Cabinet \n");

println("[1] Add");

println("[2] Delete");

println("[3] Display");

println("[4] Back to main menu");

print("Please enter: ");

String enter = input.nextLine();

switch(ErrorHandler(enter)){

case 1:{

while(true){

print("Enter item to add: ");

String add = input.nextLine();

boolean item = true;

for(String stack : stack){

if(stack.equalsIgnoreCase(add)){

println("Item already exist.");

item = false;

break;

}

}

if(item){

stack.addLast(add);

println(add + " added to the cabinet successfully.");

break;

}

}

break;

}

case 2:{

if(!stack.isEmpty()){

stack.removeLast();

println("Item deleted successfully.");

}else{

println("No item in the cabinet.");

}

break;

}

case 3:{

if(!stack.isEmpty()){

println("Items in the cabinet. \n");

println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

for(String stack : stack){

println(stack);

println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

}else{

println("No item in the cabinet.");

}

break;

}

case 4:{

match = false;

break;

}default:

}

}

}

public static void Queue(){

boolean match = true;

while(match){

println("\tQUEUE \n\n");

println("Refregirator \n");

println("[1] Add");

println("[2] Delete");

println("[3] Display");

println("[4] Back to main menu");

print("Please enter: ");

String enter = input.nextLine();

switch(ErrorHandler(enter)){

case 1:{

while(true){

print("Enter item to add: ");

String add = input.nextLine();

boolean item = true;

for(String queue : queue){

if(queue.equalsIgnoreCase(add)){

println("Item already exist.");

item = false;

break;

}

}

if(item){

queue.addLast(add);

println(add + " added to the refregirator successfully.");

break;

}

}

break;

}

case 2:{

if(!queue.isEmpty()){

queue.removeFirst();

println("Item deleted successfully.");

}else{

println("No item in the refregirator.");

}

break;

}

case 3:{

if(!queue.isEmpty()){

println("Items in the refregirator. \n");

println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

for(String queue : queue){

println(queue);

println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

}else{

println("No item in the refregirator.");

}

break;

}

case 4:{

match = false;

break;

}default:

}

}

}

public static void Trees(){

boolean match = true;

while(match){

println("\tTREES \n\n");

println("[1] In-Order");

println("[2] Pre-Order");

println("[3] Post-Order");

println("[4] Back to main menu");

print("Please enter: ");

String enter = input.nextLine();

switch(ErrorHandler(enter)){

case 1:{

println("In-Order \n");

println("[CCC, BB, HH, DD, AA, FF, GG, EE, CC]");

break;

}

case 2:{

println("Pre-Order \n");

println("[AA, BB, CCC, HH, DD, CC, EE, FF, GG]");

break;

}

case 3:{

println("Post-Order \n");

println("HH, CCC, DD, BB, FF, GG, EE, CC, AA]");

break;

}

case 4:{

match = false;

break;

}default:

}

}

}

public static void main(String[] args){

while(true){

println("\tMENU\n");

println("[1] Stack ");

println("[2] Queues ");

println("[3] Trees ");

println("[4] Exit ");

print("Please enter: ");

String enter = input.nextLine();

switch(ErrorHandler(enter)){

case 1:{

Stack();

break;

}

case 2:{

Queue();

break;

}

case 3:{

Trees();

break;

}

case 4:{

println("Program finished...");

System.exit(0);

break;

}default:

}

}

}

public static int ErrorHandler(String var){

int val = 0;

try{

val = Integer.parseInt(var);

}catch(NumberFormatException intException){

println("Invalid input.");

}

return val;

}

public static void print(String var){

System.out.print(var);

}

public static void print(int var){

System.out.print(var);

}

public static void println(String var){

System.out.println(var);

}

public static void println(int var){

System.out.println(var);

}

}