

Dėstytojas

Edvinas Kesminas

Projektas – Protmūšis

Data



Šiandien išmoksīte

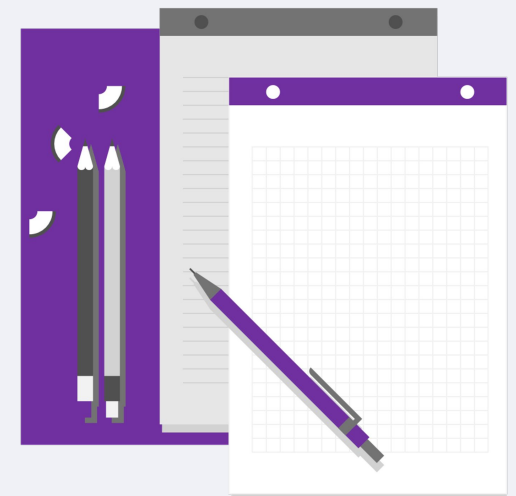
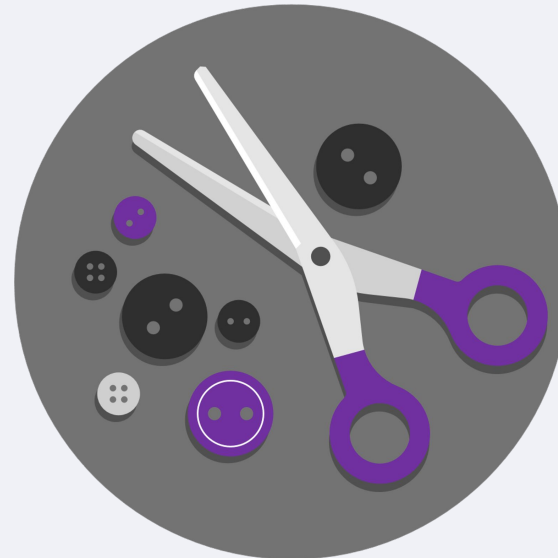
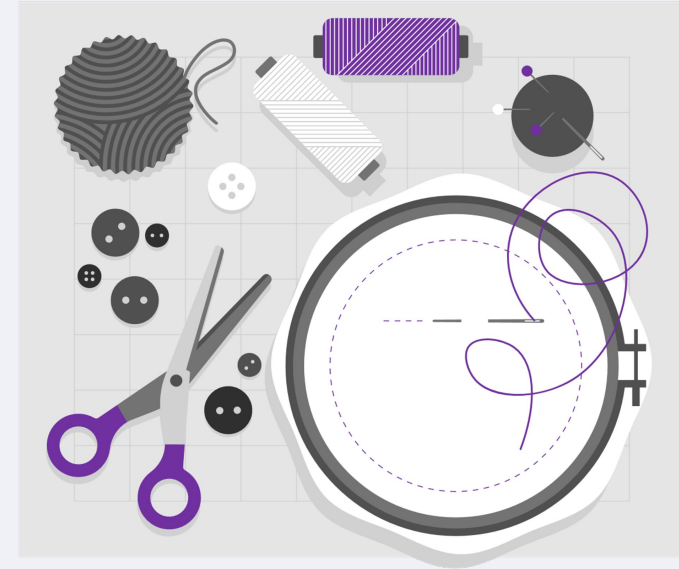
01

Projekts - Protmūšis



Projektas - Protmūšis

- Gavote užsakymą skaitmeninei Protmūšio programai suprogramuoti.
- Laikas: Dvi pamokos darbui, viena pamoka pristatymui.
- Šio uždavinio tikslas padės jums pakartoti tai ką jau išmokote ir duos tinkamas aplinkybes aplikuoti tai didesniame projekte.
- Sekdami reikalavimus ir funkcionalumo aprašymus turėsite sukurti ir pristatyti savo projektą.





Projektas nr. 1

- *Projektas yra skirtas jūsų įgūdžiams apšildyti prieš baigiamąjį darbą. Jis jums padės suprasti koks jausmas yra kurti didesnės apimties programą, kurioje tiek Jūs tiek Jūsų mokytojas turės progą pamatyti kur yra jūsų stipriosios vietos ir kurias Jūsų įgūdžių vietas būtų galima dar pagerinti (2-3 pamokos)*

- Sukurkite programą, kuri leistų prisijungti naudojant vardą ir pavardę. Prisijungus turėtų būti galimybė spręsti protmūšio uždavinius įvairiomis temomis. Programa turėtų leisti vartotojams pažiūrėti rezultatus.

Darydami projektą nedvejokite konsultacijoms ir susirašymui tiek su mokytojais tiek su grupiokais, bet nekopijuokite vieni nuo kitų identiško kodo, stenkitės rašyti patys.

- **Palengvinimas:**

- *Jeigu jaučiate, kad užduotys yra labai sudėtingos galite pakeisti sąlygą į „kurti žaidimą tik 1 žaidėjui“. Tokiu būdu jums reikės rinkti rezultatus tik vieno žaidėjo ir prisijungimas gali būti statiniai duomenys.*



Projektas nr. 1

Funckionalumas:

- Prisiijungimas
- Atsijungimas
- Žaidimo taisyklių atvaizdavimas
- Žaidimo rezultatų ir dalyvių peržiūra
- Dalyvavimas (Start game)
- Išėjimas iš žaidimo



Projektas nr. 1

- **Langai**
- Langas 1 (**Prisijungimas**): Įjungus programą programa su mumis turėtų pasisveikinti ir paprašyti prisijungti. Pirmas langas turėtų prašyti tik vardo ir pavardės įvesties. Daugiau nieko neatvaizduoti.
- Langas 2 (**Meniu**): Atlikus prisijungimą programa turėtų pasveikinti prisijungus ir atvaizduoti meniu punktus (Žiūrėti į funkcionalumo sąrašą)
- Langas 3 (**Parinktis**): Pasirinkus vieną iš meniu punktų programa turėtų atitinkamai atvaizduoti ekrane tai kas priklauso tai meniu parinkčiai (Žiūrėti detalesnius reikalavimus meniu punktams). „q“ įvestis turėtų mus grąžinti atgal į meniu.



Projektas nr. 1

- Prisijungimas:

- Prisijungimo lange turėtų turėti galimybę įvesti savo vardą ir pavardę. Įvedus vardą ir pavardę mūsų vartotojas turėtų būti užregistruotas kaip naujas įrašas. Rekomendacija naudoti Dictionary saugant vartotojus (Darome prielaidą, kad nebus antro vienodo vardo ir pavardės). Jei įvedame jau egzistuojančio vartotojo vardą ir pavardę programa turėtų mus išpėti, kad šiuo metu mes naudojame jau egzistuojančio vartotojo paskyrą. Jei tai nauja paskyra programa turėtų pasveikinti mus prisijungus ir susikūrus naują paskyrą.
- Kada sėkmingai būna prisijungta į ekraną turėtų išvesti meniu punktus. Programos metu nepamirškite kažkur laikyti šiuo metu prijungto vartotojo, kad žinotumėte kas šią akimirka žaidžia. Aktyviam vartotojui galite susikurti naują kintamąjį „CurrentUser“, kuris padės jūsų programai identifikuoti jį funkcijų metu. Konsolės viršuje visuose išskyrus prisijungimo langą turėtų matyti su koku vartotoju dabar esame prisijungę. Stilizuojant galite naudoti Console.Clear() metoda, kad išvalyti langą po parinkčių ir per naujo viską atvaizduoti.



Projekts nr. 1

- **Atsijungimas:**
- Programa turētu išvesti į ekraną, kad buvo atliktas atsijungimas ir paprašyti vartotojo įvesti vardą ir pavardę, kad prisijungti iš naujo.



Projektas nr. 1

- **Žaidimo taisyklių atvaizdavimas:**
- Programa turėtų išvesti į ekraną žaidimo taisykles. Kaip pvz galit pasiimti „Sveikiname prisijungus prie X protmūšio programos. Šis protmūšis jums leidžia pasirinkti iš X klausimų kategorijų. Pasirinkus kategoriją pradėsite žaidimą ir turėsite pasirinkti iš 4 galimų variantų, kuris yra jūsų klausimui teisingas atsakymas.“
- *Programa turėtų paprašyti vartotojo įvesti „q“ raidę jei nori grįžti atgal į meniu parinktis*



Projektas nr. 1

- Žaidimo rezultatų ir dalyvių peržiūra

- Pasirinkus šį meniu punktą ekrane turėtų pamatyti papildomą meniu. Jo viduje turėtų būti galimybę pasirinkti ar norime peržiūrėti dalyvius ar rezultatus. Jei pasirenkame dalyvius tada į ekraną turėtų išvesti visus dabartinius dalyvius. Jei pasirenkame matyti rezultatus į ekraną turėtų išvesti visus vartotojus pagal surinktų taškų skaičių mažėjimo tvarka. Prie TOP10 turėtų prirašyti pozicijos skaičių, o TOP 3 vardo gale žvaigždutę „*“ priklausomai nuo jo pozicijos pvz: 1 vieta 1 žvaigždutė ir t.t. Galite naudoti `.Values.ToList()`, o poto `.Sort` metodą.
- **Rekomendacija papildomoms sąlygoms:** *Jei darysite su keliomis klausimų kategorijomis ir nuspręsite laikyti atskirus Highscore kelioms klausimų kategorijoms galite naudoti `Dictionary<string, Dictionary<string, int>>`. Tokiu atveju jūsų sąrašas laikys kiekvienam vardui priklausančius kiekvienos kategorijos taškus. Paskyra `Dictionary<VardasPavardė, KategorijosDictionary<KategorijosVardas, Taskai>>`*



Projektas nr. 1

- **Dalyvavimas (Start game):**
- Priklausomai ar darote su keliomis kategorijomis programa turėtų išvesti kategorijų parinktį. Jei darote su viena kategorija žaidimas turėtų prasidėti ir vartotojui turėtų būti išvestas pirmas klausimas. Kiek klausimų per sesiją turėtų būti turėtumėte patys nuspręsti. Viršuje konsolės turėtų rašyti kelintas klausimas šiuo metu yra pvz 1/10. Vartotojas klausimo rezultata turėtų sužinoti iškart kai užtvirtina savo atsakymą parašydami savo atsakymo pasirinkimą pvz „1. Atsakymas 1“. Programa turėtų rašyti kiek šiuo metu taškų vartotojas turi. Programa turėtų klausti vartotojo klausimų tol kol pasiekia jūsų nustatytą sesijos klausimų kiekį. Kada visi klausimai būna atsakyti į ekraną turėtų išvesti vartotojo rezultatus: Kurie klausimai buvo atsakyti teisingai, kurie ne. *Taip pat turėtų atvaizduoti galutinius surinktus taškus ir šios sesijos poziciją tarp daugiausiai surinkusių taškų žaidėjų.* „q“ raidė turėtų vartotoją nusiųsti atgal į meniu. Kaip rekomendacija klausimams galite naudoti *List<T>* arba *Dictionary<T,K>*.



Projektas nr. 1

- Išėjimas iš žaidimo:
- Programa turėtų užsidaryti.



Projektas nr. 1

- **Papildoma (Darykite tik tuo atveju jei jau padarėte bazinį protmūšio funkcionalumą)**
- Padarykite, kad klausimai būtų parinkti atsitiktinai iš galimų klausimų
- Įdėkite kelių kategorijų parinktis
- Įdėkite salės pagalbą, kuri jums Random klasės pagalba turėtų „preliminariai“ nustatyti, kuris atsakymas yra teisingas (Tai galima padaryti parašant „s“ ir paspaudžiant „enter“)
- Įdėkite pagalbą atmesti 2 neteisingus variantus (Tai galima padaryti parašant „d“ ir paspaudžiant „enter“)
- Padarykite, kad skirtingi klausimai būtų verti skirtingo kiekio taškų priklausomai nuo jų kompleksškumo



Projekts nr. 1

- **Papildoma (Darykite tik tuo atveju jei jau padarēte bazinī protmūšio funkcionaluma)**
- Papildykite programā taip, kad programa rīkdama kokius klausimus uždūos vartotojui rīktū pagal tai kaip dažnai būna klausimas atsakomas teisingai. Pvz jei turite klausimā nr 5, kurī atsako 100% vartotojū ir turite klausimā nr 18, kurī atsako tik 10% programa turētū prioriteta dēti nr 18 klausimui.
- Įdēkite administratoriaus prisijungimā, kuris turētū prieigā prie papildomos meniu parinktīes „Konfigūracijōs“. Šiame lange galētumēt parinkti kiek klausimū per kiekvienā sesijā vartotojai turētū gauti. Pamastykite kaip reikētū sutvarkyti taškū koeficientā kada būna atliktas toks pakeitimas, nes jei prieš tai buvo 10 klausimū, o dabar 15 tada žmogus atsakēs 10/10 būtū skaičiuojamas kaip mažiau atsakēs nei 14/15.
- Įdēkite laikmatī ir sukurkite laiko limitacijas klausimams (Google helps)



<https://learn.microsoft.com/en-us/dotnet/api/system.console?view=net-7.0>

<https://learn.microsoft.com/en-us/dotnet/api/system.collections.generic.dictionary-2?view=net-7.0>

<https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/statements/iteration-statements>

<https://learn.microsoft.com/en-us/dotnet/api/system.collections.generic.list-1?view=net-7.0>

**Naudinga
informacija**