```
interface NetbankingService {
  double getBalance(String accountNo);
  Account getAccountDetails(String
accountNo);
}

class NetbankingServiceImpl implements
 NetbankingService {
  // implementation logic
}
```

```
ApplyCacheInvocationHandler handler = new ApplyCacheInvocationHandler(new NetbankingServiceImpl());
```

cglib/
javaassist

```
NetbankingService proxy = Proxy.newProxyInstance(NetbankingService.class.getClassLoader(),
                            new Class<?>[] {NetbankingService.class}, handler);
```

```
class ApplyCacheInvocationHandler implements
InvocationHandler {
    Object original;

    ApplyCacheInvocationHandler(Object original) {
        this.original = original;
    }

    public Object invoke(Object proxy, Method method,
Object[] args) {
        Cache cache = null;
        String key = null;

        cache = Cache.getInstance();
        key = method.getName() + "(";
        for(Object arg: args) {
            key+=","+ arg;
        }
        key+=")";

        if(cache.containsKey(key)) {
            return cache.get(key);
        }else {
            Object ret = method.invoke(original, args);
            cache.put(key, ret);
            return ret;
        }
    }
}
```

#Runtime
Proxy

```
double balance = proxy.getBalance("ac1001");
double balance = proxy.getBalance("ac1001");
```