

PMR: This assignment was just an extension of the previous assignment was so it was really easy. The previous assignment was not so easy so I was a little timid of changing the code but it ended up being easy with the help of the module and internet.

/\*\*

\* Candidate interface.

\*

\* @author Anika Jallipalli

\* @version 4/5/2020

\*/

public class Candidate4

{

private String name;

private int numVotes;

public Candidate4(String name, int numVotes){

this.name = name;

this.numVotes = numVotes;

}

public String getName(){

return name;

}

public int getVotes(){

return numVotes;

}

public String toString(){

return getName() + " received " + getVotes() + " votes.";

}

public void replaceName(String n){ //overloaded to set your own name to replace

name = n;

}

public void replaceVotes(int v){ //overloaded, change it to a certain number)

numVotes = v;

}

}

/\*\*

\* Write a description of class TestCandidate5 here.

\*

\* @author Anika Jallipalli

\* @version 4/5/2020

\*/

public class TestCandidate7

{

public static void main(String[] args)

{

int sum = 0;

int counter = 0;

Candidate4 john = new Candidate4("John Smith", 5000);

Candidate4 mary = new Candidate4("Mary Miller", 4000);

Candidate4 michael = new Candidate4("Michael Duffy", 6000);

Candidate4 tim = new Candidate4("Tim Robinson", 2500);

Candidate4 joe = new Candidate4("Joe Ashtony", 1800);

Candidate4 mickey = new Candidate4("Mickey Jones", 3000);

Candidate4 rebecca = new Candidate4("Rebecca Morgan", 2000);

Candidate4 kathleen = new Candidate4("Kathleen Turner", 8000);

Candidate4 tory = new Candidate4("Tory Parker", 500);

Candidate4 ashton = new Candidate4("Ashton Davis", 10000);

Candidate4[] candidates = new Candidate4[] {john, mary, michael, tim, joe, mickey, rebecca, kathleen, tory, ashton};

//original votes START

System.out.println("Original Results:");

System.out.println();

for(int i = 0; i < candidates.length; i++)

{

sum += candidates[i].getVotes();

}

System.out.println("Candidate Votes Received % of Total Votes");

for(int i = 0; i < candidates.length; i++)

{

System.out.printf("%15s %5d %2f\n",

candidates[i].getName(), candidates[i].getVotes(), ((double)candidates[i].getVotes() / sum) \* 100);

}

System.out.println();

System.out.println("Total number of votes in election: " + sum);

//original votes END

System.out.println();

int pos = 6;

sum -= candidates[pos].getVotes();

deleteByLoc(candidates, pos);

//new votes START

System.out.println("Deleted position 6:");

System.out.println();

System.out.println("Candidate Votes Received % of Total Votes");

for(int i = 0; i < candidates.length; i++)

{

if(candidates[i]!=null)

{

System.out.printf("%15s %5d %2f\n",

candidates[i].getName(), candidates[i].getVotes(), ((double)candidates[i].getVotes() / sum) \* 100);

}

else

{

i++;

}

}

System.out.println();

System.out.println("Total number of votes in election: " + sum);

//new votes END

System.out.println();

String name = "Kathleen Turner";

int location = 0;

for(int i = 0; i < candidates.length; i++)

{

if ((candidates[i] != null) && (candidates[i].getName().equals(name)))

{

location = i;

break;

} else if (candidates[i] == null)

{

location = -1;

break;

}

}

sum -= candidates[location].getVotes();

deleteByName(candidates, name);

//new new votes

System.out.println("Deleted Kathleen Turner:");

System.out.println();

System.out.println("Candidate Votes Received % of Total Votes");

for(int i = 0; i < candidates.length; i++)

{

if(candidates[i]!=null)

{

System.out.printf("%15s %5d %2f\n",

candidates[i].getName(), candidates[i].getVotes(), ((double)candidates[i].getVotes() / sum) \* 100);

}

else

{

i++;

}

}

System.out.println();

System.out.println("Total number of votes in election: " + sum);

}

public static void insertPosition(Candidate4[] a, int pos, String name, int votes)

{

Candidate4[] candidates = a;

int position = pos;

String n = name;

int v = votes;

for(int i=candidates.length-1;i>position;i--)

{

candidates[i] = candidates[i-1];

}

Candidate4 newperson = new Candidate4(n, votes);

candidates[position] = newperson;

}

public static void insertCandidate(Candidate4[] a, String name, String newperson, int votes)

{

Candidate4[] candidates = a;

String newb = newperson;

String n = name;

int v = votes;

int pos = 0;

Candidate4 donald = new Candidate4(newb, votes);

for(int i=0;i<candidates.length;i++)

{

if(candidates[i].getName().equals(n))

{

pos = i;

}

}

for(int i=candidates.length-1;i>pos;i--)

{

candidates[i] = candidates[i-1];

}

candidates[pos]= donald;

}

public static void deleteByLoc(Candidate4[] a, int location)

{

if ((location > 0) && (location < a.length))

{

//move items up in the array -

for(int i = location; i < a.length -1; i++)

{

a[i] = a[i + 1];

}

a[a.length-1] = null;

}

}

public static void deleteByName(Candidate4[] a, String find)

{

int location = 0;

int i;

// find location of item you want to delete

for(i = 0; i < a.length; i++)

{

if ((a[i] != null) && (a[i].getName().equals(find)))

{

location = i;

break;

} else if (a[i] == null)

{

location = -1;

break;

}

}

if ((i != a.length) && (location >= 0))

{ //move items up in the array

for(i = location; i < a.length -1; i++)

{

a[i] = a[i + 1];

}

a[a.length-1] = null;

}

}

}

/\*\*

\* arraylist verison

\*

\* @author Anika Jallipalli

\* @version 4/5/2020

..

\*/

import java.util.\*;

public class TestCandidate8

{

public static void main(String[] args)

{

int sum = 0, counter = 0;

ArrayList<Candidate4> c = new ArrayList<Candidate4>();

Candidate4 john = new Candidate4("John Smith", 5000);

c.add(john);

Candidate4 mary = new Candidate4("Mary Miller", 4000);

c.add(mary);

Candidate4 michael = new Candidate4("Michael Duffy", 6000);

c.add(michael);

Candidate4 tim = new Candidate4("Tim Robinson", 2500);

c.add(tim);

Candidate4 joe = new Candidate4("Joe Ashtony", 1800);

c.add(joe);

Candidate4 mickey = new Candidate4("Mickey Jones", 3000);

c.add(mickey);

Candidate4 rebecca = new Candidate4("Rebecca Morgan", 2000);

c.add(rebecca);

Candidate4 kathleen = new Candidate4("Kathleen Turner", 8000);

c.add(kathleen);

Candidate4 tory = new Candidate4("Tory Parker", 500);

c.add(tory);

Candidate4 ashton = new Candidate4("Ashton Davis", 10000);

c.add(ashton);

System.out.println("Original Results:");

for(Candidate4 t : c)

{

sum += t.getVotes();

counter++;

}

System.out.println();

System.out.println("Candidate Votes Received % of Total Votes");

for(Candidate4 t : c)

{

System.out.printf("%15s %5d %2f\n",

t.getName(), t.getVotes(), ((double)t.getVotes() / sum) \* 100);

}

System.out.println();

System.out.println("Total number of votes in election: " + sum);

System.out.println();

System.out.println("Deleted location 6:");

System.out.println();

deleteByLoc(c, 6);

sum = 0;

for(Candidate4 t : c)

{

sum += t.getVotes();

counter++;

}

System.out.println("Candidate Votes Received % of Total Votes");

for(Candidate4 t : c)

{

System.out.printf("%15s %5d %2f\n",

t.getName(), t.getVotes(), ((double)t.getVotes() / sum) \* 100);

}

System.out.println();

System.out.println("Total number of votes in election: " + sum);

System.out.println();

System.out.println("Deleted Kathleen Turner:");

System.out.println();

deleteByName(c, "Kathleen Turner");

sum = 0;

for(Candidate4 t : c)

{

sum += t.getVotes();

counter++;

}

System.out.println("Candidate Votes Received % of Total Votes");

for(Candidate4 t : c)

{

System.out.printf("%15s %5d %2f\n",

t.getName(), t.getVotes(), ((double)t.getVotes() / sum) \* 100);

}

System.out.println();

System.out.println("Total number of votes in election: " + sum);

}

public static void insertPosition(List<Candidate4> list, int location, String name, int votes)

{

// insert item into ArrayList

list.add(location, new Candidate4(name, votes));

}

public static void insertCandidate(List<Candidate4> list, String find, String name, int votes)

{

int location = 0;

// find location of item you want to insert before

for(int index = 0; index < list.size(); index++)

{

if (list.get(index).getName().equals(find))

{

location = index;

}

}

// insert item into ArrayList

list.add(location, new Candidate4(name, votes));

}

public static void deleteByLoc(List<Candidate4> list, int location)

{

// delete item from ArrayList

list.remove(location);

}

public static void deleteByName(List<Candidate4> list, String find)

{

int location = 0;

int index;

// find location of item you want to delete

for(index = 0; index < list.size(); index++)

{

if (list.get(index).getName().equals(find))

{

location = index;

break;

}

}

// delet item from ArrayList

if (index != list.size())

{

list.remove(location);

}

}

}