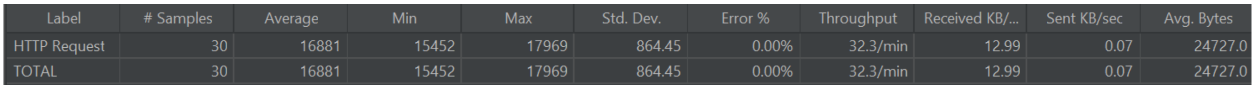
Customer cards

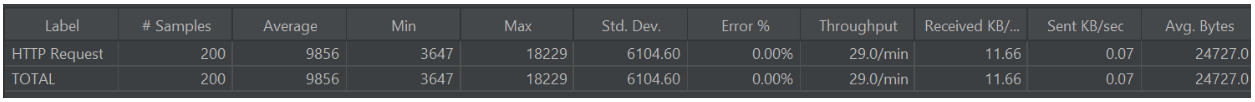
Before:



Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
HTTP Request	30	16881	15452	17969	864.45	0.00%	32.3/min	12.99	0.07	24727.0
TOTAL	30	16881	15452	17969	864.45	0.00%	32.3/min	12.99	0.07	24727.0

Fix:
Don’t repeat method calls, instead of calling **CustomersDAO.getInstance().getCustomers()** multiple times, retrieve the customers once and store them in a variable
Calculate the average spendings outside the loop to avoid recalculating it for each customer

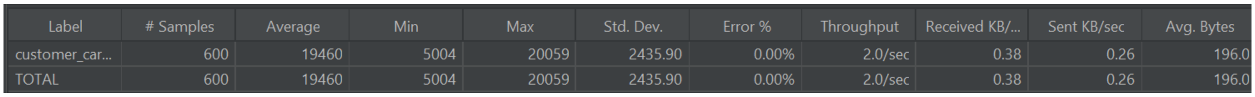
After:



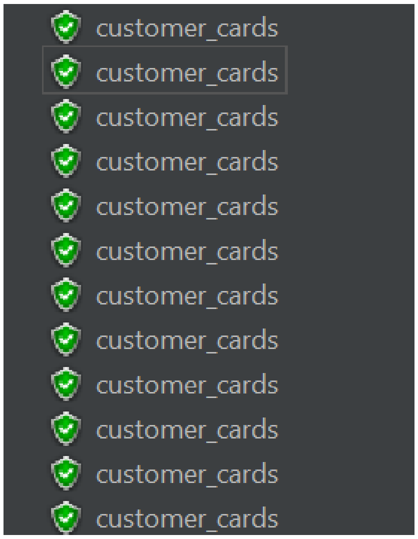
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
HTTP Request	200	9856	3647	18229	6104.60	0.00%	29.0/min	11.66	0.07	24727.0
TOTAL	200	9856	3647	18229	6104.60	0.00%	29.0/min	11.66	0.07	24727.0

Print cards

Average throughput:

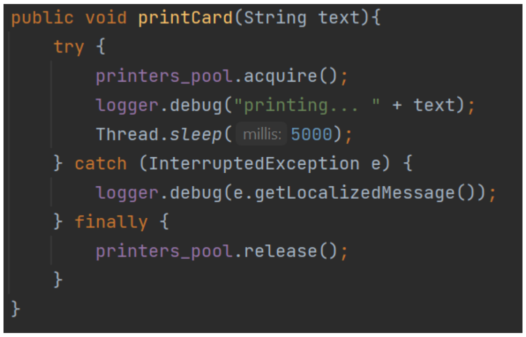


Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
customer_car...	600	19460	5004	20059	2435.90	0.00%	2.0/sec	0.38	0.26	196.0
TOTAL	600	19460	5004	20059	2435.90	0.00%	2.0/sec	0.38	0.26	196.0



✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards
✓ customer_cards

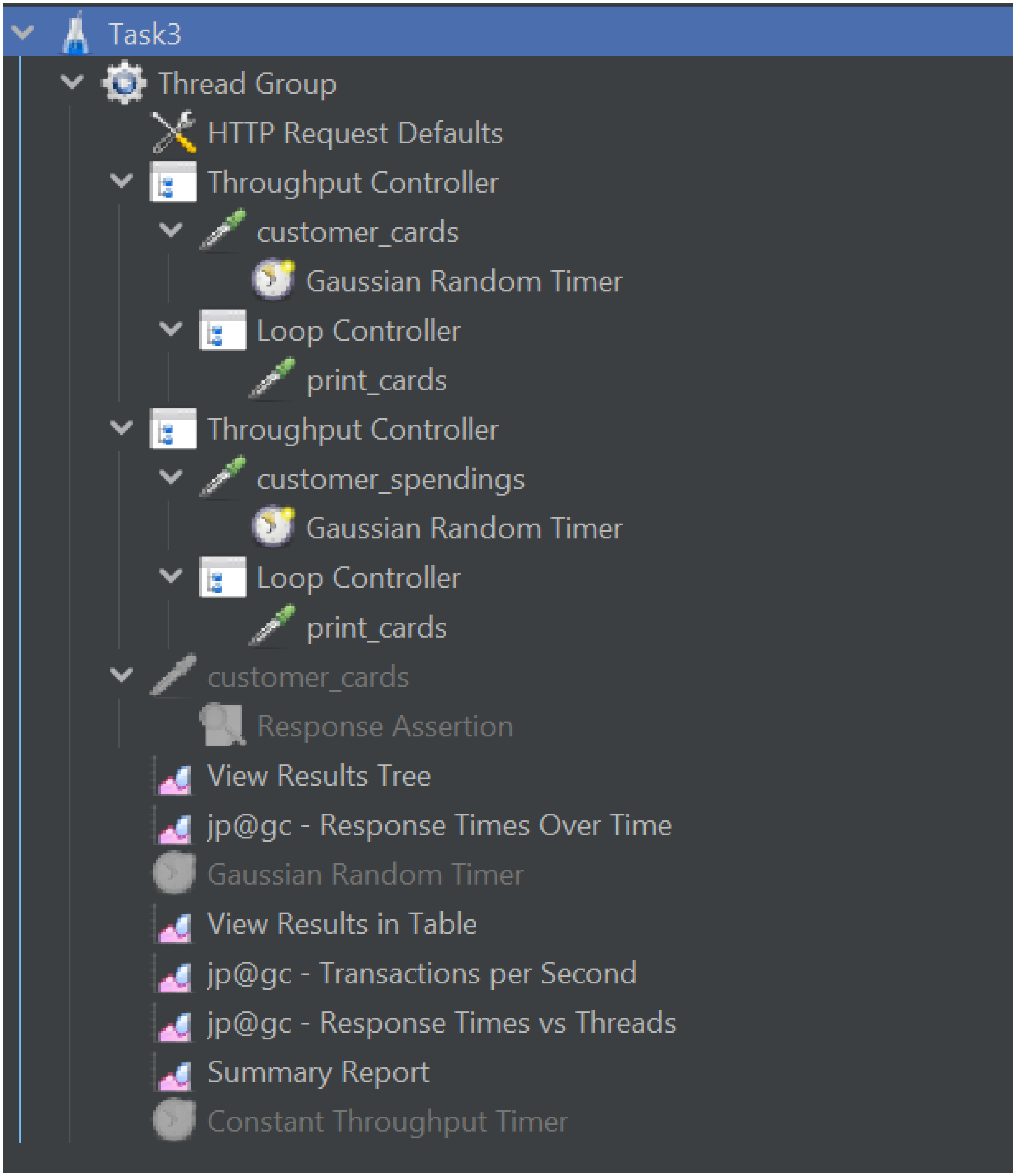
How many printers:



```java
public void printCard(String text){
    try {
        printers_pool.acquire();
        logger.debug("printing... " + text);
        Thread.sleep( millis: 5000);
    } catch (InterruptedException e) {
        logger.debug(e.getLocalizedMessage());
    } finally {
        printers_pool.release();
    }
}
```

It takes a single printer 5 seconds to print a card so through, with 10 printers we get a ratio of 2 cards/second which is equal to the throughput from testing. Each printer increases the throughput by 0.2 so in order to achieve ~5.5 cards/second we would need 28 printer, so in conclusion:
We would need 18 additional printers to meet the quota

Task 3:



```
Task3
  Thread Group
    HTTP Request Defaults
    Throughput Controller
      customer_cards
        Gaussian Random Timer
      Loop Controller
        print_cards
    Throughput Controller
      customer_spendings
        Gaussian Random Timer
      Loop Controller
        print_cards
    customer_cards
      Response Assertion
    View Results Tree
    jp@gc - Response Times Over Time
    Gaussian Random Timer
    View Results in Table
    jp@gc - Transactions per Second
    jp@gc - Response Times vs Threads
    Summary Report
    Constant Throughput Timer
```