

Inlämning Effektiv Python

Du ska bygga ett banksystem. Känns det igen? Nu ska vi dock bygga det med fokus/attention på bra och effektiv kod. Vi ska testa olika saker och mäta (=lära) vad som händer om vi implementerar olika algoritmer och datastrukturer.

Detta är en ITERATIV övning så du får INTE bygga sista versionen redan nu, utan varje version ska vara en egen BRANCH på Github

Hur lämnar ni in:

Github (privat) och ge mig rättighet att se. Dvs bjud in stefan.holmberg@systemmentor.se som Collaborator. Sen skriver du in URL:en till Github repositoryt som svar (kommentarsfältet) här. Sen under handledningstid (skola) ska du kunna VISA din kod för Stefan/Simon och prata/förklara din kod.

FÖLJ SPEC = (står det lista är det inget dictionary osv osv)

VERSION 1

Du ska ha en LISTA med OBJEKT (kund-klass)

Varje kund ska ha:

created (timestamp)

last updated (timestamp)

name

birthdate (date)

kontonummer (1234-1234567890) formatvalidering (4 siffror sedan - sedan 10 siffror)

saldo

FUNKTIONALITET

Det som skall implementeras är ju bla en GetAccountNo som loopar igenom listan efter ett specifikt konto och returnerar det

- Skriv en som skapar 100000000 (10 miljoner) kunder och lägger in i din lista. Kontonummer ska läggas in i SEKVENSIELL ORDNING

1111-0000000001, sen 1111-0000000002, 1111-0000000003 osv osv

Du ska INTE använda list comprehension utan en helt vanlig for-loop

- Skriv ut på skärmen hur lång tid det tar (ms) att skapa kunderna

- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-0000001000.

- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-0009999999.

- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-9999999999 (finns ju ej)

VERSION 2

Ändra så du använder en DICTIONARY (kontonummer blir nyckel)

Ändra så dina kunder skapas med LIST COMPREHENSION

1111-0000000001, sen 1111-0000000002, 1111-0000000003 osv osv

- Skriv ut på skärmen hur lång tid det tar (ms) att skapa alla kunder nu

- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-0000001000.

- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-0009999999.

- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-9999999999 (finns ju ej)

VERSION 3

Lägg tillbaka så det är en lista. Lägg in i RANDOM ORDER (generera kontonummer random) - men verifiera så det inte blir dubletter
Sortera listan efter du skapat den.

För VG använd <https://stackabuse.com/quicksort-in-python/> som mall (quicksort) och modifiera den så den sorterar dina konton på kontonummer

För G använd inbyggda sort eller en Bubblesort eller vad du vill för att sortera på kontonummer
Implementera shortcutting - dvs om du letar efter XXXX-000001232 men kommer till XXXX-000001233 utan träff avbryt !

- Skriv ut på skärmen hur lång tid det tar (ms) att skapa alla kunder nu
- Skriv ut på skärmen hur lång tid det tar (ms) att sortera alla kunder

VERSION 4 VG

Implementera binärsökning när man söker i listan

- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-0000001000.
- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-0009999999.
- Skriv ut på skärmen hur lång tid det tar (ms) att söka i listan för att hitta 1111-9999999999 (finns ju ej)