

In-memory Database

- محدودیت زمان: ۵۰۰ میلی‌ثانیه
- محدودیت حافظه: ۵۰ مگابایت

هدف تمرین: یادگیری عمیق‌تر *static storage class*, *dynamic memory allocation*, *const function*, *const function args*

در این تمرین از شما انتظار می‌رود کلاسی طراحی کنید که بتواند تعدادی *key-value* (جفت مرتبی از رشته در نظر بگیرید) را درون حافظه نگهداری کند.

Key	Value
K1	AAA,BBB,CCC
K2	AAA,BBB
K3	AAA,DDD
K4	AAA,2,01/01/2015
K5	3,ZZZ,5623

طراحی کلاس این پایگاه داده باید به گونه‌ای باشد که در تمام طول عمر برنامه، تنها یک نمونه (Instance) از این پایگاه داده ساخته شود. برای این منظور باید از مفهوم [Singleton design pattern](#) بهره ببرید.

این کلاس حداقل باید توابع زیر را داشته باشد:

- تابع `getInstance` (طبق طراحی Singleton)
- تابعی که یک جفت رشته (`key` و `value` متناظر با آن) ورودی می‌گیرد و در حافظه نگهداری می‌کند (اگر این `key` از قبل وجود داشت، تغییری ایجاد نکند)
- تابعی `const` که به رشته به عنوان `key` ورودی می‌گیرد و `value` متناظر با آن در حافظه را بازمی‌گرداند (اگر `key` در حافظه موجود نبود، رشته خالی بازگرداند)
- تابعی `const` که تعداد `key-value`های موجود در حافظه را بازمی‌گرداند

حال در ادامه، می‌خواهیم برنامه‌ای بنویسیم تا از این پایگاه داده استفاده کند و مقادیری در آن بنویسد یا از آن بخواند.

پیاده‌سازی Singleton

از این پترن زمان‌هایی استفاده می‌شود که نیاز است مطمئن باشیم تنها یک آبجکت از کلاس موردنظر ساخته می‌شود. (به خاطر اصل توحید، اگر قرار بود خدا رو با کلاس مدل کنیم، یقیناً سینگلتون می‌کردیم تا در هیچ لحظه‌ای دوتا خدا نداشته باشیم!) ما هم باید به این موضوع توجه داشته باشیم و اجازه کپی کردن یا تولید پایگاه داده‌ای جدید را از برنامه‌نویسان

برای پیاده‌سازی از مراحل زیر کمک بگیرید:

- تابعی استاتیک به اسم `getInstance` تعریف کنید تا در جواب رفرنسی به آبجکت یکتای ما برگرداند
- دیفالت کانستراکتور را `private` تعریف کنید
- کپی کانستراکتور را پرایوت کنید و تعریفی برای آن ارائه نکنید

برای توضیحات بیشتر به این [پیوند](#) سر بزنید.

پی‌نوشت

- مفهوم **in-memory database** را به صورت ساده، آبجکتی فرض کنید که مقادیر را در حافظهٔ اصلی ذخیره می‌کند و در آینده همان مقادیر را بازپس می‌دهد.
- توجه داشته باشید که محدودیتی در تعداد `key-value`ها وجود ندارد! تنها محدودیت، مقدار فضای خالی رم خواهد بود!
- توابعی که امکان `const` شدند دارند، `const` تعریف کنید.
- ورودی توابع را تا جایی که امکانش وجود داشت، `const reference` تعریف کنید.

ورودی و خروجی

ابتدای هر خط ورودی با یک دستور شروع می‌شود:

- دستور `add`: پس از این دستور، یک رشته به عنوان `key` و یک رشته به عنوان `value` داده می‌شود
- دستور `get`: پس از این دستور، یک `key` داده می‌شود و انتظار می‌رود `value` متناظر با آن در خط بعدی چاپ شود
- دستور `size`: این دستور ورودی ندارد. پس از دریافت این دستور، تعداد `key-value`های درون حافظه چاپ شود
- دستور `exit`: این دستور ورودی ندارد. برنامه پس از دریافت این دستور باید بسته شود

ورودی نمونه

```
add users_count 1000
size
add online_users_count 20
get users_count
add users_count 2000
get users_count
get users
size
exit
```

خروجی نمونه

1

1000

1000

2