

Messenger Application

از شما خواسته شده تا برنامه پیام رسانی طراحی کنید که کاربران با اتصال به آن، امکان تبادل پیام و ارتباط با یکدیگر را داشته باشند.

بخش اول

فاز اول: در بخش اول قصد داریم یک پایلاد سازی شبکه (Socket Programming) به دو کاربر آلیس و باب امکان ارسال پیام به سرور را بدهیم. به این صورت که پیام رسانی server می‌باشد و آلیس و باب client برای سرور هستند که با اتصال به آن امکان ارسال پیام و دریافت جواب از او دارند. برای پیاده سازی نیاز به کلاس های Server.java، Alice.java و Bob.java داریم.

ابتدا کلاس های Server.java و Alice.java را به گونه ای تعریف کنید که کلاینت (آلیس) به سرور وصل شود و پس از اتصال، سرور به آلیس پیام "Enter name:" را ارسال کند. آلیس با دریافت این پیام و چاپ آن رشته "Alice" را وارد می‌کند و این رشته به سرور ارسال می‌شود. در مرحله بعد نیز به همان شکل سرور از آلیس رمز او را می‌پرسد و آلیس در جواب پیامش، رمز را وارد می‌کند و برای سرور ارسال می‌کند.

سرور پس از دریافت اسم و رمز آلیس، آنها را در آرایه ای از جنس User (که صرفاً دارای ویژگی های نام و رمز میباشد) ذخیره می‌کند. پس از آنکه اطلاعات آلیس در این آرایه ذخیره شد، سرور منتظر دریافت رشته از آلیس است و او هر پیامی که به سرور ارسال کند، سرور همان متن را به صورت اکو ارسال می‌کند. برای مثال اگر آلیس رشته "Hi!" را وارد کند، سرور این رشته را دریافت کرده و عبارت "Echo: Hi!" را برای آلیس ارسال می‌کند و آلیس آن را چاپ می‌کند.

فاز دوم: اگر آلیس با سرور در ارتباط باشد اما ناگهان باب خواستار اتصال به سرور باشد، سرور نمی‌تواند به صورت موازی با آلیس و باب ارتباط برقرار کند. در ادامه می‌خواهیم با استفاده از Multi-Thread Programming (برنامه نویسی چند نخی) امکان همزمان باب و آلیس (و حتی تعدادی بیشتر کاربر) به این پیام رسانی را فراهم کنیم.

برای این کار کافیست هر زمان کلاینتی درخواست اتصال به سرور داد، سوکتی از طرف سرور برای او اختصاص یابد که این کار با اختصاص دادن یک ترد به هر کلاینت ممکن می‌شود. برای مثال، آلیس به سرور درخواست اتصال می‌دهد و سرور تردی به وجود می‌آورد که با آن سوکت مخصوص آلیس را مدیریت می‌کند. سپس هنگامی که باب درخواست اتصال به سرور بدهد، سرور با مشاهده این درخواست، ترد دیگری به باب اختصاص می‌دهد تا عملیات او را انجام دهد و همزمان ارتباط خود با آلیس را حفظ کند. (راهنمایی: اضافه کردن یک کلاس ClientHandler.java که از Runnable ارث بری می‌کند)

نمره اضافه:

- اعمال Synchronization برای مدیریت بخش های بحرانی (در صورت نبود بخش بحرانی آن را با یک متغیر مشترک به وجود آورید).
- داشتن گرافیک با JavaFX