Python - შუალედური გამოცდა #1

--------------------------------

#1 მინი-ბიბლიოთეკა

--------------------

თქვენი მიზანია შექმნათ „მინი-ბიბლიოთეკის მართვის სისტემა“, სადაც:

- მომხმარებელს შეუძლია დაამატოს წიგნი (სათაური, ავტორი, წელი)

- შეინახოს ისინი სიაში

- მოძებნოს წიგნი სათაურით

- დაინახოს ყველა წიგნის სია (სათაური + ავტორი)

დეტალები:

-----------

1. შექმენით წინასწარი "სია" სადაც მოთავსებული იქნება 10 თქვენს მიერ არჩეული წიგნის სახელი/ავტორი და წელი

2. მომხმარებელს აქვს საშუალება: წიგნი დაამატოს ამ ფორმატით სახელი/ავტორი და წელი

3. მომხმარებელს აქვს საშუალება: ნახოს უკვე არსებული წიგნები რაც ბიბლიოთეკაშია (10 წიგნი)

4. აირჩიოს მისთვის სასურველი და გაიტანოს ბიბლიოთეკიდან "წასაკითხად"

5. მომხმარებლის მიერ დამატებული წიგნი უნდა დაემატოს ბიბლიოთეკის სიას და მომდევნო გამოძახებაზე 10-ის მაგივრად უნდა გამოჩნდეს 11 და ა.შ. სანამ მომხმარებელი დაამატებს წიგნებს

6. ვალიდაცია არაა საჭირო lower/upper და ა.შ. (მომხმარებელს ყოველთვის შეჰყავს სწორი ინფორმაცია)

#2 პროგრამა "21" ქულა

---------------------

წესები:

- დასტა შედგება 52 კარტისაგან - 4 ვარიანტი ("ყვავი" "ჯვარი" "გული" "აგური")

- 2-დან 10-მდე + Jack + Queen + King + Ace

- კარტების ღირებულება - 2-10მდე = თავისივე მნიშვნელობა (ანუ 3 = 3, 7 = 7 და ა.შ.) ; J,Q,K = 10 ; Ace = 11 (მარტივი ვარიანტი, უდრის მხოლოდ 11-ს)

- მოთამაშე და კომპიუტერი იღებენ კარტებს (თავდაპირველად 2-2, შემთხვევითობის პრინციპით)

- მოთამაშეს შეუძლია აირჩიოს "add" (დამატება) ან "stop" (გაჩერდეს)

- კომპიუტერი თავის მხრივ იღებს კარტებს მანამ, სანამ ქულა < 17

- ვინც 21-ს მიუახლოვდება, მაგრამ არ გადააჭარბებს, ის იგებს

თუ მოთამაშე მოიგებს გამოგვაქვს "თქვენ მოიგეთ" თუ კომპიუტერი "თქვენ წააგეთ" ფრეზე ვარიგებთ ხელახლა.

შეგიძლიათ გამოიყენოთ:

------------------------

1. ციკლები

2. პირობითი ოპერატორები

3. ფუნქცია

4. მოდულები

#3 ATM Machine

--------------

პროგრამა მუშაობს როგორც ბანკომატი:

- მომხმარებელს აქვს ანგარიშზე X თანხა რომელსაც წინასწარ ვინახავთ

- მომხმარებელს შეუძლია შეამოწმოს ბალანსი (ბრძანებით)

- მომხმარებელს შეუძლია თანხის: "გატანა", "შემოტანა", ბალანსის ნახვა

- აუცილებელი ვალიდაციები: ვერ უნდა გაიტანოს იმაზე მეტი რაც ანგარიშზე აქვს და ასევე ერთჯერადად ვერ უნდა შეიტანოს 1000ზე მეტი

- ბანკომატი მუშაობს მხოლოდ ლარზე

- ყველა ოპერაცია უნდა ჩაიწეროს ლოგერში და შეიხანოს ლოგ ფაილში (გატანა, შემოტანა)

- მომხმარებელს ყოველთვის შემოჰყავს სწორი ბრძანებები არაა საჭირო ცალკე ვალიდაციების გაწერა

#4 ლატარიის სიმულატორი

--------------------------

- კომპიუტერი ირჩევს 6 შემთხვევით რიცხვს 1–დან 49-მდე

- მოთამაშეს შეჰყავს 6 რიცხვი

- ითვლება, რამდენი დაემთხვა

- წინასწარ გაწერილია გასათამაშებელი თანხა

- logging ინახავს ყველა გათამაშებას და შედეგს

- მომხმარებელს ყოველთვის შემოჰყავს სწორი მნიშვნელობები

--------------------------------------------------------

ლოგიკა:

1. 6-6 დამთხვევის შემთხვევაში თამაშდება JACKPOT

2. 6-5 დამთხვევის შემთხვევაში JACKPOT-ის თანხას ვაკლებთ 40%-ს

3. 6-4 დამთხვევის შემთხვევაში JACKPOT-ის თანხას ვაკლებთ 60%-ს

4. 6-3 დამთხვევის შემთხვევაში JACKPOT-ის თანხას ვაკლებთ 80%-ს

5. 6-2 და 6-1 დამთხვევის შემთხვევაში მოთამაშე ვერაფერს ვერ იგებს

#5 რეგისტრაციის სიმულატორი

-------------------------------

მომხმარებელს ვარეგისტრირებთ ჩვენს ვებ-გვერდზე სადაც არის 4 ველი: "ელ-ფოსტა", "სახელი", "ზედმეტსახელი" და პაროლი

- სახელის გარდა ყველა ველი უკვე შევსებულია, რომელსაც ვინახავთ წინასწარ მაგ: user@mail.com, ... george777 და password123

- მომხმარებელს შეჰყავს მხოლოდ სახელი, რომელიც აუცილებლად უნდა იყოს ტექსტური ტიპი (ლათინური ასოები და პატარა ასოები)

- სხვა ნებისმიერი ტიპის მონაცემთა ტიპის შემთხვევაში უნდა გავუტანოთ შესაბამისი შეტყობინება

- მაგალითი 1: შემოიყვანა: 123 -> "შემოყვანილია რიცხვითი მნიშვნელობა, შემოიტანეთ მხოლოდ string პატარა რეგისტრში"

- მაგალითი 2: შემოიყვანა: @# -> "შემოყვანილია სიმბოლოები, შემოიტანეთ მხოლოდ string პატარა რეგისტრში" და ა.შ. ყველა შესაძლო ტიპი უნდა განსაზღვროთ

- ლათინურისგან განსხვავებულს უნდა დაუწეროთ რომ არის სხვა ენა

- მომხმარებელი თუ შემოიყვანს სტრინგს უნდა დაარეგისტრიროთ და გამოიტანოთ ყველა შენახული მონაცემი:"ელ-ფოსტა", "სახელი", "ზედმეტსახელი" და პაროლი