**რა გვინდა?**

აუქციონები შედგება ლოტებისგან. ლოტი არის ვაჭრობის საგანი.

ლოტი შეიძლება იყოს რამდენიმე. ერთი სესიის ფარგლებში განიხილება ერთი ლოტი. ე.ი. სესიის მონაწილეები არიან მოვაჭრეები და ლოტი.

ლოტის მახასიათებლები: **LotModel**

* დასახელება
* აღწერა
* რაოდენობა
* საწყისი ფასი (ერთეულის)
* მიმდინარე ფასი (ერთეულის)
* მინიმალური ბიჯი
* ბიჯების რაოდენობა
* აუქციონის ჩატარების თარიღი

ვაჭრობის ისტორია: **TradeHistoryModel**

* ლოტი
* მოვაჭრე
* აუქციონისტი
* მოვაჭრის დაფიქსირებული ფასი
* ბიჯის რიგითობა
* თარიღი

მოვაჭრეების მახასიათებლები: **TraderModel**

* სარეგისტრაციო ნომერი
* სახელი
* იურიდიული ფორმა
* საიდენტიფიკაციო ნომერი
* განაცხადის შემომტანის სახელი
* ხელმძღვანელის სახელი
* ხელმძღვანელის სტატუსი
* საკონტაქტო N
* მომსახურე ბანკის სახელი
* ანგარიშის ნომერი
* Swift
* აუქციონში მონაწილეობის თარიღი

Auctioneer

* სახელი
* გვარი
* მომხმარებელი
* პაროლი

აუქციონისტს არეგისტრირებს ადმინისტრატორი.

ადმინისტრატორი ბაზის დონეზე განისაზღვრება.

ლოტებს და მონაწილეებს არეგისტრირებს აუქციონისტი

აუქციონისტს უნდა ქონდეს ლოტის რეგისრაციის ფორმა და

მოვაჭრის რეგისტრაციის ფორმა. ლოტისა და მოვაჭრის რეგისტრაციის დროს აუქციონისტი უთითებს აუქციონის ჩატარების თარიღს და შემდგომში ამ თარიღის მიხედვით ხდება კონკრეტული აუქციონის მონაწილეების დადგენა.

ვაჭრობის პროცესში სანამ ერთი ლოტი არ გაიყიდება მეორე ლოტზე ვაჭრობა არ იწყება. ნებისიერ მოვაჭრეს შეუძლია მიმდინარე ლოტზე ფსონის დადება.

აუქციონისტი ლოტების რიგიდან იღებს მიმდინარე ლოტს, რომელიც გამოჩნდება ეკრანზე. ასევე გამოჩნდება აუცნიონის მონაწილეების სარეგისტრაციო ნომრები. აუქციონისტი რომელ ნომერსაც დააჭერს ეს ნიშნავს რომ ამ მონაწილემ გააკეთა ფსონი. ეს ინფორამცია უნდა ჩაიწეროს ბაზაში ვაჭრობის ისტორიის ცხრილში.

სერვისიები:

* **ინფორმაციის ბაზაში შეტანისთვის:**
  + ფუნქცია, რომელსაც უნდა გავატანოთ (LotModel) და ბაზაში დაამატებს ობიექტს.
  + ფუნქცია, რომელსაც უნდა გავატანოთ (TraderModel) და ბაზაში დაამატებს ობიექტს.
  + Edit ფუნქცია, რომელსაც უნდა გავატანოთ (LotModel) და ბაზაში დაარედაქტირებს ობიექტს.
  + Edit ფუნქცია, რომელსაც უნდა გავატანოთ (TraderModel) და ბაზაში დაარედაქტირებს ობიექტს.
* **ბაზაში ინფორმაციის შეტანის და გამოტანისთვის:**
  + ფუნქცია, რომელსაც უნდა გავატანოთ (TradeHistoryModel), ბაზაში დაამატებს ობიექტად და დაგვიბრუნებს (TradeHistoryModel).
  + Edit ფუნქცია, რომელსაც გავატანთ (TradeHistoryModel) ბოლო ობიექტის შეცვლილ ვერსიას და დაგვიბრუნებს (TradeHistoryModel).
* **ბაზიდან ინფორმაციის გამოტანისთვის:**
  + სრული და ნაწილობრივი ისტორიის დაბრუნება: - ფუნქცია, რომელსაც გავატანთ (Date) და დაგვიბრუნებს List <TradeHistoryModel> მონაცემებს გატანებული თარიღიდან მოყოლებული მიმდინარე თარიღთან უახლოესი ჩანაწერის ჩათვლით.
  + სრული და ნაწილობრივი ისტორიის დაბრუნება: - ფუნქცია, რომელსაც გავატანთ ლოტის id-ს და დაგვიბრუნებს List <TradeHistoryModel> მონაცემებს გატანებული თარიღიდან მოყოლებული მიმდინარე თარიღთან უახლოესი ჩანაწერის ჩათვლით გამარჯვებულების ჩამონათვალს.
  + სრული და ნაწილობრივი ისტორიის დაბრუნება: - ფუნქცია, რომელსაც გავატანთ Trader-ის id-ს და დაგვიბრუნებს List <TradeHistoryModel> მონაცემებს გატანებული თარიღიდან მოყოლებული მიმდინარე თარიღთან უახლოესი ჩანაწერის ჩათვლით გამარჯვებულების ჩამონათვალს.
  + Trader-ების პოვნა ყველა შესაძლო ჩანაწერის მიხედვით (უნდა ვატანდე Trader-ის მოდელს).
  + ლოტების პოვნა სახელის და თარიღის დიაპაზონის მიხედვით.
  + თარიღის მიხედვით ლოტზე გამარჯვებულების პოვნა.
* აუქციონისტის დარეგისტრირება.
* აუქციონისტის ავტორიზაცია.
* ადმინის ავტორიზაცია.