

Cohort Laptop



JPMC Tech Start

## Repositories

Filter by:

All

Public

### Repository



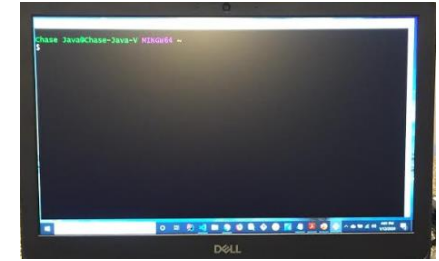
java-main



victoria-hinkle-moore

fork

Cohort Repo: first-lastname



Instructor Laptop



java-main

push origin master

pull upstream master

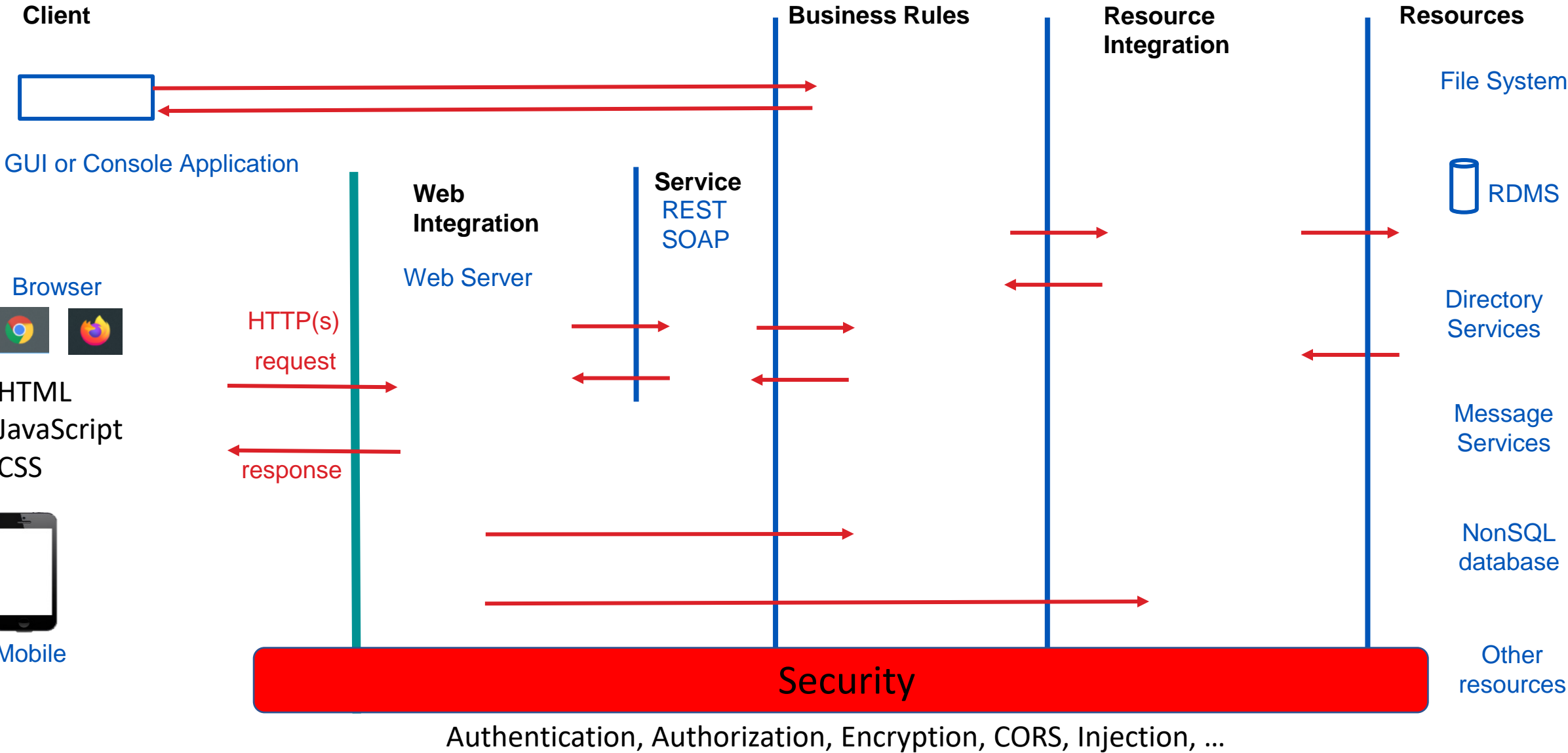
push origin master

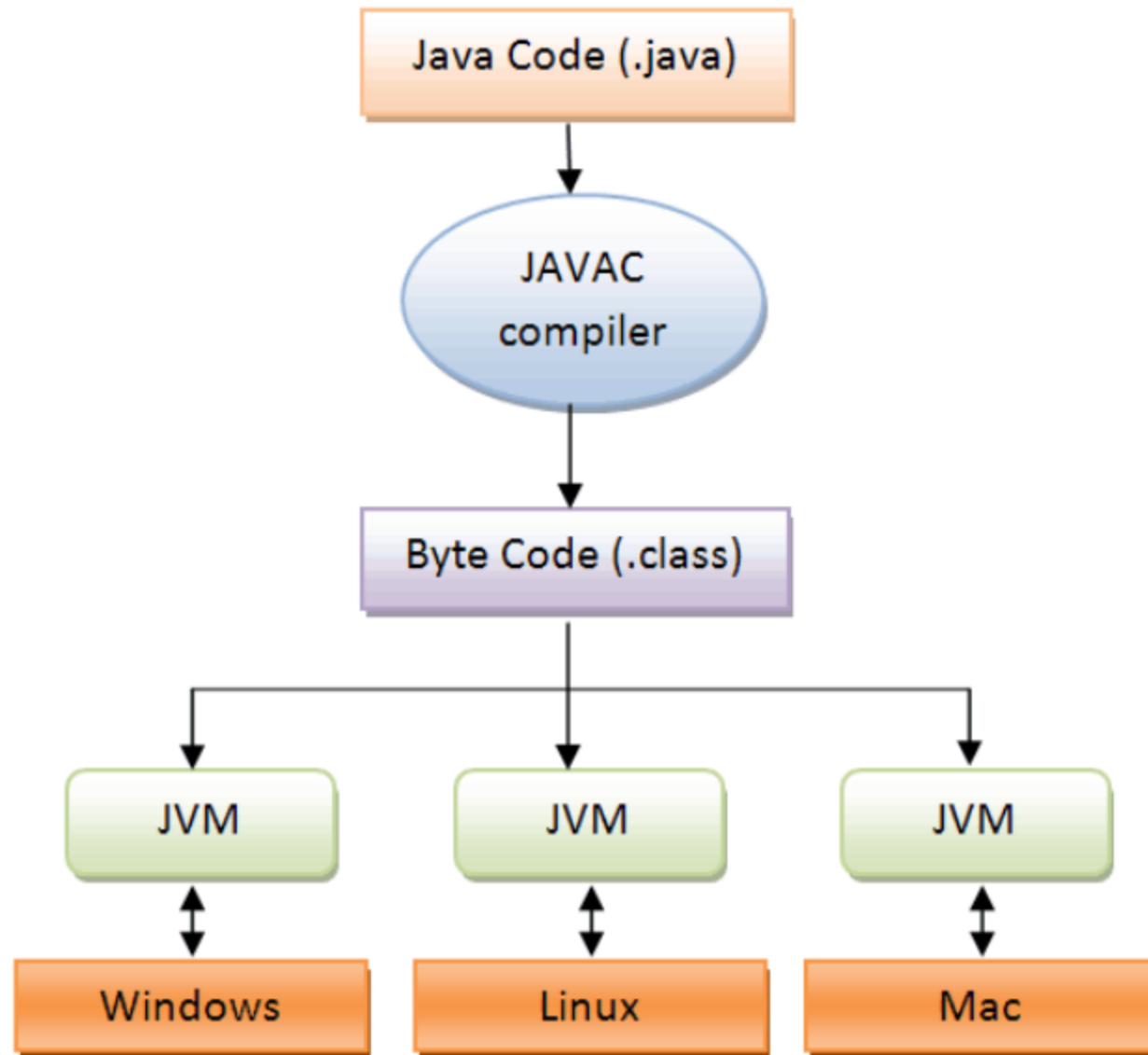
- Submit your exercise work

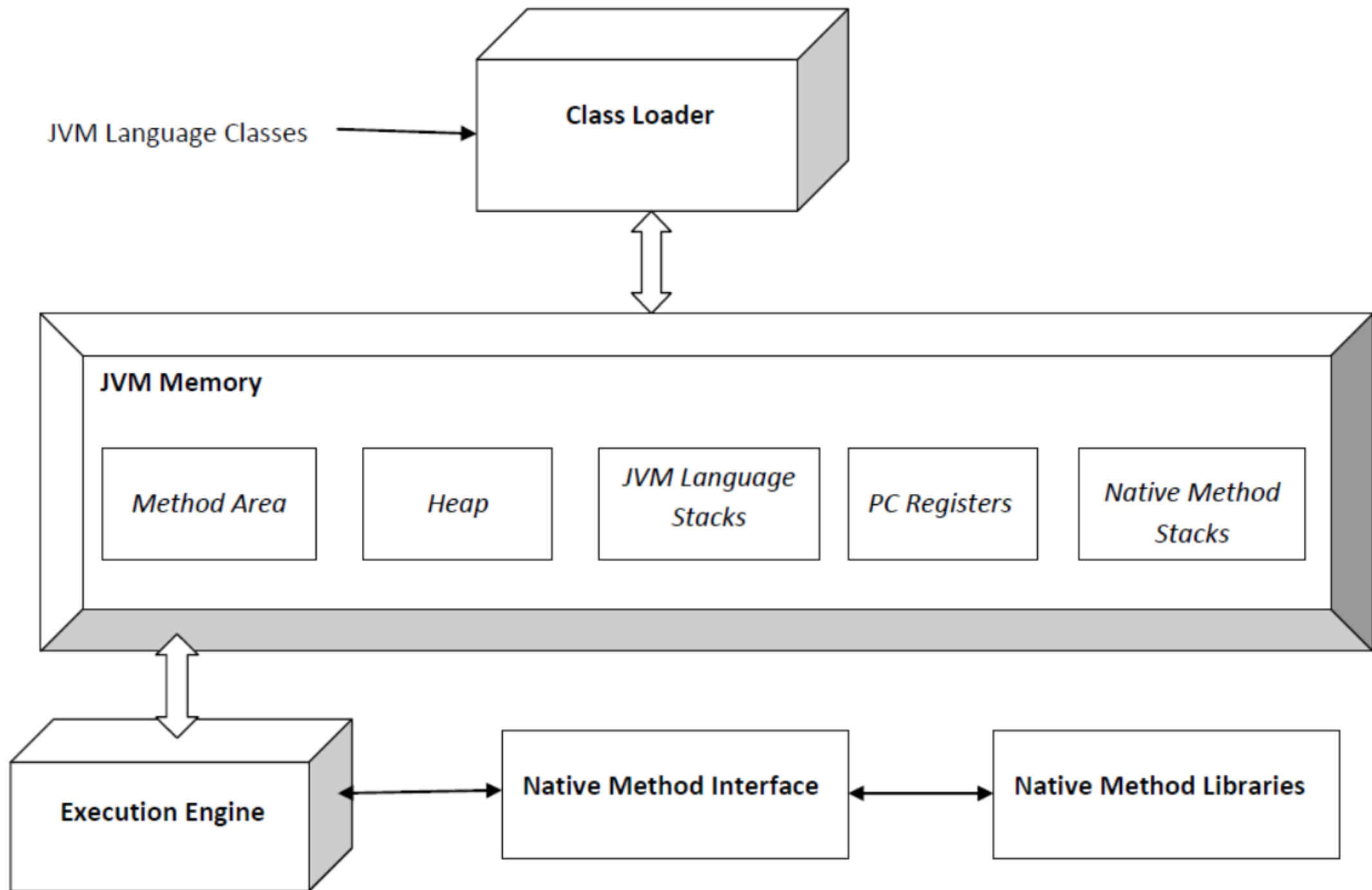
# Daily Git Commands

- Execute the git commands from the directory top of your repository
  - Current directory should be your-name folder inside your workspace folder
- Git Commands to **Pull** Daily Work From BitBucket
  - `git pull upstream master`
- Git Commands to **Push** Your Work To BitBucket
  - `git add -A`
  - `git commit -m "with message"`
  - `git push origin master`

# Application Architecture







TYPE	DESCRIPTION	DEFAULT	SIZE	EXAMPLE LITERALS	RANGE OF VALUES
boolean	true or false	false	1 bit	true, false	true, false
byte	twos complement integer	0	8 bits	(none)	-128 to 127
char	unicode character	\u0000	16 bits	'a', '\u0041', '\101', '\\', '\', '\n', 'β'	character representation of ASCII values 0 to 255
short	twos complement integer	0	16 bits	(none)	-32,768 to 32,767
int	twos complement integer	0	32 bits	-2, -1, 0, 1, 2	-2,147,483,648 to 2,147,483,647
long	twos complement integer	0	64 bits	-2L, -1L, 0L, 1L, 2L	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
float	IEEE 754 floating point	0.0	32 bits	1.23e100f, -1.23e-100f, .3f, 3.14F	upto 7 decimal digits
double	IEEE 754 floating point	0.0	64 bits	1.23456e300d, -1.23456e-300d, 1e1d	upto 16 decimal digits

### Operator Precedence

Operators	Precedence
postfix	<i>expr++ expr--</i>
unary	<i>++expr --expr +expr -expr ~ !</i>
multiplicative	<i>* / %</i>
additive	<i>+ -</i>
shift	<i>&lt;&lt; &gt;&gt; &gt;&gt;&gt;</i>
relational	<i>&lt; &gt; &lt;= &gt;= instanceof</i>
equality	<i>== !=</i>
bitwise AND	<i>&amp;</i>
bitwise exclusive OR	<i>^</i>
bitwise inclusive OR	<i> </i>
logical AND	<i>&amp;&amp;</i>
logical OR	<i>  </i>
ternary	<i>? :</i>
assignment	<i>= += -= *= /= %= &amp;= ^=  = &lt;&lt;= &gt;&gt;= &gt;&gt;&gt;=</i>

Precedence	Operator	Type	Associativity
15	() [] .	Parentheses Array subscript Member selection	Left to Right
14	++ --	Unary post-increment Unary post-decrement	Right to left
13	++ -- + - ! ~ ( type )	Unary pre-increment Unary pre-decrement Unary plus Unary minus Unary logical negation Unary bitwise complement Unary type cast	Right to left
12	* / %	Multiplication Division Modulus	Left to right
11	+ -	Addition Subtraction	Left to right
10	<< >> >>>	Bitwise left shift Bitwise right shift with sign extension Bitwise right shift with zero extension	Left to right
9	< <= > >= instanceof	Relational less than Relational less than or equal Relational greater than Relational greater than or equal Type comparison (objects only)	Left to right
8	== !=	Relational is equal to Relational is not equal to	Left to right
7	&	Bitwise AND	Left to right
6	^	Bitwise exclusive OR	Left to right
5		Bitwise inclusive OR	Left to right
4	&&	Logical AND	Left to right
3		Logical OR	Left to right
2	? :	Ternary conditional	Right to left
1	= += -= *= /= % =	Assignment Addition assignment Subtraction assignment Multiplication assignment Division assignment Modulus assignment	Right to left



## Logical operator table

The following table sums up the different logical operators:

A	B	!A	A && B	A    B	A ^ B
TRUE	TRUE	FALSE	TRUE	TRUE	FALSE
TRUE	FALSE	FALSE	FALSE	TRUE	TRUE
FALSE	TRUE	TRUE	FALSE	TRUE	TRUE
FALSE	FALSE	TRUE	FALSE	FALSE	FALSE

return datatype

- primitive
- reference (Classname, ...)
- void

## Anatomy of a Method

method name

```
public int returnNotOne(int number, Customer customer)
```

```
{
```

```
// statements – end with a semi colon
```

```
// blocks – conditional and looping
```

```
return number + 1;
```

```
}
```

block for concrete method

method arguments

datatype argName

modifiers (others include static and final)

scope

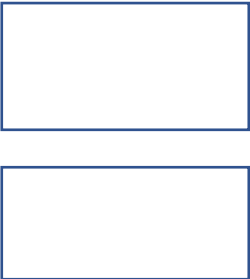
- private – class itself
- default (no modifier) – plus other classes in same package
- protected – plus subclasses in another package
- public – plus all other classes

Stack

Heap

Static Heap

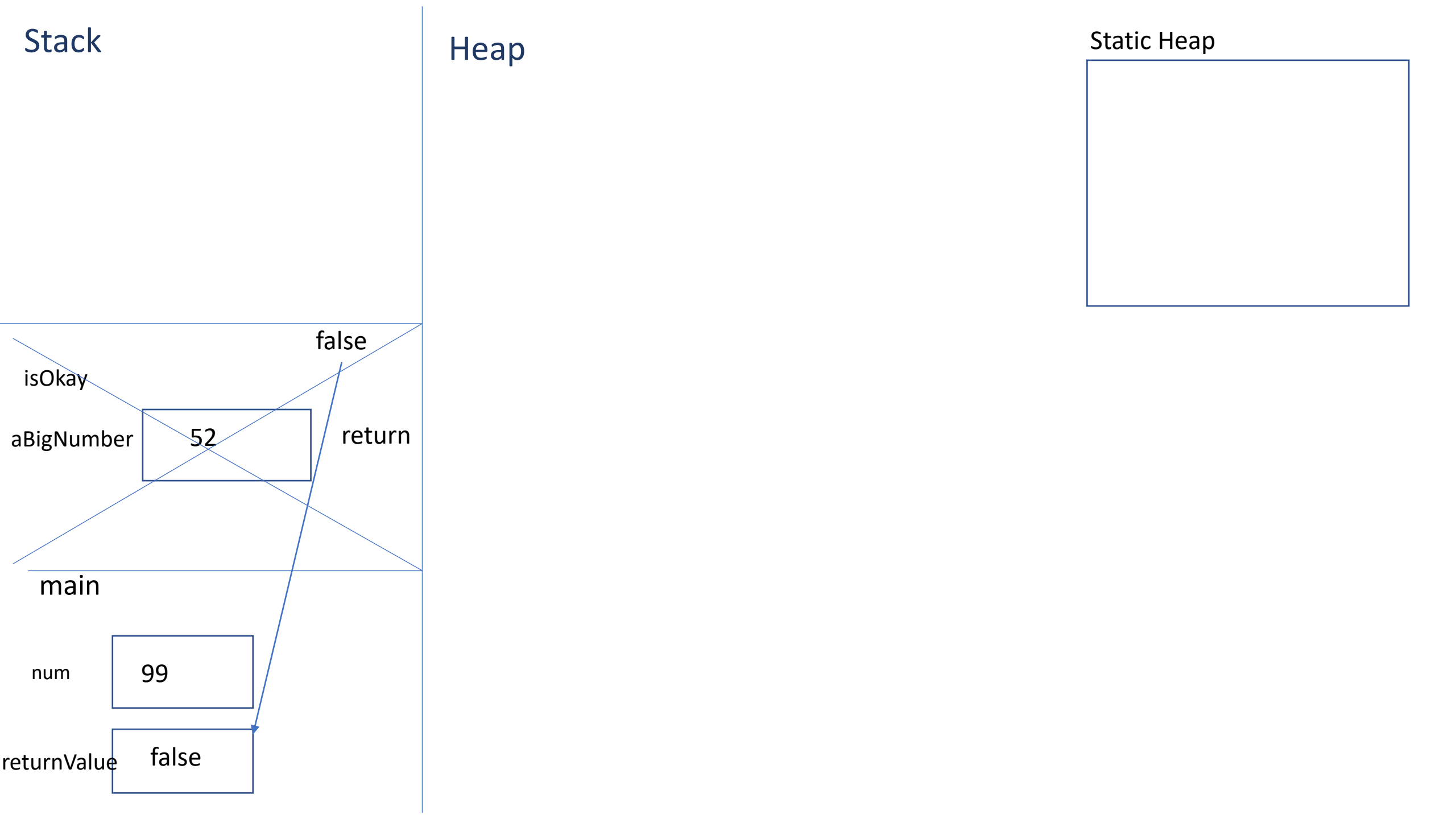
main



Stack

Heap

Static Heap



# Stack

# Heap

# Static Heap

