

## Zadání úlohy Java RMI: Databázový server

### Zadání:

Implementujte server, který nabídne základní možnost ukládání dat. Server bude mít možnost obsluhovat vícero 'databází' a vícero klientů současně.

### Datové struktury:

Server a klient budou používat následující datové struktury (objekty):

- dbname – String  
identifikace databáze – textový řetězec (max. délka 50).
- DBRecord  
struktura obsahující jeden databázový záznam  
skládá se z:
  - tscreate – časové razítko vzniku záznamu – čas vytvoření záznamu na serveru  
(String – '2011-09-23 14:34:56')
  - tsmodify – časové razítko změny záznamu – čas poslední změny záznamu na serveru  
(String – '2011-10-02 14:34:57')
  - key – klíč – jednoznačný identifikátor (Integer)
  - message – zpráva – vlastní zpráva (String)

### Funkce poskytované serverem:

Server bude klientovi poskytovat následující funkce (metody):

- String[] listDB()  
vypíše existující databáze
- boolean createDB(String dbname) throws DBExistException  
vytvoří databázi daného jména
- Integer insert(String dbname, Integer key, String message) throws DBNotFoundException, DuplicateKeyException  
vytvoří v databázi nový záznam
- Integer update(String dbname, Integer key, String message) throws DBNotFoundException, KeyNotFoundException  
aktualizuje záznam daný klíčem key na hodnotu message
- DBRecord get(String dbname, Integer key) throws DBNotFoundException, KeyNotFoundException  
vrátí zprávu příslušející danému klíči
- DBRecord[] getA(String dbname, Integer[] key) throws DBNotFoundException, KeyNotFoundException  
vrátí pole záznamů příslušející daným klíčům, operace se povede bez chyby pouze tehdy, pokud se povedou najít všechny odpovědi
- void flush()  
zapiše změny na disk

### Implementační poznámky:

Databáze bude reprezentována souborem (co databáze to jeden soubor – např. databáze osoby bude uložena v souboru osoby.dbcsv). Záznam bude zapsán formátem CSV. V rámci zjednodušení není nutné, aby v době běhu byl synchronizován obsah 'programové' databáze se souborovou reprezentací (o synchronizaci se postará funkce flush()). Po startu databázového serveru serveru bude stav serveru synchronizován s uloženými soubory (tzn. vrátí se do stavu, ve kterém byl ukončen). Musí být umožněno, aby mohlo být připojeno více klientů pracujících s různými databázemi (nutná atomicita některých operací).

Poznámka: Doporučuje se u všech proměnných typu String nepoužívat řetězce obsahující uvozovky nebo středníky (kvůli kolizím v CSV).

Ukázka databázového souboru adresy.dbcsv:

```
"2011-09-23 14:34:56";"2011-09-24 14:34:56";"1";"Jan Novotný, Zikova 7, Praha"  
"2010-03-20 18:23:56";"2011-09-21 14:26:56";"256";"Adéla Krátká, Lovosice 345"  
"2011-08-20 14:23:56";"2011-09-24 14:24:56";"666";"Dr. Zlo, Pekelná 666, Gomora"
```

### Dávkové zpracování

Vzhledem k urychlení odevzdávání je požadováno aby úlohy šly spouštět z příkazové řádky a aby klient umožňoval dávkové zpracování.

```
./server [IP_adresa] [cislo_portu]
```

, kde IP\_adresa a cislo\_portu specifikuje, kde má server naslouchat (v případě nezadání parametrů bude server poslouchat na všech síťových rozhraních a využije standardní porty)

./client IP\_adresa cislo\_portu konf\_soubor  
, kde IP\_adresa a cislo\_portu specifikují, kde je potřeba hledat server a v souboru je posloupnost příkazů pro klienta. Příkazy jsou psány pomocí CSV a kopírují požadované funkce. V případě že příkaz skončí chybovým stavem (výjimka), je uživatel upozorněn a pokračuje se ve vyhodnocování následujícího příkazu Např.

```
"listdb"  
"createdb";"adresy"  
"insert";"adresy";"1";"Jan Novotný, Zikova 7, Praha"  
"insert";"adresy";"256";"Adéla Krátká, Lovosice 345"  
"insert";"adresy";"666";"Dr. Zlo, Pekelná 666, Gomora"  
"update";"adresy";"256";"Adéla Dlouhá, Lovosice 345"
```

#### ***Ukázkový výstup:***

Obsah souboru konf\_soubor

```
"listdb"  
"insert";"adresy";"1";"Jan Novotný, Zikova 7, Praha"  
"createdb";"adresy"  
"listdb"  
"insert";"adresy";"256";"Adéla Krátká, Lovosice 345"  
"update";"adresy";"256";"Adéla Dlouhá, Lovosice 345"  
"ghCHR6";"parametr"
```

Výstup z klientské aplikace

```
>> listdb  
<< Databases:  
>> Inserting into database "adresy" record ["1";"Jan Novotný, Zikova 7, Praha"]  
<< ERROR - Database "adresy" does not exists  
>> createdb adresy  
<< Database "adresy" created  
>> listdb  
<< Databases: 'adresy'  
>> Inserting into database "adresy" record ["256";"Adéla Krátká, Lovosice 345"]  
<< DB "adresy" - record inserted  
>> Updating record "256" in database "adresy" with "Adéla Dlouhá, Lovosice 345"  
<< DB "adresy" - record updated  
>> ERROR – command "ghchr6" not implemented
```