using System;

using System.Collections.Generic;

using System.Linq;

enum ObjektaTips { Katalogs, Fails }

class Objekts

{

public string Nosaukums { get; set; }

public Objekts Augstākais { get; set; }

public List<Objekts> Apakšobjekti { get; set; } = new List<Objekts>();

public ObjektaTips Veids { get; set; }

public Objekts(string nos, ObjektaTips veids, Objekts augstakais = null)

{

Nosaukums = nos;

Veids = veids;

Augstākais = augstakais;

}

public void RādītSaturs()

{

foreach (var elem in Apakšobjekti)

{

if (elem.Veids == ObjektaTips.Fails)

Console.WriteLine("- " + elem.Nosaukums);

else

Console.WriteLine("[" + elem.Nosaukums + "]");

}

}

public void PievienotObjektu(string nos, ObjektaTips veids)

{

if (Apakšobjekti.Any(x => x.Nosaukums.Equals(nos, StringComparison.OrdinalIgnoreCase)))

{

Console.WriteLine("Šāds nosaukums jau pastāv.");

return;

}

Apakšobjekti.Add(new Objekts(nos, veids, this));

}

public void DzēstObjektu(string nos)

{

var mērķis = Apakšobjekti.FirstOrDefault(x => x.Nosaukums.Equals(nos, StringComparison.OrdinalIgnoreCase));

if (mērķis == null)

{

Console.WriteLine("Nav atrasts objekts ar šādu nosaukumu.");

return;

}

if (mērķis.Veids == ObjektaTips.Katalogs && mērķis.Apakašobjekti.Any())

{

Console.Write($"Mape '{nos}' nav tukša. Turpināt dzēšanu? (j/n): ");

if (Console.ReadLine().ToLower() != "j")

{

Console.WriteLine("Dzēšana pārtraukta.");

return;

}

}

Apakšobjekti.Remove(mērķis);

Console.WriteLine($"Objekts '{nos}' ir izņemts.");

}

public Objekts AtrastApakšu(string nos)

{

return Apakšobjekti.FirstOrDefault(x => x.Nosaukums.Equals(nos, StringComparison.OrdinalIgnoreCase) && x.Veids == ObjektaTips.Katalogs);

}

public void AtvērtFailu(string nos)

{

var datne = Apakšobjekti.FirstOrDefault(x => x.Nosaukums.Equals(nos, StringComparison.OrdinalIgnoreCase) && x.Veids == ObjektaTips.Fails);

if (datne != null)

Console.WriteLine("~ " + datne.Nosaukums + " ~");

else

Console.WriteLine("Fails nav atrodams.");

}

public string CeļaAdrese()

{

return Augstākais == null ? Nosaukums + ":/" : Augstākais.CeļaAdrese() + Nosaukums + "/";

}

}

class Programma

{

public static void Main()

{

var sakne = new Objekts("sakne", ObjektaTips.Katalogs);

var pozicija = sakne;

while (true)

{

Console.Write(pozicija.CeļaAdrese() + "> ");

var ievade = Console.ReadLine()?.Trim();

if (string.IsNullOrEmpty(ievade)) continue;

var daļas = ievade.Split(' ', 2);

var komanda = daļas[0].ToLower();

var args = daļas.Length > 1 ? daļas[1] : "";

try

{

switch (komanda)

{

case "newdir":

if (!string.IsNullOrWhiteSpace(args))

pozicija.PievienotObjektu(args, ObjektaTips.Katalogs);

break;

case "newfile":

if (!string.IsNullOrWhiteSpace(args))

pozicija.PievienotObjektu(args, ObjektaTips.Fails);

break;

case "remove":

if (!string.IsNullOrWhiteSpace(args))

pozicija.DzēstObjektu(args);

break;

case "list":

pozicija.RādītSaturs();

break;

case "goto":

if (args == "..")

{

if (pozicija.Augstākais != null)

pozicija = pozicija.Augstākais;

}

else

{

var celu = pozicija.AtrastApakšu(args);

if (celu != null)

pozicija = celu;

else

Console.WriteLine("Mape nav atrasta.");

}

break;

case "open":

if (!string.IsNullOrWhiteSpace(args))

pozicija.AtvērtFailu(args);

break;

case "exit":

return;

default:

Console.WriteLine("Nederīga komanda.");

break;

}

}

catch

{

Console.WriteLine("Kļūda.");

}

}

}

}