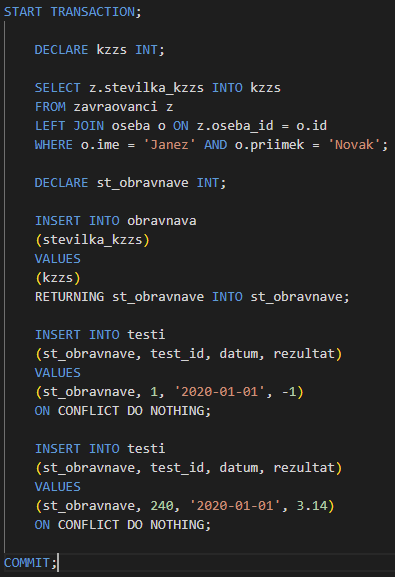
4. domača naloga

Aleksander Piciga

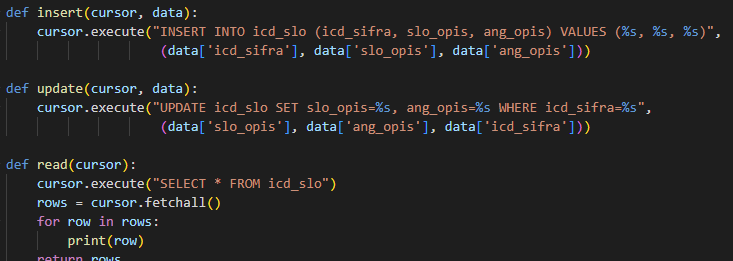
## A Naloga

Pacient pride na nujno sprejemno točko zdravstvenega doma, zaradi nenadnega poslabšanja stanja. Ker je zavarovani pacient že bil na spremljanju zaradi slabšega delovanja jeter, se ga takoj napoti na nujne hitre teste v specializirani laboratorij. Rezultati testov so na voljo v slabih 15minutah, med tem pa zdravnik opravi nujni pregled preden se pacienta napoti na pravilni oddelek za nujno oskrbo. Zaradi nujnoti testov in ker je pacient že v sistemu je po napotitvi na kritični oddelek potrebno vnesti novo obravnavo (poslabšanje stanja) in vnesti takoj izvedene teste. Pacientove podatke se dobi na podlaki imena, saj je že v sistemu in v hitrem odvijanju dogodkov ni časa za iskanje kartice s kzzs številko.



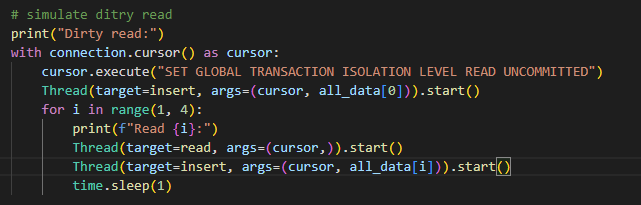
## Naloga

Uporabljene funkcije, ki se izvajajo na ločenih nitih:



Vsaka od teh funkcij prejme cursor za izvajanje operacij in podatke s katerimi deluje. Funkcija read podatke izpiše, da lahko opazimo spremembe. Tabela icd\_slo je za večjo preglednost pred izvajanjem izpraznjena.

**DIRITY READ**



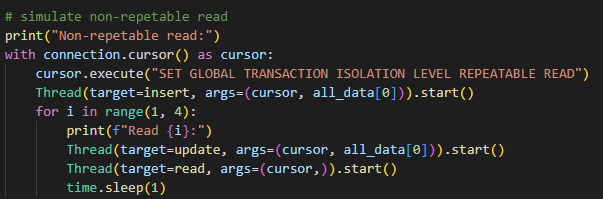
Potek transakcij / akcij:

* Začetek inserta (T1)
* Select prebere rezultate (T2), vključno z uncommited zapisom od T1
* Insert se zavrže (T1)
* Select prebere rezultate (T2) – vstavljenega podatka ni več

**Rešitev:**

Stopnjo izolacije nastavimo na katerokoli izmed: READ COMMITED, REPETABLE READ, SERIALIZABLE. Vse te preprečijo branje podatkov od inserte (T1) dokler se ne izvede commit stavek znotraj T1.

**NON-REPETABLE READ**

****

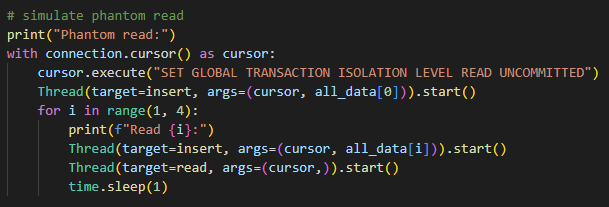
Potek transakcij / akcij:

* Izvede se select od T1
* Podatek posodobi T2
* Izvede se commit T2
* Izvede se novi select znotraj T1 (ta prebere posodobljen podatek od commited T2)

**Rešitev:**

Stopnjo izolacije nastavimo na REPEATABLE READ ali SERIALIZABLE. Stopnja READ UNCOMMITED ne zadošča, saj je T2 že izvedel commit vendar vseeno pride do neželjenega obnašanja.

**PHANTOM READ**

****

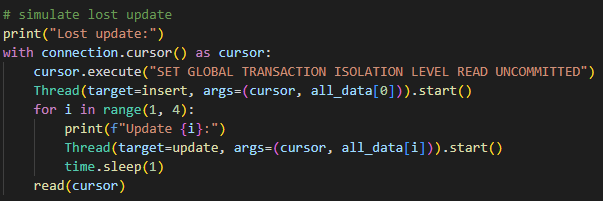
Potek transakcij / akcij:

* Izvede se select od T1
* Podatek vstavi T2
* Izvede se commit T2
* Izvede se novi select znotraj T1 (ta prebere **nov** podatek od commited T2)

**Rešitev:**

Gre za posebno obliko non-repetable reada vendar sedaj T2 ne izvaja updata pač pa insert, zato stopnja izolacije REPEATABLE READ ne zadošča več. Potrebujemo SERIALIZABLE stopnjo izolacije.

**LOST UPDATE**

****

Potek transakcij / akcij:

* Izvede se update od T1
* Izvede se update od T2
* Izvede se commit T1
* Izvede se commit T2

Ohrani se zgolj zadnja posodobitev.

**Rešitev:**

Uporaba REPEATABLE READ ali pa SERILIZABLE.