

Chapter 1: Type Inference

Find the number hidden by the colored pattern

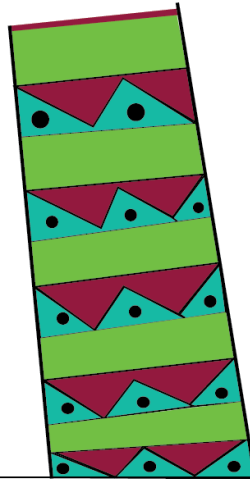
16

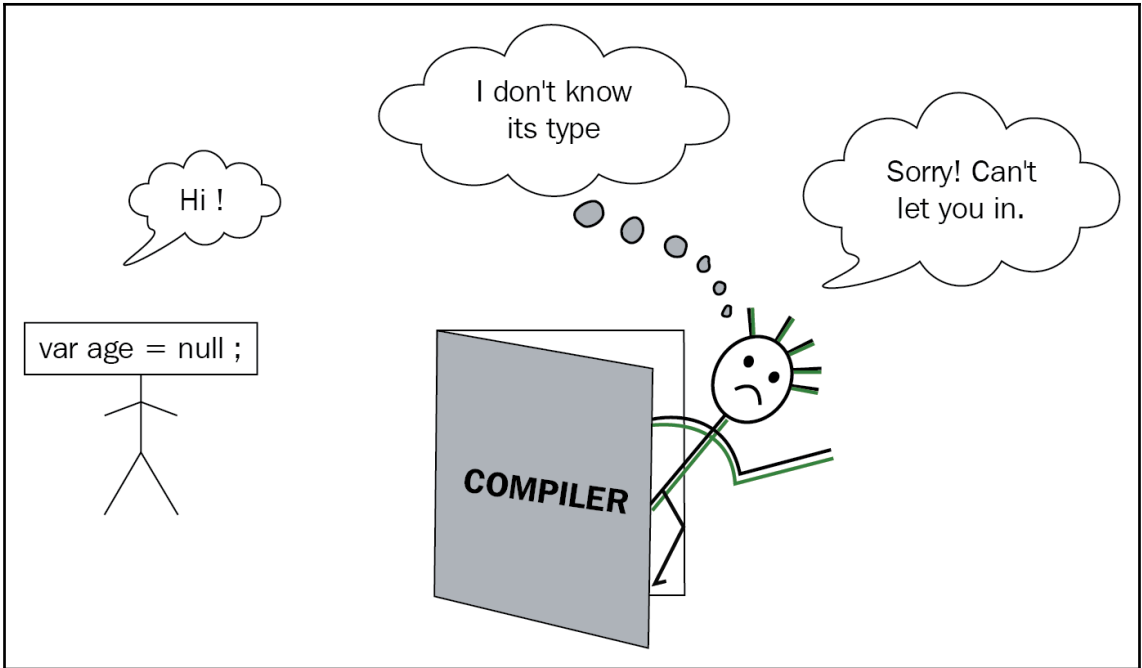
06

68

88

98

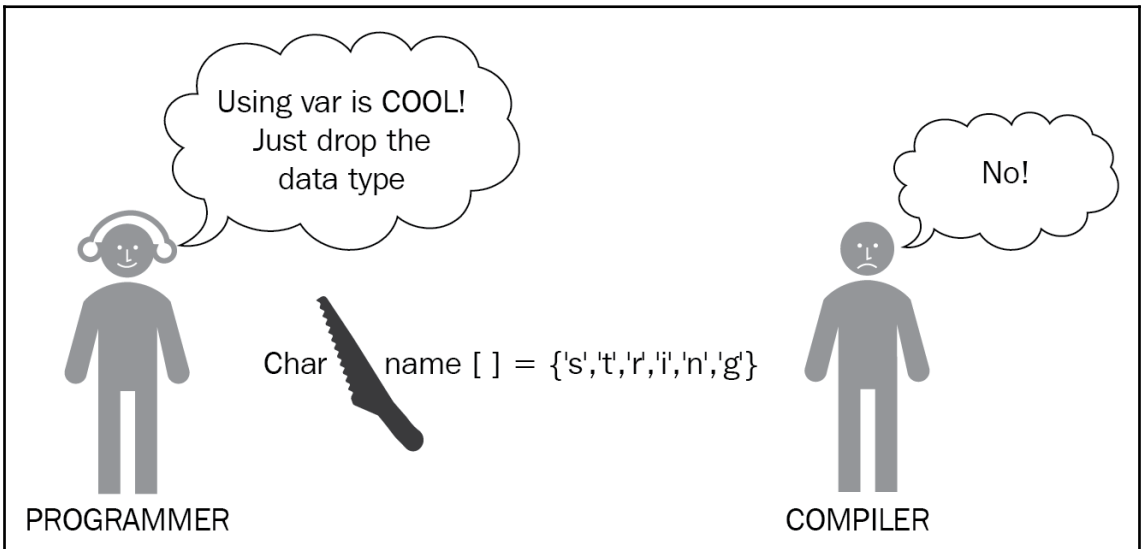
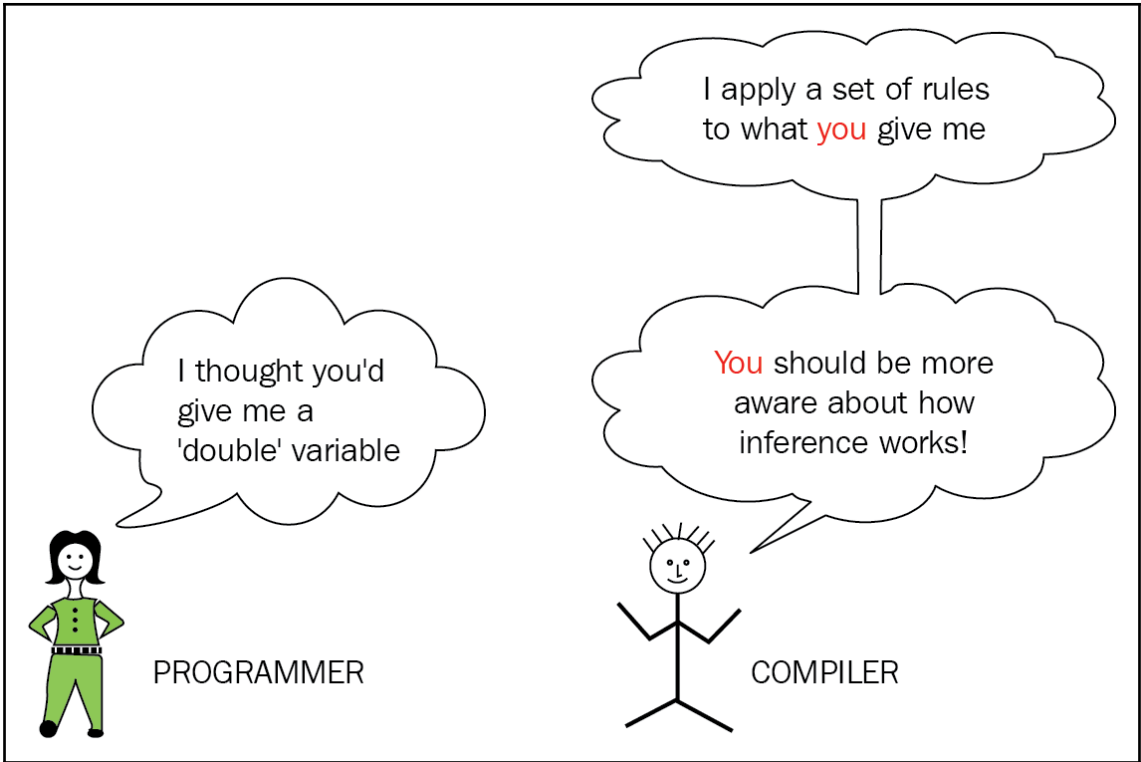


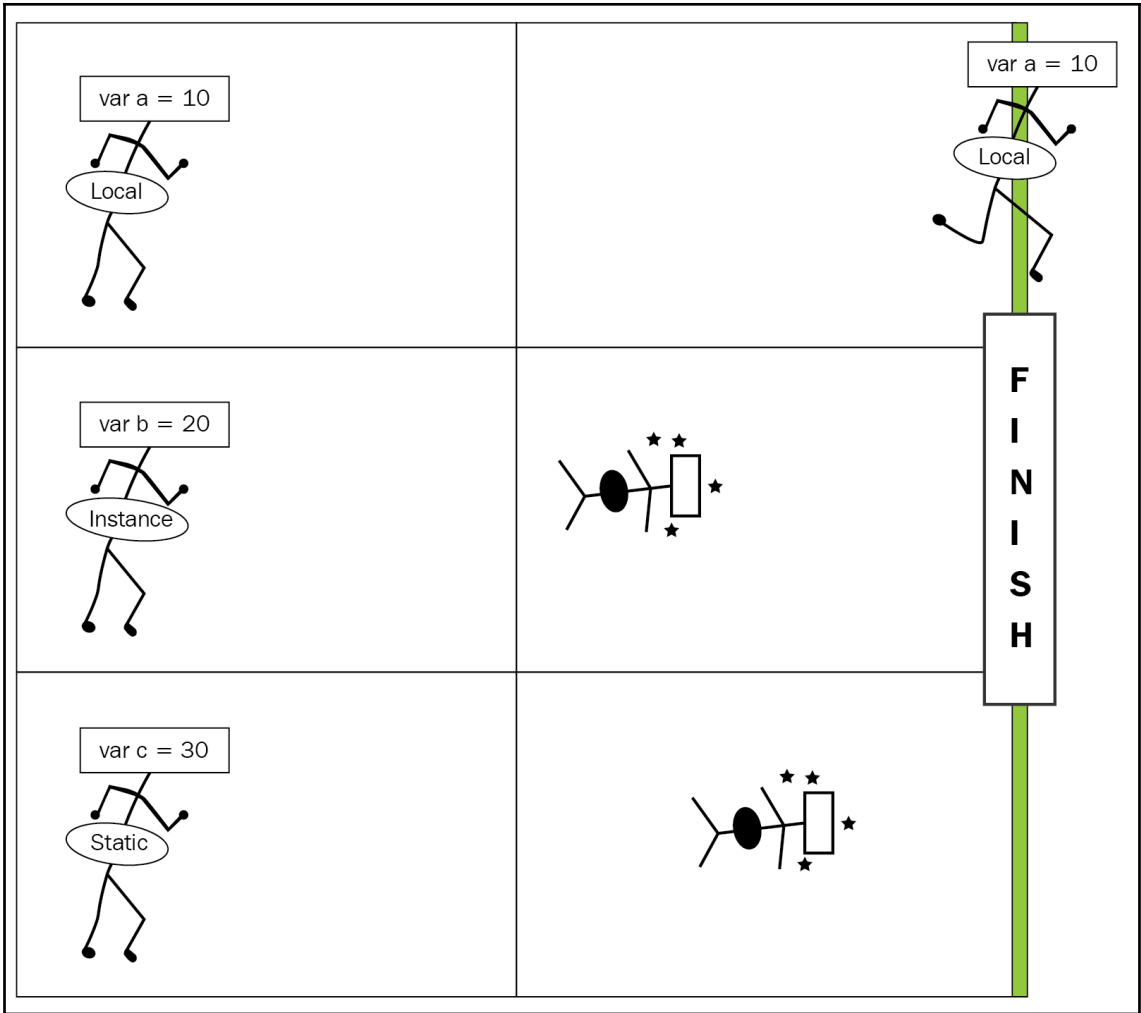


```

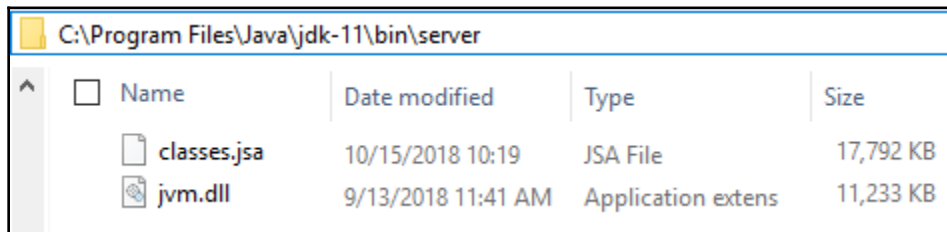
class Person {
    {
        } // instance initializer block
    static {
        } // static initializer block
    Person () {
        } // constructor
        if (....) {
            } // if construct
        }
    void aMethod () {
        switch (..) {
            case .. : {
                } // switch constructs
            }
        } // at beginning,
        // middle or end
        // of a method
    for ( ) {
        } // for - loop initialization
        // for loop body
        // do, do - while loops
    }
}

```





Chapter 2: AppCDS



Name	Date modified	Type	Size
classes.jsa	10/15/2018 10:19	JSA File	17,792 KB
jvm.dll	9/13/2018 11:41 AM	Application extens	11,233 KB

```
C:\Users\Mala Gupta >java -Xshare:dump
narrow_class_base = 0x0000000800000000, narrow_class_shift = 3
Allocated temporary class space: 1073741824 bytes at 0x00000008c0000000
Allocated shared space: 3221225472 bytes at 0x0000000800000000
Loading classes to share ...
Loading classes to share: done.
Rewriting and linking classes ...
Rewriting and linking classes: done
Number of classes 1273
  instance classes = 1213
  obj array classes = 52
  type array classes = 8
Updating ConstMethods ... done.
Removing unshareable information ... done.
Scanning all metaspace objects ...
Allocating RW objects ...
Allocating RO objects ...
Relocating embedded pointers ...
Relocating external roots ...
Dumping symbol table ...
Relocating SystemDictionary::_well_known_classes[] ...
Removing java_mirror ... done.
mc space:      8416 [ 0.0% of total] out of 65536 bytes [ 12.8% used] at 0x0000000800000000
rw space: 4034120 [ 22.2% of total] out of 4063232 bytes [ 99.3% used] at 0x0000000800010000
ro space: 7332808 [ 40.4% of total] out of 7340032 bytes [ 99.9% used] at 0x000000080003f0000
md space:    2560 [ 0.0% of total] out of 65536 bytes [ 3.9% used] at 0x00000008000af0000
od space: 6556656 [ 36.1% of total] out of 6619136 bytes [ 99.1% used] at 0x00000008000b00000
total  : 17934560 [100.0% of total] out of 18153472 bytes [ 98.8% used]
```

```

[0.010s][info][class,load] opened: C:\Program Files\Java\jdk-11\lib\modules
[0.020s][info][class,load] java.lang.Object source: shared objects file
[0.020s][info][class,load] java.io.Serializable source: shared objects file
[0.020s][info][class,load] java.lang.Comparable source: shared objects file
[0.020s][info][class,load] java.lang.CharSequence source: shared objects file
[0.020s][info][class,load] java.lang.String source: shared objects file
[0.020s][info][class,load] java.lang.reflect.AnnotatedElement source: shared objects file
[0.020s][info][class,load] java.lang.reflect.GenericDeclaration source: shared objects file
[0.020s][info][class,load] java.lang.reflect.Type source: shared objects file
.
.
[0.082s][info][class,load] ConquerWorld source: file:/C:/Users/MalaGupta/AppDataSharing/code/
[0.082s][info][class,load] java.lang.NamedPackage source: shared objects file
[0.082s][info][class,load] java.lang.PublicMethods$MethodList source: shared objects file
[0.082s][info][class,load] java.lang.PublicMethods$Key source: shared objects file
[0.082s][info][class,load] java.lang.Void source: shared objects file
[0.083s][info][class,load] jdk.internal.misc.TerminatingThreadLocal$1 source: shared objects file
[0.083s][info][class,load] java.lang.Shutdown source: shared objects file
[0.083s][info][class,load] java.lang.Shutdown$Lock source: shared objects file

```

```

[0.010s][info][class,load] opened: C:\Program Files\Java\jdk-11\lib\modules
[0.019s][info][class,load] java.lang.Object source: jrt:/java.base
[0.019s][info][class,load] java.io.Serializable source: jrt:/java.base
[0.020s][info][class,load] java.lang.Comparable source: jrt:/java.base
[0.020s][info][class,load] java.lang.CharSequence source: jrt:/java.base
[0.020s][info][class,load] java.lang.String source: jrt:/java.base
[0.020s][info][class,load] java.lang.reflect.AnnotatedElement source: jrt:/java.base
[0.020s][info][class,load] java.lang.reflect.GenericDeclaration source: jrt:/java.base
[0.020s][info][class,load] java.lang.reflect.Type source: jrt:/java.base
.
.
[0.109s][info][class,load] ConquerWorld source: file:/C:/Users/MalaGupta/AppDataSharing/code/
[0.109s][info][class,load] java.lang.NamedPackage source: jrt:/java.base
[0.109s][info][class,load] java.lang.PublicMethods$MethodList source: jrt:/java.base
[0.109s][info][class,load] java.lang.PublicMethods$Key source: jrt:/java.base
[0.109s][info][class,load] java.lang.Void source: jrt:/java.base
[0.110s][info][class,load] jdk.internal.misc.TerminatingThreadLocal$1 source: jrt:/java.base
[0.110s][info][class,load] java.lang.Shutdown source: jrt:/java.base
[0.110s][info][class,load] java.lang.Shutdown$Lock source: jrt:/java.base

```

```

C:\Mala\code>java -Xshare:off -XX:+UseAppCDS -XX:DumpLoadedClassList=myclasses.lst -cp appcds.jar AppCDS
OpenJDK 64-Bit Server VM warning: Ignoring obsolete option UseAppCDS; AppCDS is automatically enabled

```

```

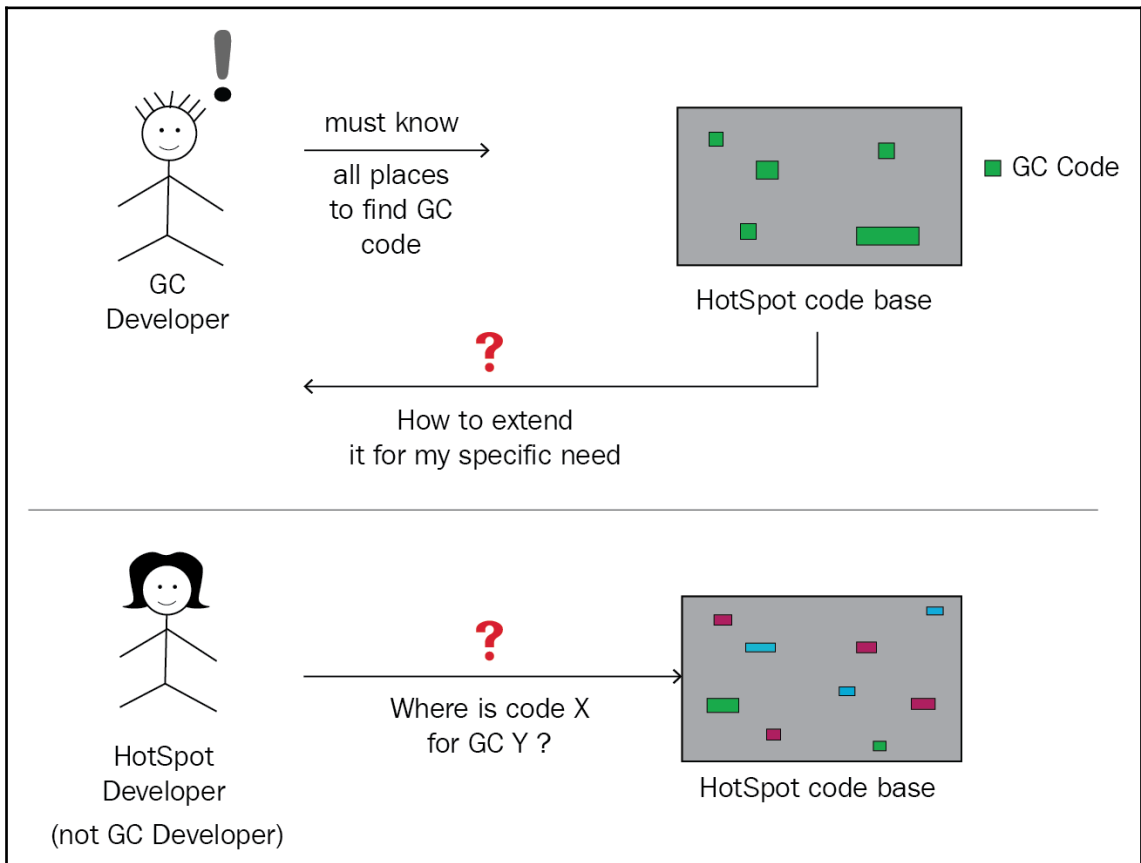
C:\Mala\code\Java11BookCode>jar cvf appcds.jar com/egovaguru/appcds/*.class
added manifest
adding: com/egovaguru/appcds/Cotton.class(in = 207) (out= 174)(deflated 15%)
adding: com/egovaguru/appcds/Plastic.class(in = 209) (out= 175)(deflated 16%)
adding: com/egovaguru/appcds/PlasticBottle.class(in = 233) (out= 171)(deflated 26%)
adding: com/egovaguru/appcds/Wood.class(in = 203) (out= 173)(deflated 14%)

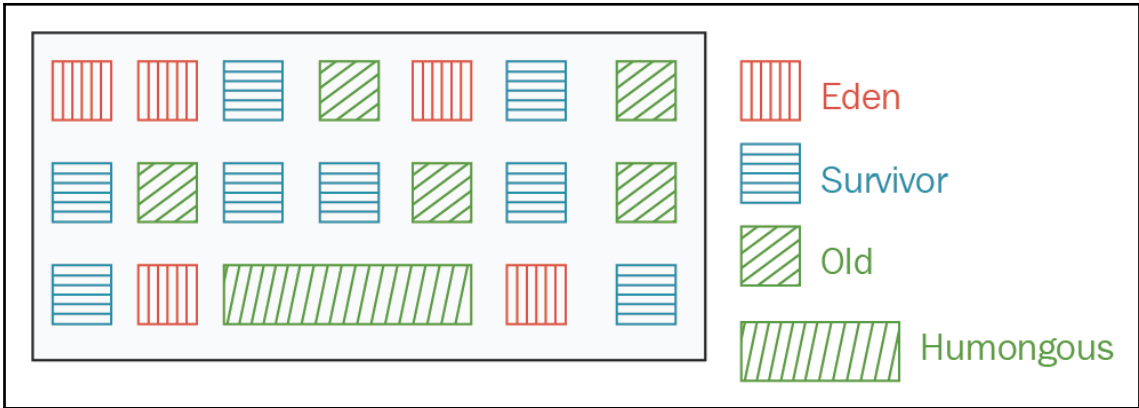
```

```
java/lang/Object
java/lang/String
java/io/Serializable
java/lang/Comparable
java/lang/CharSequence
java/lang/Class
java/lang/Cloneable
java/lang/ClassLoader
java/lang/System
java/lang/Throwable
java/lang/Error
.
.
.
AppCDS
com/ejavaguru/appcds/Plastic
jdk/internal/misc/TerminatingThreadLocal$1
java/lang/Shutdown
java/lang/Shutdown$Lock
```

```
C:\Mala\code\Java11BookCode>java -Xshare:dump -XX:SharedClassListFile=myappCDS.lst -XX:SharedArchiveFile=appCDS.jsa -cp
appcds.jar
narrow_class_base = 0x0000000800000000, narrow_class_shift = 3
Allocated temporary class space: 1073741824 bytes at 0x00000008c0000000
Allocated shared space: 3221225472 bytes at 0x0000000800000000
Loading classes to share ...
Preload Warning: Cannot find AppCDS
Loading classes to share: done.
Rewriting and linking classes ...
Rewriting and linking classes: done
Number of classes 683
  instance classes = 604
  obj array classes = 71
  type array classes = 8
Updating ConstMethods ... done.
Removing unshareable information ... done.
Scanning all metaspace objects ...
Allocating RW objects ...
Allocating RO objects ...
Relocating embedded pointers ...
Relocating external roots ...
Dumping symbol table ...
Relocating SystemDictionary::_well_known_klasses[] ...
Removing java_mirror ... done.
mc space: 5680 [ 0.1% of total] out of 65536 bytes [ 8.7% used] at 0x0000000800000000
rw space: 2092064 [ 22.5% of total] out of 2097152 bytes [ 99.8% used] at 0x0000000800010000
ro space: 3769640 [ 40.5% of total] out of 3801088 bytes [ 99.2% used] at 0x0000000800210000
md space: 2560 [ 0.0% of total] out of 65536 bytes [ 3.9% used] at 0x00000008005b0000
od space: 3233648 [ 34.7% of total] out of 3276800 bytes [ 98.7% used] at 0x00000008005c0000
total : 9103592 [100.0% of total] out of 9306112 bytes [ 97.8% used]
```


Chapter 3: Garbage Collector Optimizations





```
[0.012s][info][gc,heap] Heap region size: 1M ——— ①
[0.018s][info][gc      ] Using G1 ——— ②
[0.018s][info][gc,heap,coops] Heap address: 0x0000000741c00000, size: 3044 MB,
Compressed Oops mode: Zero based, Oop shift amount: 3
[0.309s][info][gc,start  ] GC(0) Pause Young (Normal) (G1 Evacuation Pause) ③
[0.309s][info][gc,task   ] GC(0) Using 4 workers of 4 for evacuation ④
[0.311s][info][gc,phases ] GC(0) Pre Evacuate Collection Set: 0.0ms
[0.312s][info][gc,phases ] GC(0) Evacuate Collection Set: 1.5ms
[0.312s][info][gc,phases ] GC(0) Post Evacuate Collection Set: 0.2ms
[0.312s][info][gc,phases ] GC(0) Other: 1.1ms
[0.313s][info][gc,heap   ] GC(0) Eden regions: 14->0(113) ⑤
[0.330s][info][gc,heap   ] GC(0) Survivor regions: 0->2(2)
[0.332s][info][gc,heap   ] GC(0) Old regions: 0->0
[0.333s][info][gc,heap   ] GC(0) Humongous regions: 43->43
[0.334s][info][gc,metaspace] GC(0) Metaspace: 2066K->2066K(1056768K)
```

```
[1.264s][info][gc,start  ] GC(5) Pause Young (Concurrent Start) (G1 Humongous Allocation)
[1.264s][info][gc,task   ] GC(5) Using 4 workers of 4 for evacuation
[1.267s][info][gc,phases ] GC(5) Pre Evacuate Collection Set: 0.0ms
[1.267s][info][gc,phases ] GC(5) Evacuate Collection Set: 1.8ms
[1.268s][info][gc,phases ] GC(5) Post Evacuate Collection Set: 0.2ms
[1.268s][info][gc,phases ] GC(5) Other: 1.0ms
[1.269s][info][gc,heap   ] GC(5) Eden regions: 36->0(20)
[1.269s][info][gc,heap   ] GC(5) Survivor regions: 2->2(7)
[1.270s][info][gc,heap   ] GC(5) Old regions: 0->0
[1.270s][info][gc,heap   ] GC(5) Humongous regions: 148->148
[1.271s][info][gc,metaspace] GC(5) Metaspace: 2073K->2073K(1056768K)
[1.271s][info][gc      ] GC(5) Pause Young (Concurrent Start) (G1 Humongous Allocation)
184M->149M(192M) 7.197ms
[1.271s][info][gc,cpu    ] GC(5) User=0.02s Sys=0.00s Real=0.01s
```

```

[20.498s][info][gc,task      ] GC(196) Using 4 workers of 4 for full compaction ①
[20.498s][info][gc,start      ] GC(196) Pause Full (G1 Humongous Allocation) ②
[20.499s][info][gc,phases,start] GC(196) Phase 1: Mark live objects
[20.502s][info][gc,stringtable ] GC(196) Cleaned string and symbol table, strings: 2609 processed, 23 removed,
symbols: 5461
[20.502s][info][gc,phases      ] GC(196) Phase 1: Mark live objects 3.246ms
[20.502s][info][gc,phases,start] GC(196) Phase 2: Prepare for compaction
[20.504s][info][gc,phases      ] GC(196) Phase 2: Prepare for compaction 1.450ms
[20.504s][info][gc,phases,start] GC(196) Phase 3: Adjust pointers
[20.509s][info][gc,phases      ] GC(196) Phase 3: Adjust pointers 4.849ms
[20.513s][info][gc,phases,start] GC(196) Phase 4: Compact heap
[20.515s][info][gc,phases      ] GC(196) Phase 4: Compact heap 1.795ms
[20.517s][info][gc,heap        ] GC(196) Eden regions: 0->0(152)
[20.517s][info][gc,heap        ] GC(196) Survivor regions: 0->0(19)
[20.518s][info][gc,heap        ] GC(196) Old regions: 3->3
[20.519s][info][gc,heap        ] GC(196) Humongous regions: 2983->2983
[20.519s][info][gc,metaspace   ] GC(196) Metaspace: 2114K->2022K(1056768K)
[20.519s][info][gc             ] GC(196) Pause Full (G1 Humongous Allocation) 2984M->2983M(3044M) 20.705ms ⑤
[20.519s][info][gc,cpu        ] GC(196) User=0.02s Sys=0.02s Real=0.02s
[20.520s][info][gc,marking     ] GC(193) Concurrent Mark From Roots 61.566ms
[20.520s][info][gc,marking     ] GC(193) Concurrent Mark Abort
[20.520s][info][gc             ] GC(193) Concurrent Cycle 63.136ms
[20.523s][info][gc,heap,exit  ] Heap
[20.523s][info][gc,heap,exit  ] garbage-first heap total 3117056K, used 3055583K [0x0000000741c00000,
0x0000000800000000)
[20.524s][info][gc,heap,exit  ] region size 1024K, 1 young (1024K), 0 survivors (0K)
[20.524s][info][gc,heap,exit  ] Metaspace used 2037K, capacity 4544K, committed 4864K, reserved 1056768K
[20.524s][info][gc,heap,exit  ] class space used 185K, capacity 408K, committed 512K, reserved 1048576K

```

Chapter 4: Miscellaneous Improvements in JDK 10

No images.

Chapter 5: Local Variable Syntax for Lambda Parameters

No images.

Chapter 6: Epsilon GC

```
C:\Users\Mala Gupta\code>java -XX:+UnlockExperimentalVMOptions -XX:+UseEpsilonGC
HelloEpsilonGC
[0.014s][info][gc] Resizeable heap; starting at 190M, max: 3043M, step: 128M
[0.015s][info][gc] Using TLAB allocation; max: 4096K
[0.015s][info][gc] Elastic TLABs enabled; elasticity: 1.10x
[0.016s][info][gc] Elastic TLABs decay enabled; decay time: 1000ms
[0.016s][info][gc] Using Epsilon
[0.018s][info][gc,heap,coops] Heap address: 0x0000000741cb0000, size: 3043 MB,
Compressed Oops mode: Zero based, Oop shift amount: 3
Hello to Epsilon GC!
[0.076s][info][gc,heap,exit ] Heap
[0.076s][info][gc,heap,exit ] Epsilon Heap
[0.076s][info][gc,heap,exit ] Allocation space:
[0.077s][info][gc,heap,exit ] space 194816K, 0% used [0x0000000741cb0000,
0x0000000741d762e0, 0x000000074daf0000)
[0.078s][info][gc ] Total allocated: 792 KB
[0.079s][info][gc ] Average allocation rate: 3809505 KB/sec
```

```
[0.009s][info][gc] Non-resizeable heap; start/max: 40M
[0.010s][info][gc] Using TLAB allocation; max: 4096K
[0.011s][info][gc] Elastic TLABs enabled; elasticity: 1.10x
[0.011s][info][gc] Elastic TLABs decay enabled; decay time: 1000ms
[0.011s][info][gc] Using Epsilon
[0.011s][info][gc,heap,coops] Heap address: 0x0000000fd800000, size: 40 MB, Compressed Oops mode: 32-bit
[0.065s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 2M (5.08%) used
[0.071s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 4M (10.92%) used
[0.080s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 6M (17.05%) used
[0.083s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 8M (22.05%) used
[0.086s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 10M (27.05%) used
[0.088s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 12M (32.05%) used
[0.093s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 14M (37.05%) used
[0.095s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 16M (42.05%) used
[0.100s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 18M (47.05%) used
[0.103s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 20M (52.05%) used
[0.105s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 22M (57.05%) used
[0.109s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 24M (62.05%) used
[0.113s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 26M (67.05%) used
[0.114s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 28M (72.05%) used
[0.116s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 30M (77.05%) used
[0.118s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 32M (82.05%) used
[0.120s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 34M (87.05%) used
[0.123s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 36M (92.05%) used
[0.125s][info][gc ] Heap: 40M reserved, 40M (100.00%) committed, 38M (97.05%) used
Terminating due to java.lang.OutOfMemoryError: Java heap space
```

```

[0.011s][info][gc] Non-resizeable heap; start/max: 40M
[0.012s][info][gc] Using TLAB allocation; max: 4096K
[0.013s][info][gc] Elastic TLABs enabled; elasticity: 1.10x
[0.013s][info][gc] Elastic TLABs decay enabled; decay time: 1000ms
[0.013s][info][gc] Using Epsilon
[0.013s][info][gc,heap,coops] Heap address: 0x0000000fd800000, size: 40 MB, Compressed Oops mode: 32-bit
[0.068s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 2M (5.08%) used
[0.078s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 4M (10.92%) used
[0.083s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 6M (17.05%) used
[0.090s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 8M (22.05%) used
[0.096s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 10M (27.05%) used
[0.100s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 12M (32.05%) used
[0.104s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 14M (37.05%) used
[0.105s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 16M (42.05%) used
[0.107s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 18M (47.05%) used
[0.108s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 20M (52.05%) used
[0.110s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 22M (57.05%) used
[0.112s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 24M (62.05%) used
[0.114s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 26M (67.05%) used
[0.116s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 28M (72.05%) used
[0.118s][info][gc] ] Heap: 40M reserved, 40M (100.00%) committed, 30M (77.05%) used
[0.119s][info][gc,heap,exit] ] Heap
[0.119s][info][gc,heap,exit] ] Epsilon Heap
[0.120s][info][gc,heap,exit] ] Allocation space:
[0.120s][info][gc,heap,exit] ] space 40960K, 78% used [0x0000000fd800000, 0x00000000ff752018, 0x00000001
[0.120s][info][gc] ] Total allocated: 32072 KB
[0.120s][info][gc] ] Average allocation rate: 100510197 KB/sec

```

Chapter 7: The HTTP Client API

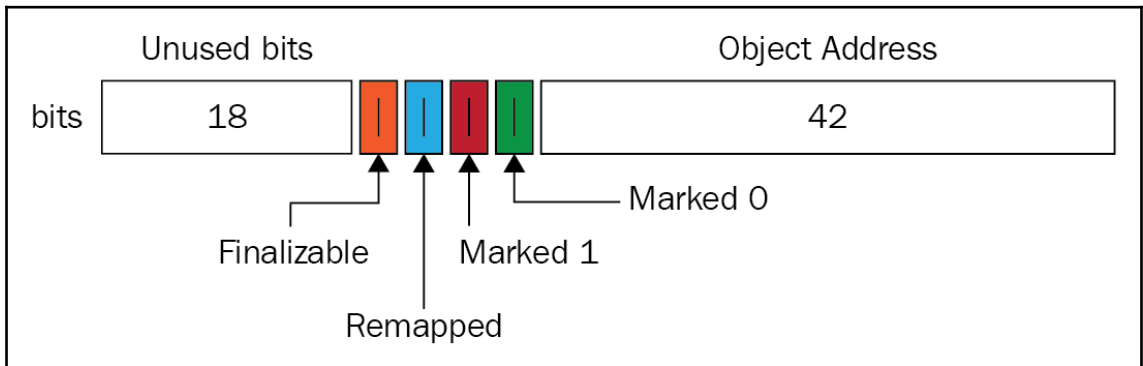
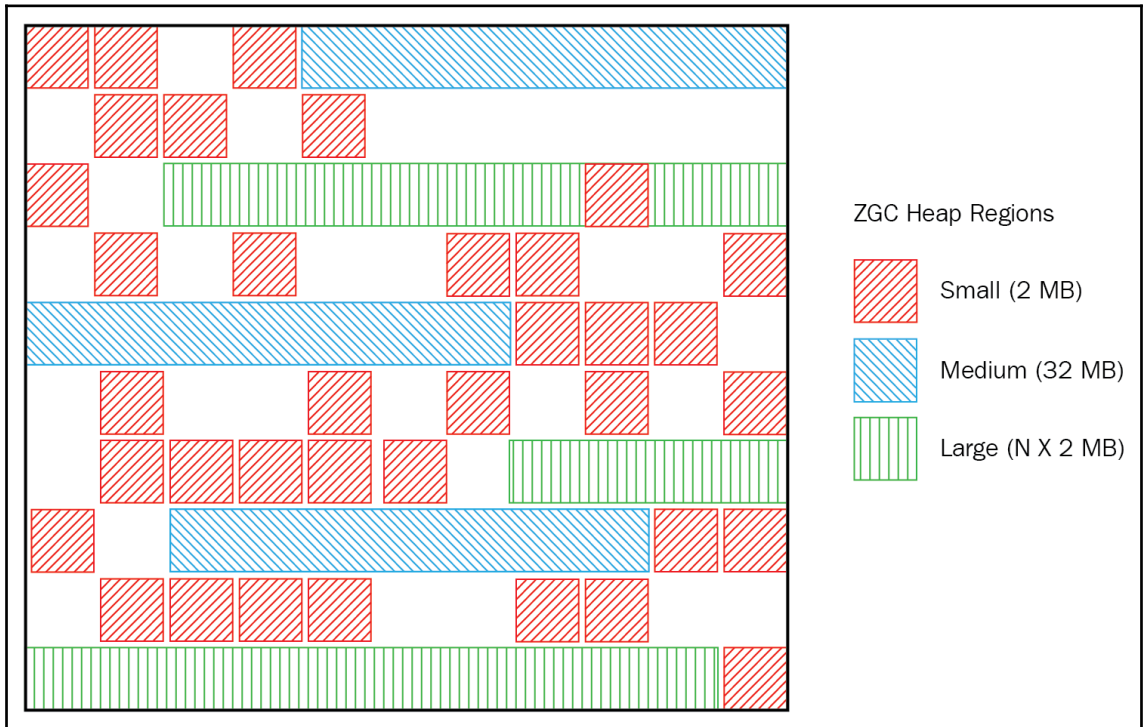
The screenshot shows the Java API documentation for the `Httpclient` class. The left sidebar contains navigation tabs: OVERVIEW, MODULE, PACKAGE, CLASS (highlighted), USE, TREE, DEPRECATED, INDEX, HELP. Below these are sections for ALL CLASSES, SUMMARY, and NESTED. The main content area is titled "Class Httpclient" and includes the following information:

- Module: [java.net.http](#)
- Package: [java.net.http](#)
- Class: **Httpclient**
- Superclasses: `java.lang.Object`, `java.net.http.Httpclient`
- Code snippet:

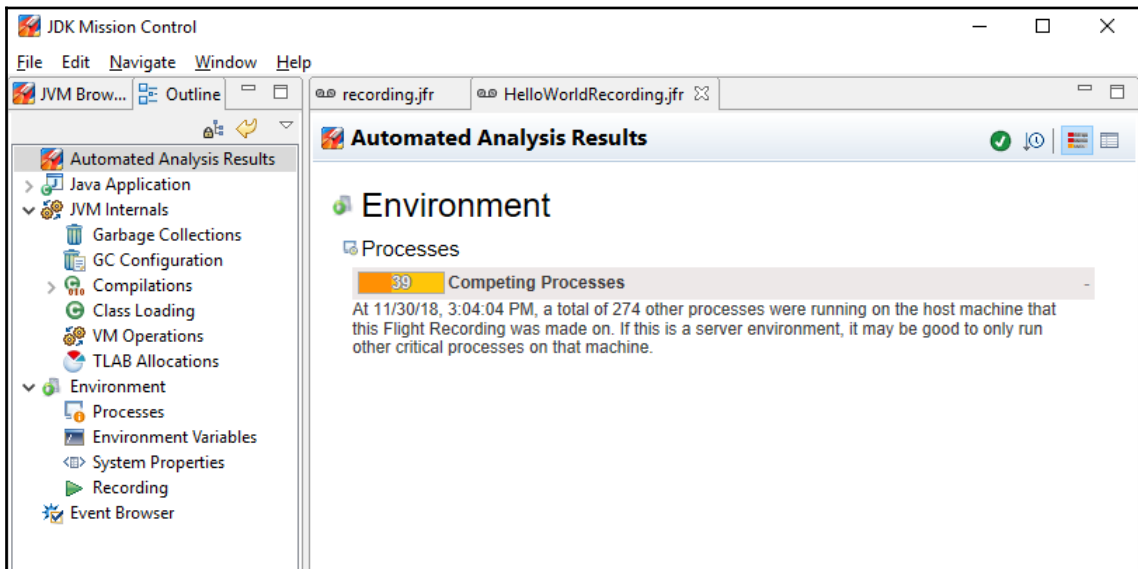
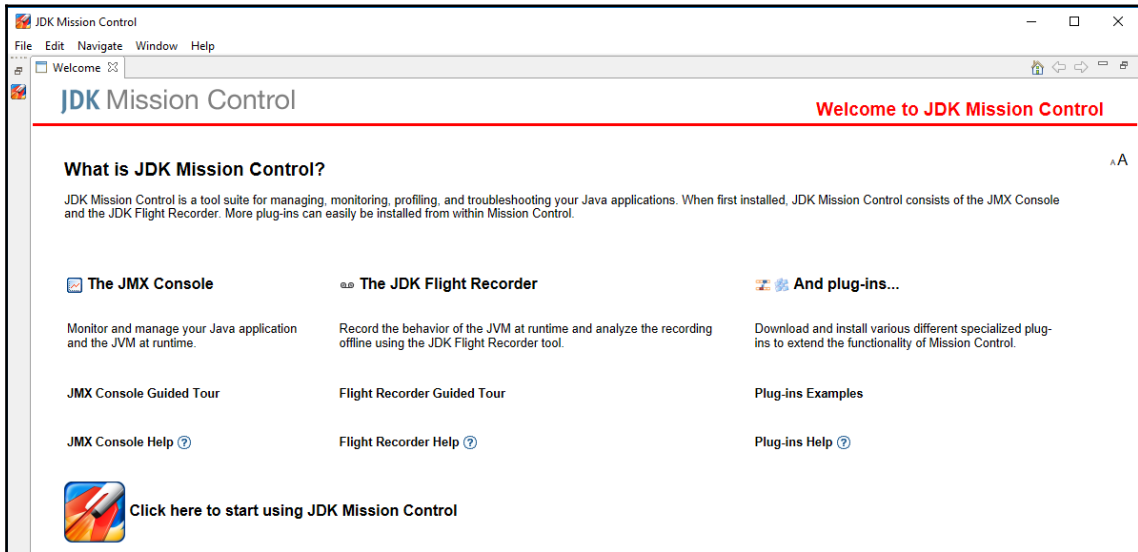
```
public abstract class Httpclient extends Object
```
- Description: "An HTTP Client." and "An `Httpclient` can be used to send requests and retrieve their responses. An `Httpclient` is created in either synchronous or asynchronous mode (HTTP/1.1 or HTTP/2), whether to follow redirects, a proxy, an authenticator, etc. Once created, an `Httpclient` provides configuration information, and resource sharing, for all requests sent through it. A `BodyHandler` must be supplied for each `HttpRequest` sent. The `BodyHandler` determines how to handle the response body (typically) are available. Whether the response body bytes have been read or not depends on the mode of the client. Requests can be sent either synchronously or asynchronously." followed by a list of methods:
 - `send(HttpRequest, BodyHandler)` blocks until the request has been sent and the response is received.
 - `sendAsync(HttpRequest, BodyHandler)` sends the request and receives the response asynchronously. `CompletableFuture` completes when the response becomes available. The returned `CompletableFuture` completes when the response becomes available. The returned `CompletableFuture` completes when the response becomes available. The returned `CompletableFuture` completes when the response becomes available.
- Class Hierarchy:
 - `java.lang.Object`
 - `java.net.http.Httpclient`
- Code snippet:

```
public abstract class Httpclient extends Object
```
- Description: "An HTTP Client." and "An `Httpclient` can be used to send requests and retrieve their responses. An `Httpclient` is created in either synchronous or asynchronous mode (HTTP/1.1 or HTTP/2), whether to follow redirects, a proxy, an authenticator, etc. Once created, an `Httpclient` provides configuration information, and resource sharing, for all requests sent through it. A `BodyHandler` must be supplied for each `HttpRequest` sent. The `BodyHandler` determines how to handle the response body (typically) are available. Whether the response body bytes have been read or not depends on the mode of the client. Requests can be sent either synchronously or asynchronously." followed by a list of methods:
 - `send(HttpRequest, BodyHandler)` blocks until the request has been sent and the response is received.
 - `sendAsync(HttpRequest, BodyHandler)` sends the request and receives the response asynchronously. `CompletableFuture` completes when the response becomes available. The returned `CompletableFuture` completes when the response becomes available. The returned `CompletableFuture` completes when the response becomes available. The returned `CompletableFuture` completes when the response becomes available.

Chapter 8: ZGC



Chapter 9: Flight Recorder and Mission Control



The screenshot shows the Eclipse IDE interface with the 'Automated Analysis Results' window open. The left sidebar contains a tree view of analysis categories:

- Automated Analysis Results
 - Java Application
 - Threads
 - Memory
 - Lock Instances
 - File I/O
 - Socket I/O
 - Method Profiling
 - Exceptions
 - Thread Dumps
 - JVM Internals
 - Garbage Collections
 - GC Configuration
 - Compilations
 - Class Loading
 - VM Operations
 - TLAB Allocations
 - Environment
 - Processes
 - Environment Variables
 - System Properties
 - Recording
 - Event Browser

The main pane displays 'Automated Analysis Results' for 'Epsilon.jfr'. The results are organized into sections:

- Java Application**
 - Lock Instances**
 - 100 Context Switches** - The program context switches a lot and many threads wait on the same monitor. Consider using fewer threads, or try to decrease lock contention by other means.
 - 100 Java Blocking** - Threads in the application were blocked on locks for a total time of 2 h 40 min. The most common monitor class was 'PrintStream', which was blocked on 18,822 times. The following regular expression was used to exclude threads from this rule: `{.*weblogic.socket.Muxer.*}`
 - JVM Internals**
 - 50 Discouraged VM Options** - Due to experimental VM options not being fully supported and may thus be unreliable they should not be used in a production environment. Unless you have to use an experimental option, you should avoid the '-XX:+UnlockExperimentalVMOptions' command line option.
- Environment**
 - Processes**
 - 39 Competing Processes** - At 12/1/18, 6:00:17 AM, a total of 271 other processes were running on the host machine that this Flight Recording was made on. If this is a server environment, it may be good to only run other critical processes on that machine.

JDK Mission Control

File Edit Navigate Window Help

JVM Brow... Outline

Epsilon.jfr

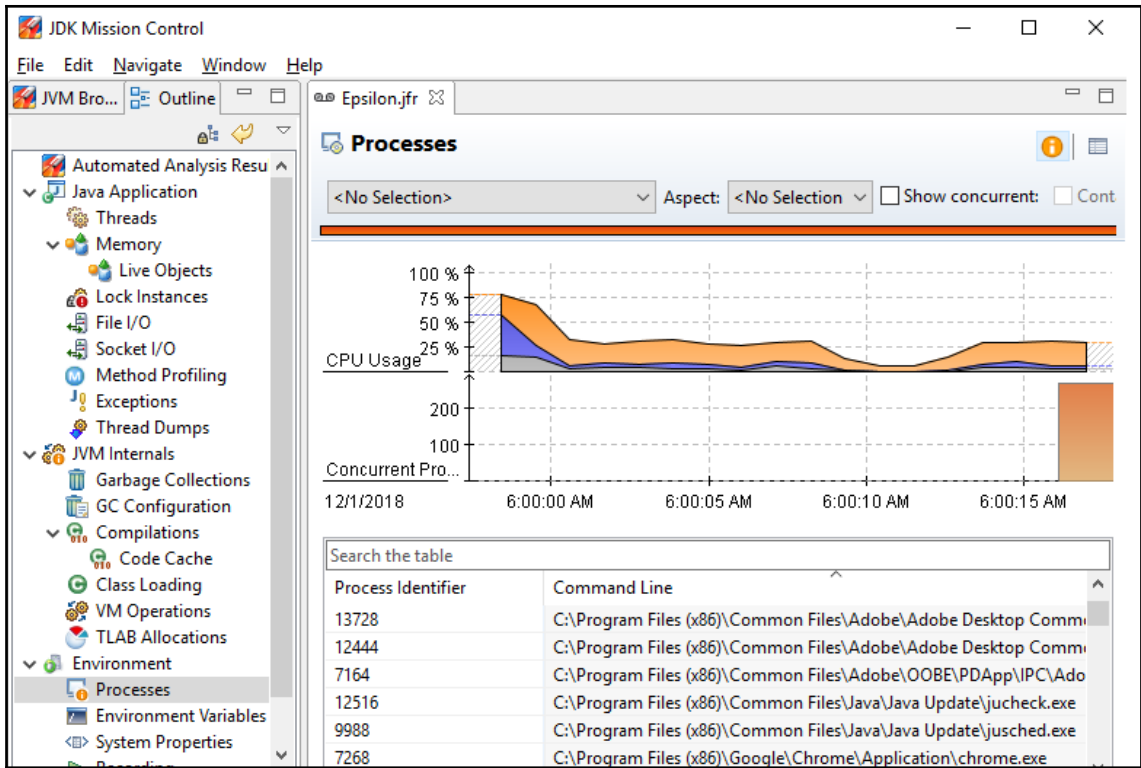
Lock Instances

<No Selection> Aspect: <No Selection> Show concurrent: Contained

Monitor Class	Total Blocked Time	Distinct T...	Count
java.io.PrintStream	2 h 40 min	500	18,822
java.lang.Object	372.182 ms	14	14

Monitor Address	Total Blocked Time	Distinct T...	Count
0x271BBC05D88	2 h 40 min	500	18,822
0x271BBB67C08	191.115 ms	7	7
0x271BBB65D08	181.067 ms	7	7

Thread	Total Blocked Time	Count
Thread-19	19.941 s	48
Thread-2	19.940 s	50
Thread-9	19.939 s	50
Thread-18	19.938 s	50
Thread-12	19.936 s	50

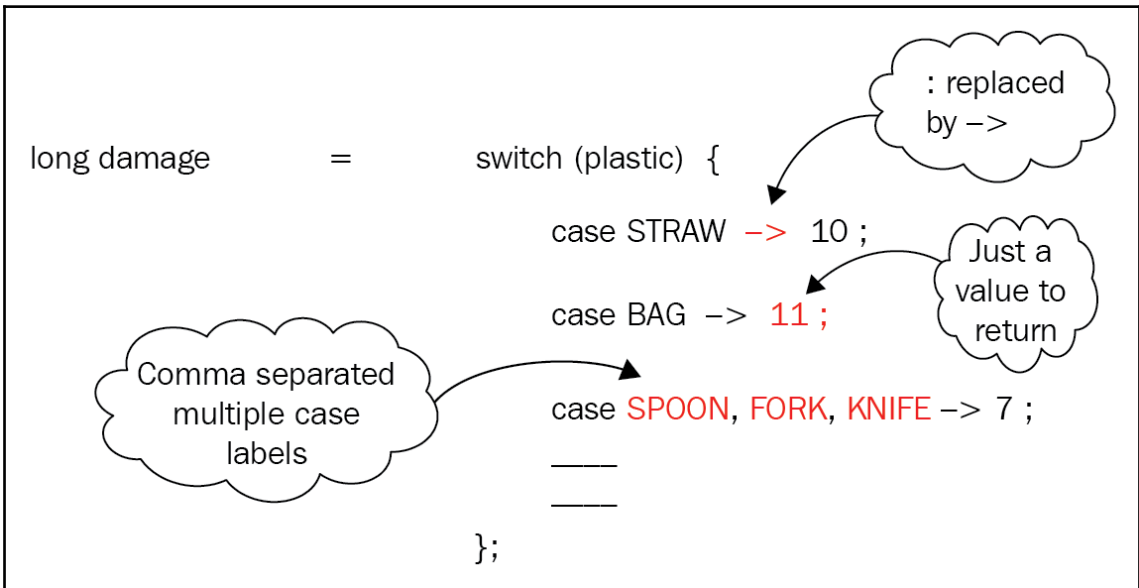


Chapter 10: Miscellaneous Improvements in JDK 11

No images.

Chapter 11: Switch Expressions

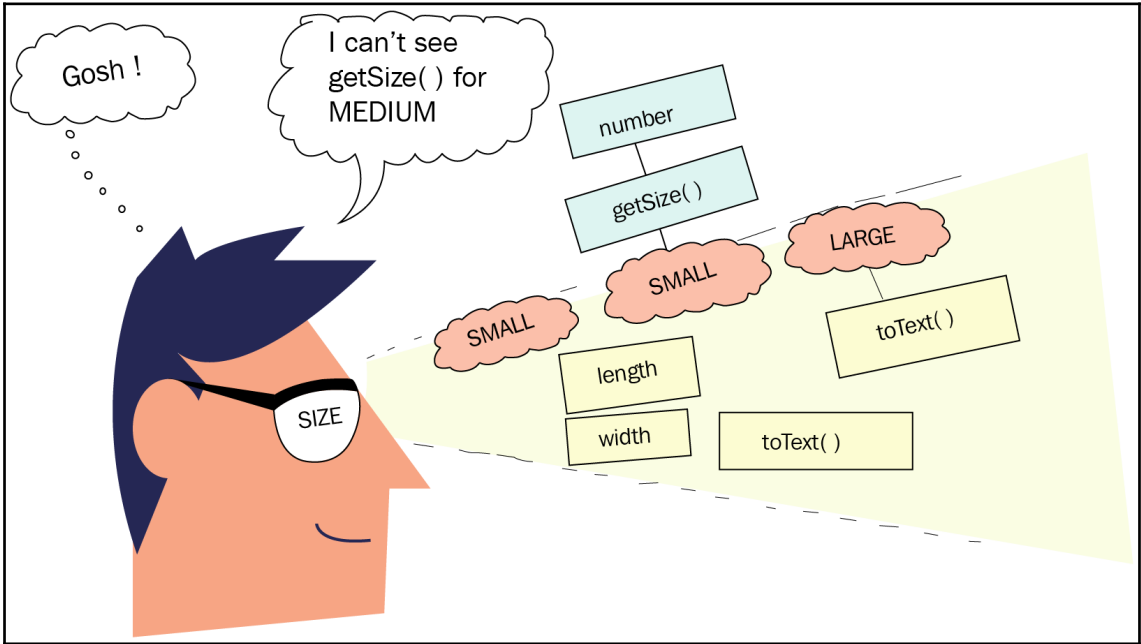
What I want 😊	What I get 😞
<ul style="list-style-type: none">① examine variable size② initialize variable length accordingly③ Output value of variable length	<pre>switch (size){ case xs: length=20; System.out.println (length); break; case s: _____ _____ _____</pre>

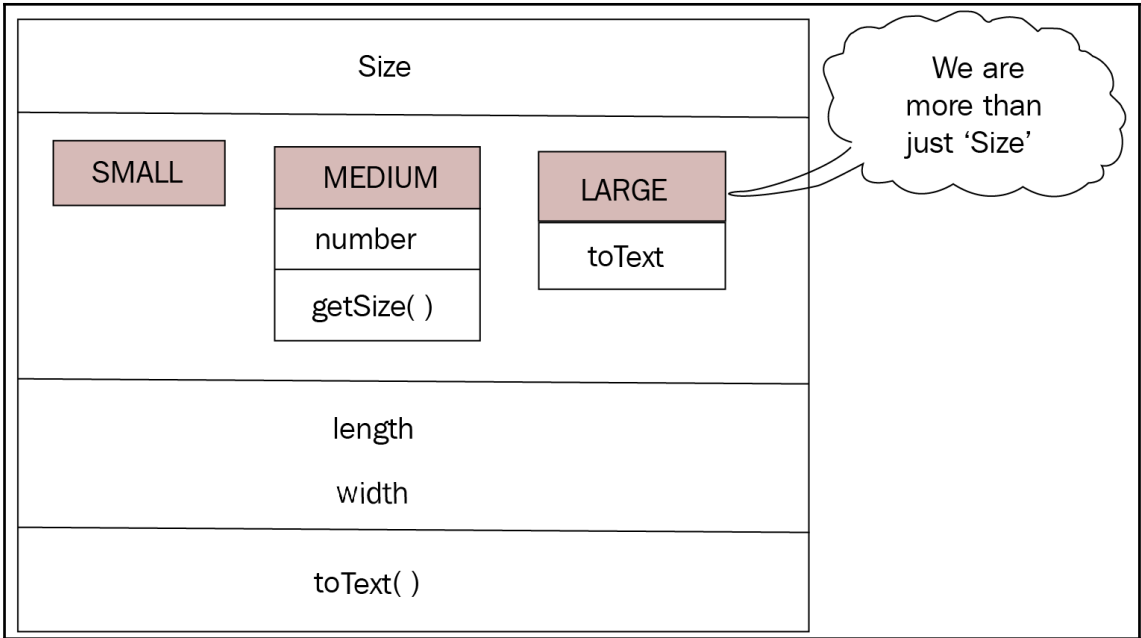


Chapter 12: Miscellaneous Improvements in JDK 12

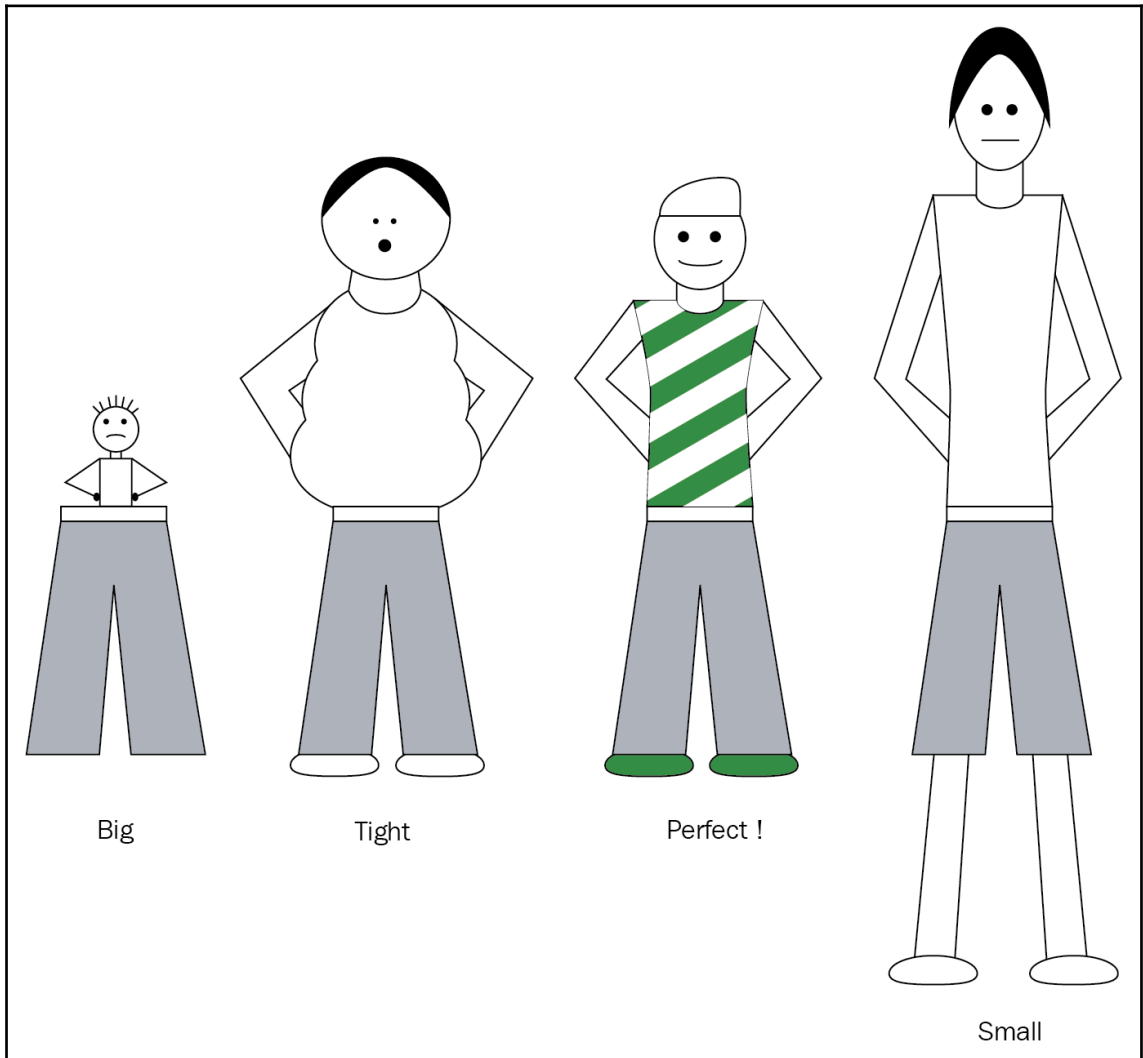
No images.

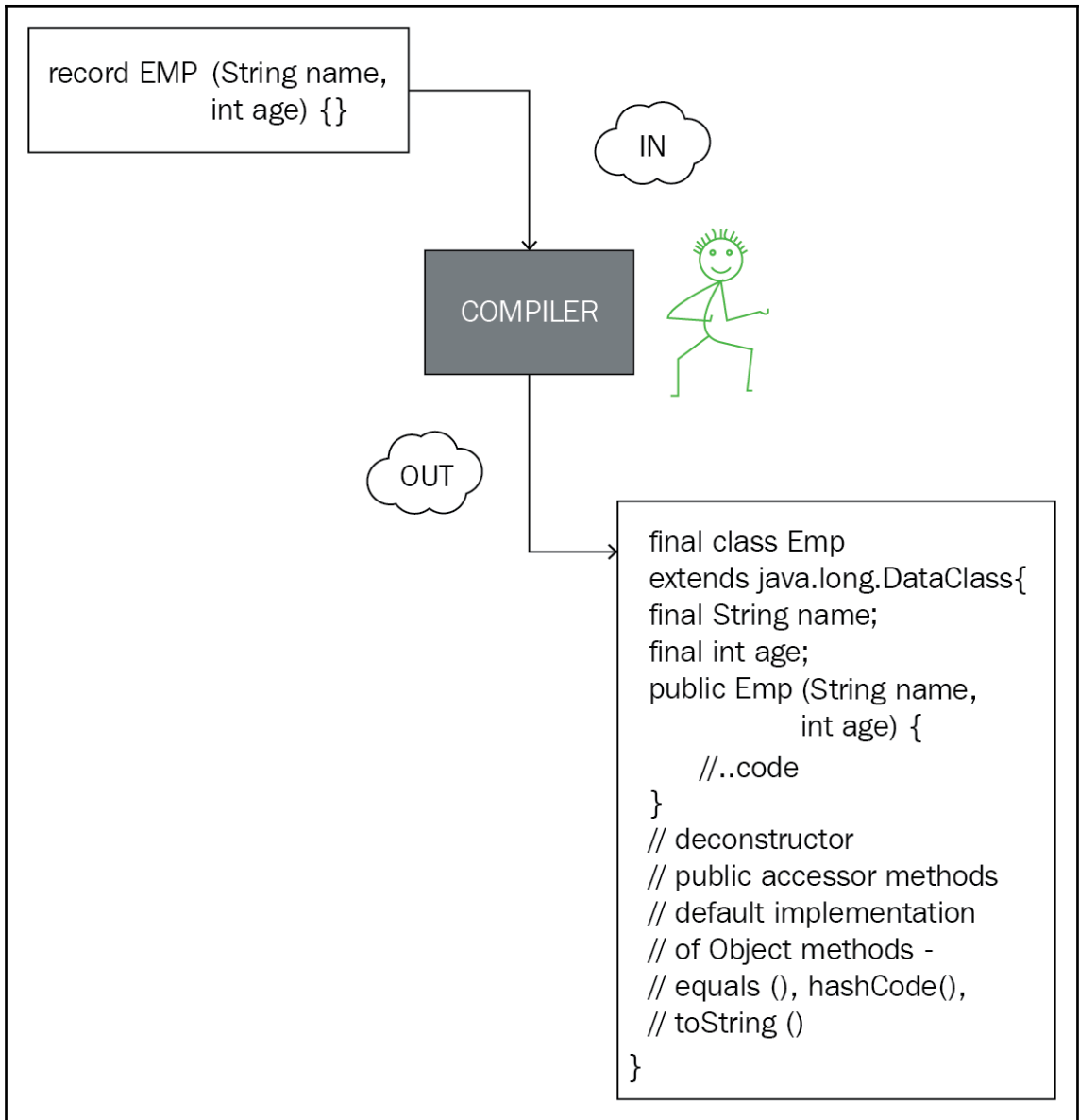
Chapter 13: Enhanced Enums in Project Amber

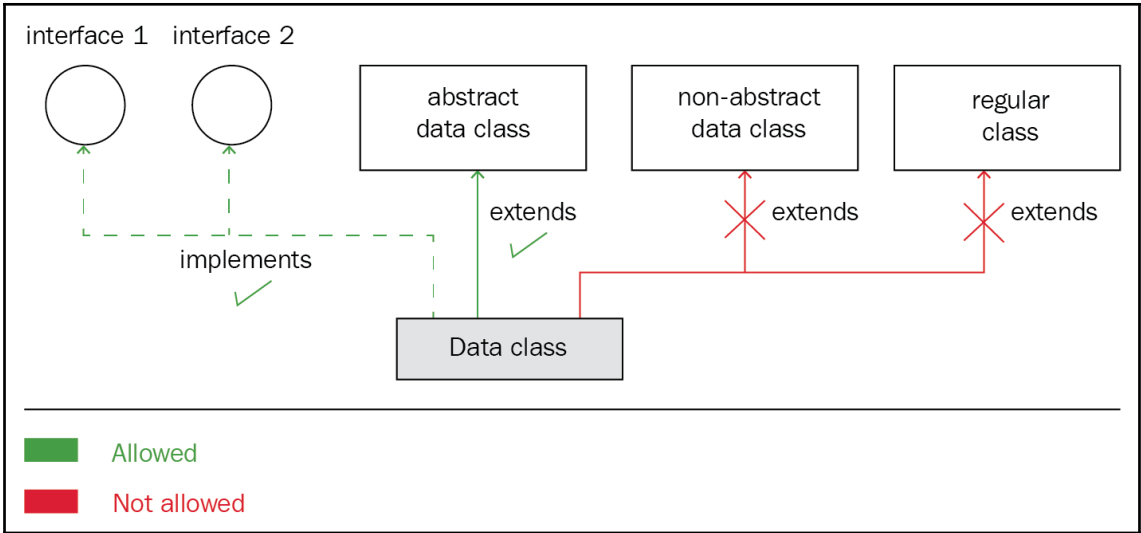




Chapter 14: Data Classes and Their Usage







Chapter 15: Raw String Literals

No images.

Chapter 16: Lambda Leftovers

No images.

Chapter 17: Pattern Matching

obj instanceof Ocean	obj matches Ocean o
((Ocean)obj).getBottles();	o.getBottles();