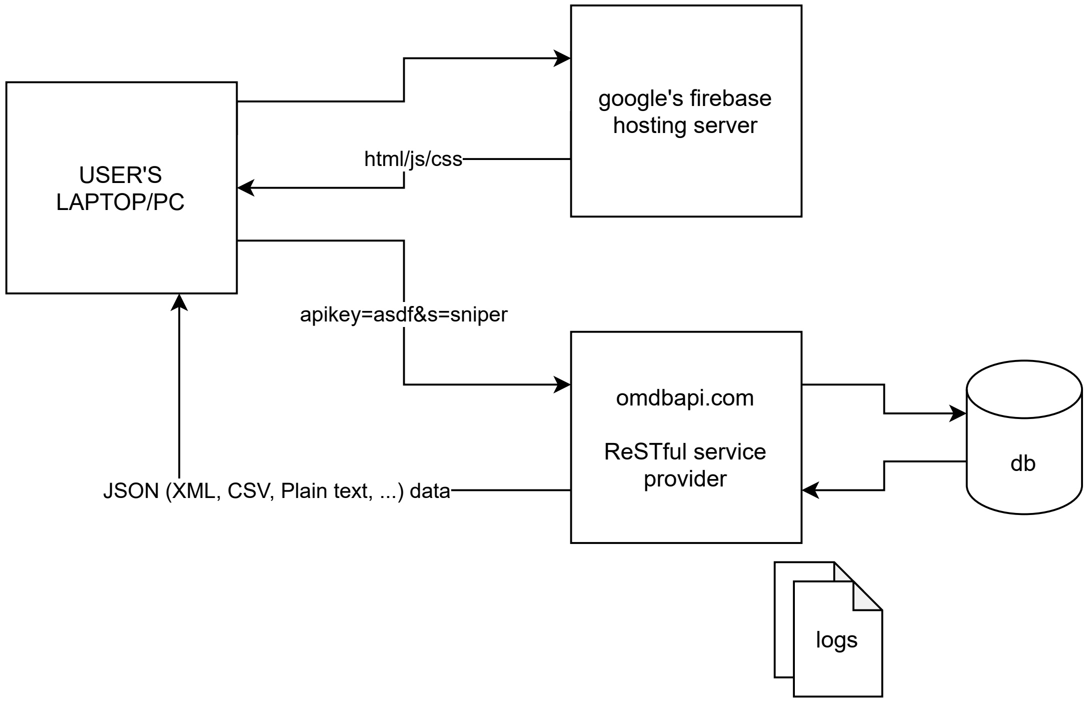
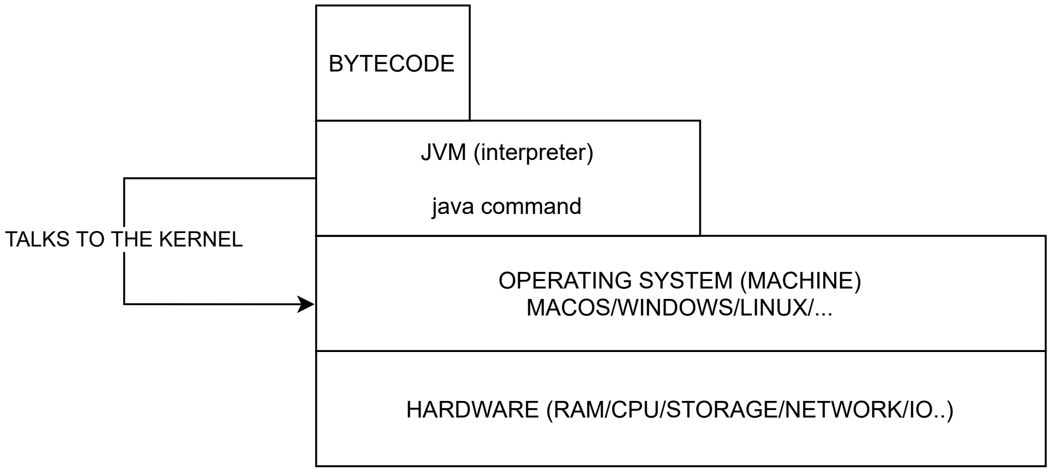




Primitives

1. Integers
   a. byte --> 1 byte (or 8 bits)
   b. short --> 2 bytes (16 bits)
   c. int --> 4 bytes (32 bits)
   d. long --> 8 bytes (64 bits)

2. Real numbers
   a. float --> 4 bytes (32 bits)
   b. double --> 8 bytes (64 bits)

3. Characters
   a. char --> 2 bytes (16 bits, allow unicode)

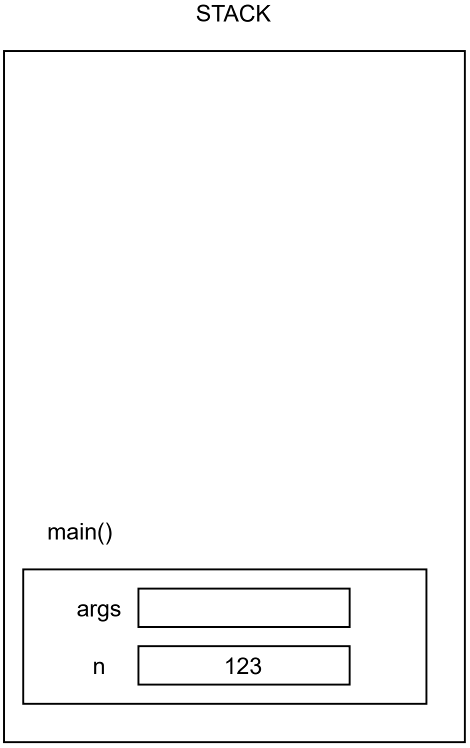
4. Logical
   a. boolean --> 1 bit (true / false)

*no sizeof operator to actually check the size of a variable
** no unsigned datatypes

References

Variables of :
1. class
2. interface
3. enum
4. annotation
5. array

* if the variable is not a primitive, then it is a reference
** size occupied by a reference itself is 4 bytes in 32 bit system and 8 bytes in a 64 bit system



Wrapper classes:

Part of java.lang package
byte --> java.lang.Byte
short --> Short
int --> Integer
long --> Long

float --> Float
double --> Double

char --> Character

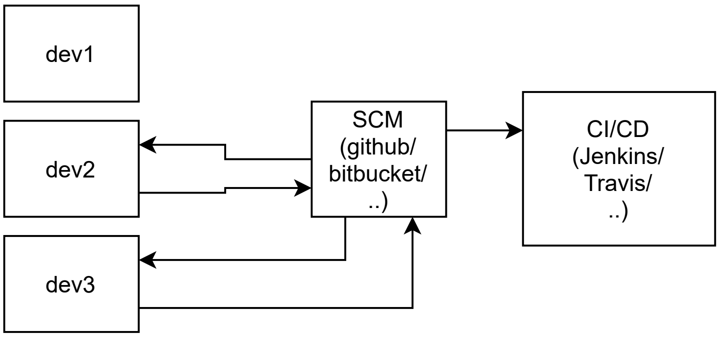
boolean --> Boolean

Programming constructs

1. Sequence
   a. top-bottom
   b. left-right
   c. change the sequence by invoking functions

2. Selection
   a. if-else
   b. switch-case
   c. ternary operator

3. Iteration
   a. while
   b. for
   c. do-while
   d. enhanced for loop (for-each loop)
   e. recursion (don't use)



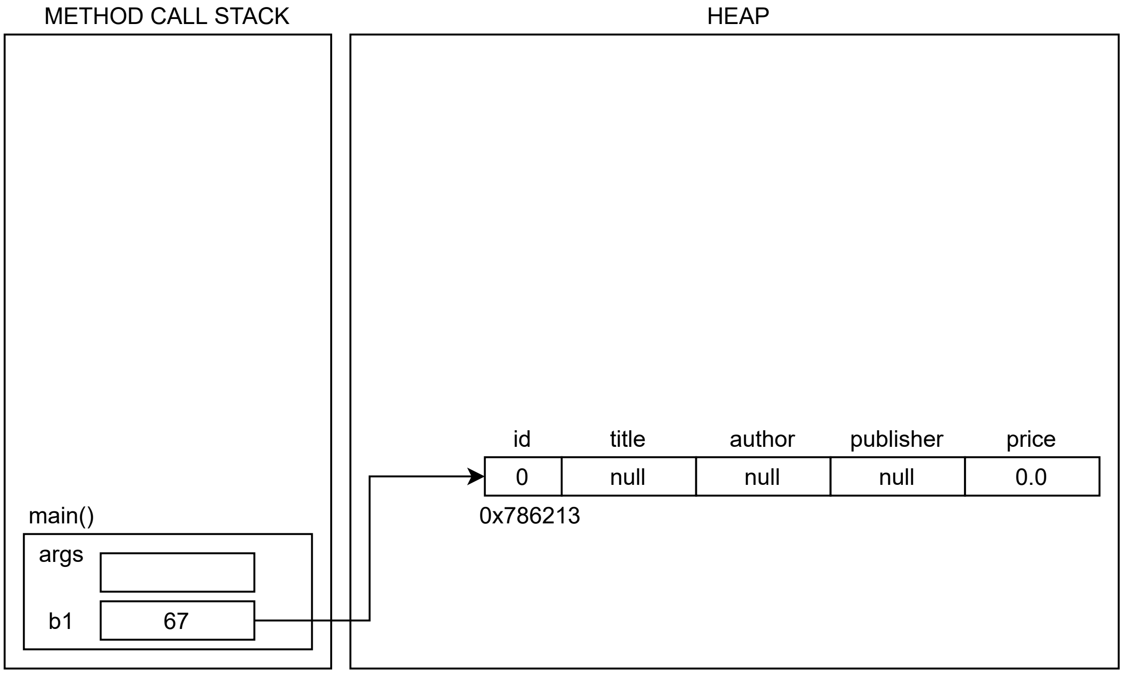
CI/CD

code pull from scm
compile
run tests
check code coverage
check code quality (sonarqube)
run integrated tests
build
deploy

Elements of OOP

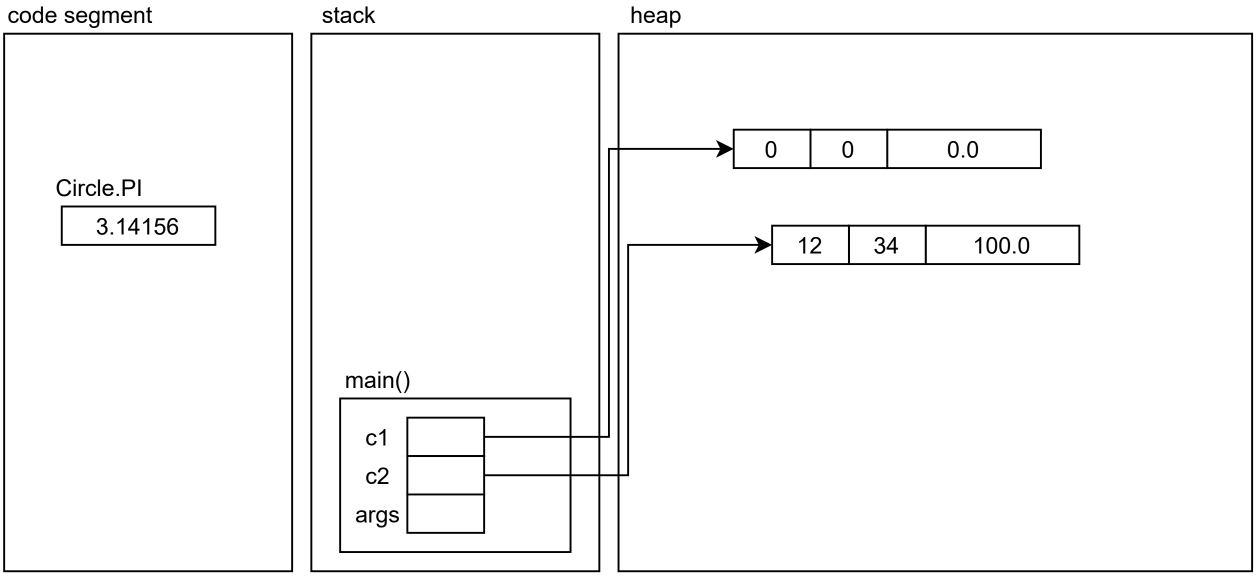
MAJOR ELEMENTS
1. Abstraction - A class hides the implementation details from the user of the class.
2. Encapsulation - Restricting access to members of an object. Keywords: private, public, protected
3. Hierarchy - Aggregation, Composition, Inheritance, Association (for code reusability)
4. Modularity

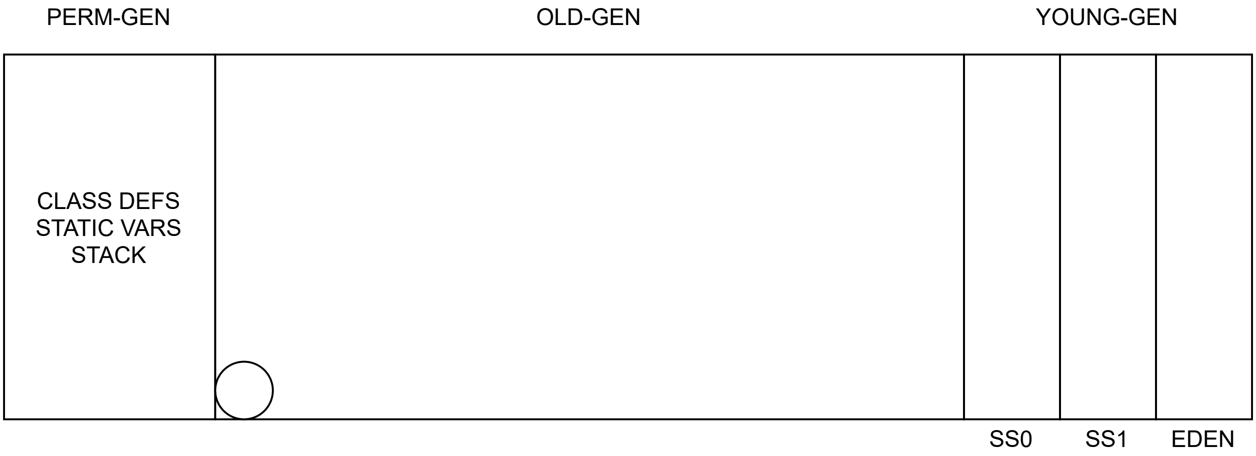
MINOR ELEMENTS
1. Typing
2. Persistence
3. Concurrency

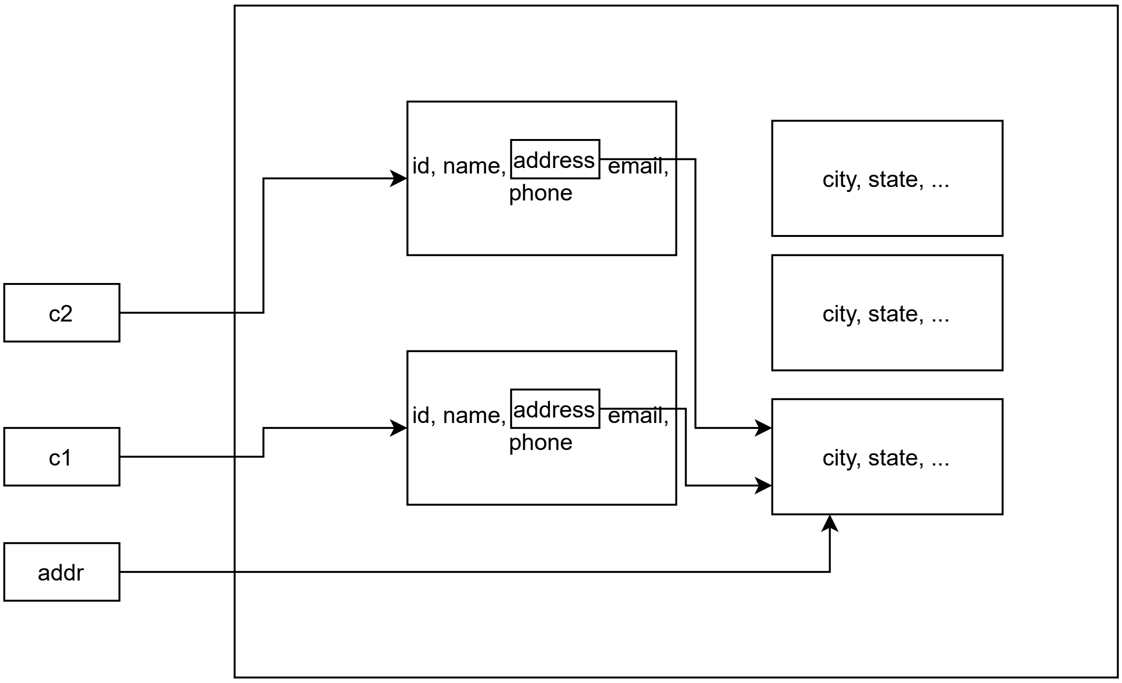


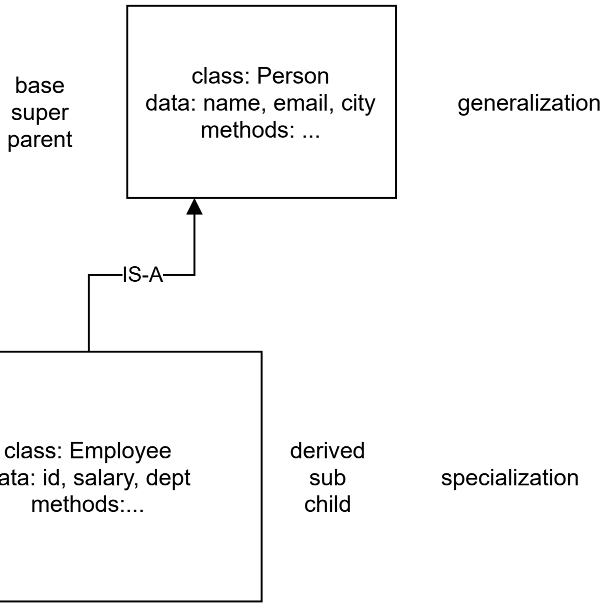
ref/addr table

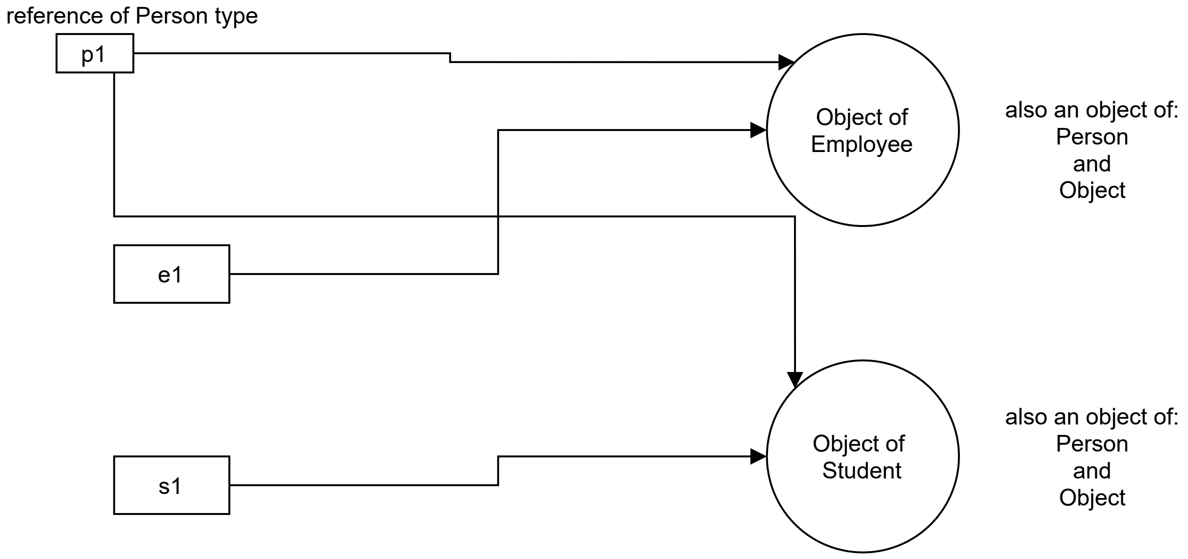
ref#.	addr	type
67.	0x786213.	com.epsilon.training.entity.Book

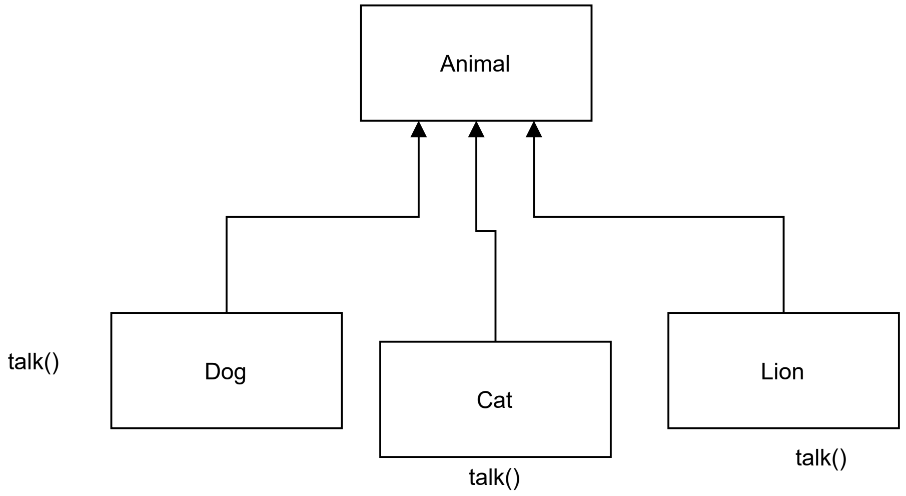


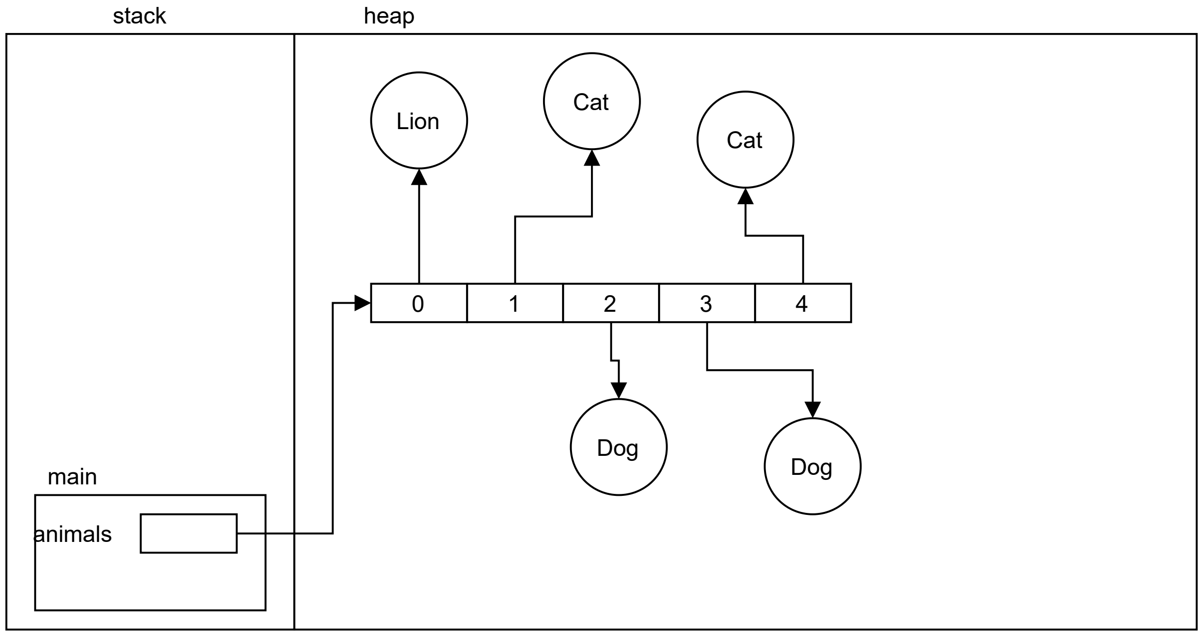

PERM-GEN
OLD-GEN
YOUNG-GEN
CLASS DEFS
STATIC VARS
STACK
SS0
SS1
EDEN

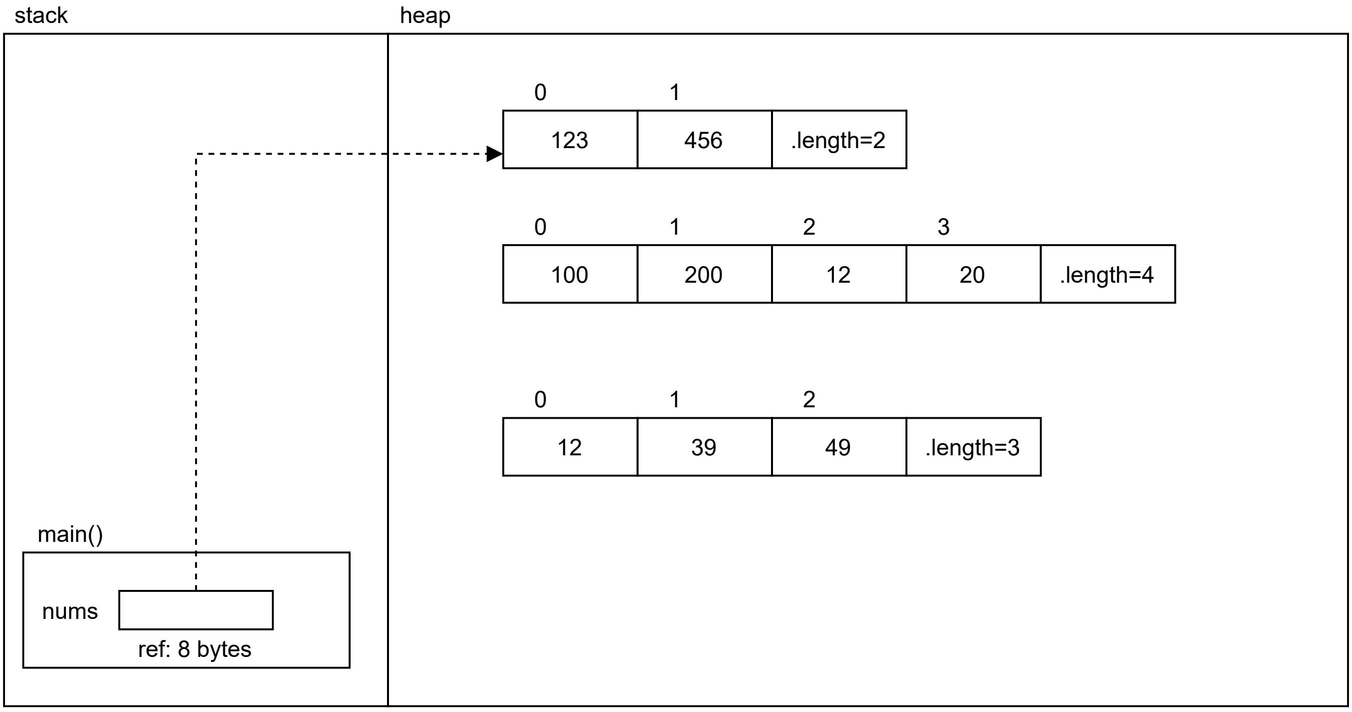

id, name, address email, phone
id, name, address email, phone
city, state, ...
city, state, ...
city, state, ...
c2
c1
addr

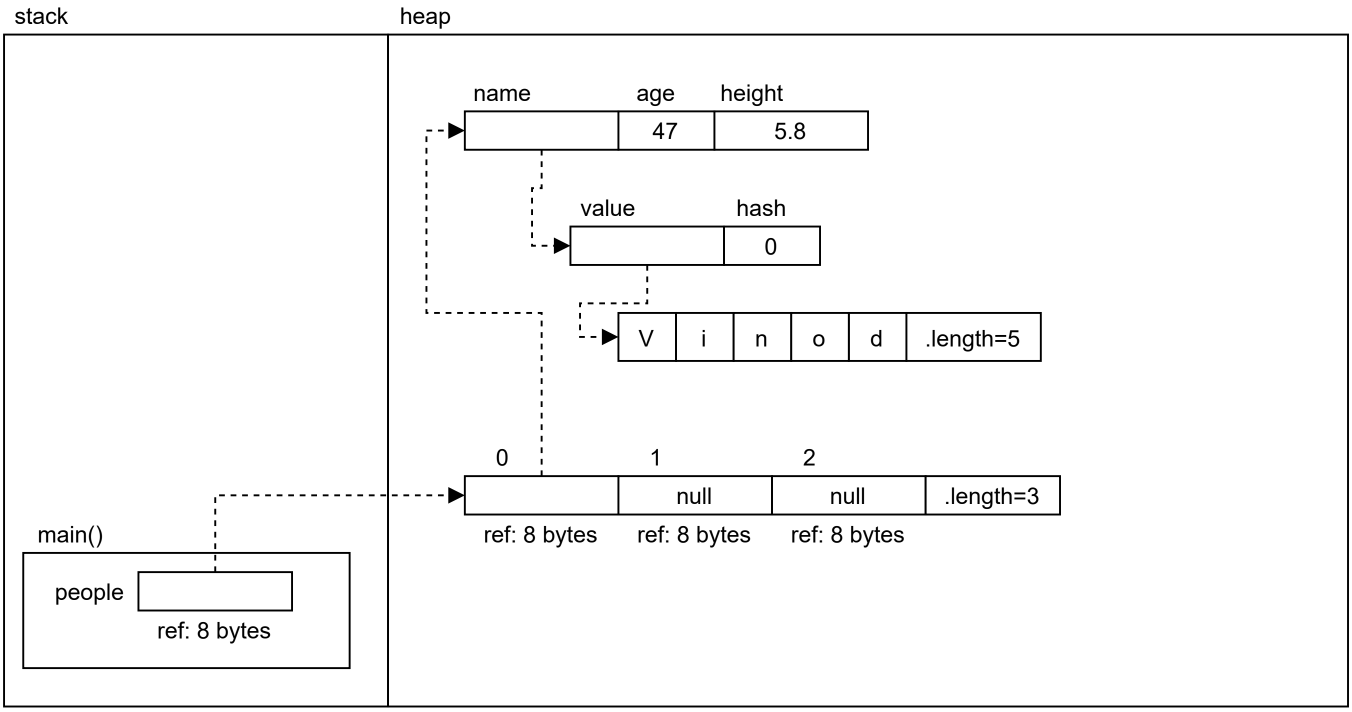

base
super
parent
class: Person
data: name, email, city
methods: ...
generalization
IS-A
class: Employee
data: id, salary, dept
methods:...
derived
sub
child
specialization

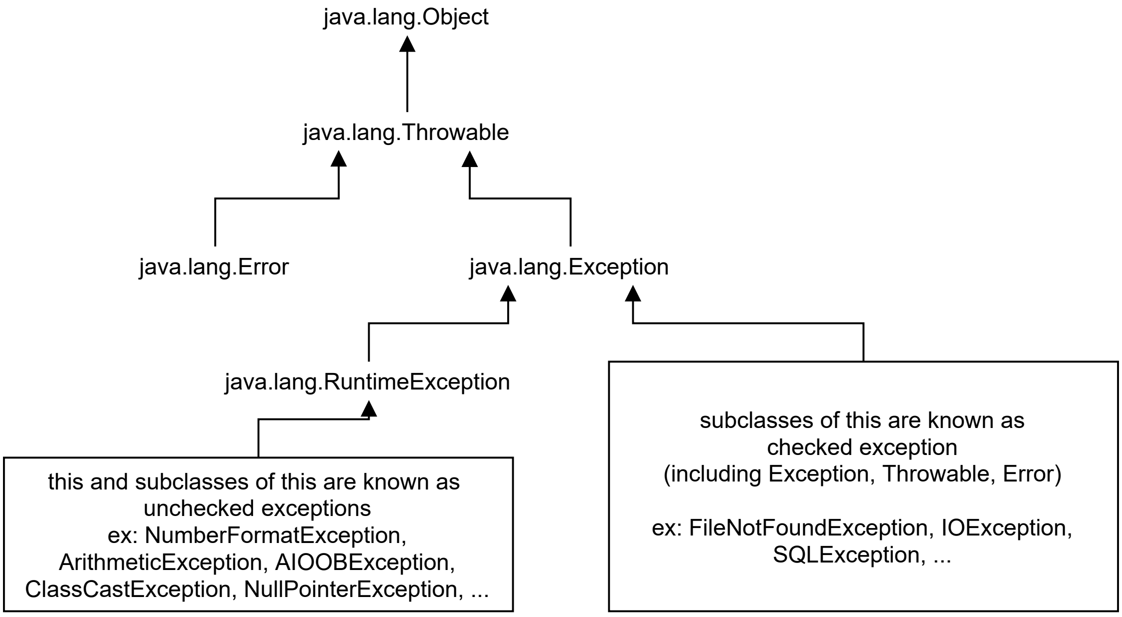

reference of Person type
p1
Object of Employee
also an object of: Person and Object
e1
Object of Student
also an object of: Person and Object
s1


Animal
talk()
Dog
Cat
talk()
Lion
talk()


stack
heap
Lion
Cat
Cat
0
1
2
3
4
Dog
Dog
main
animals



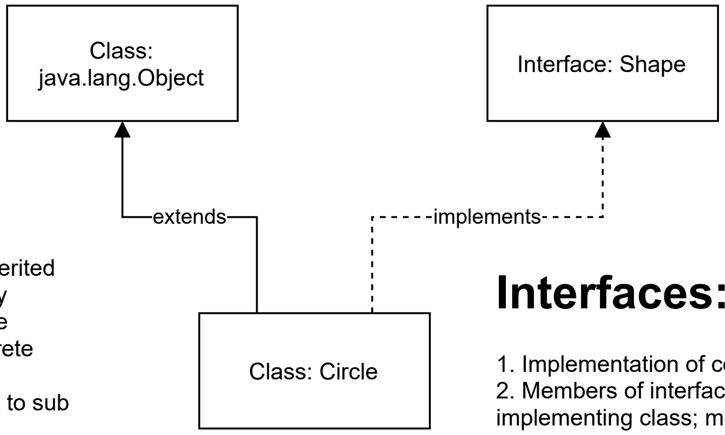




this and subclasses of this are known as
unchecked exceptions
ex: NumberFormatException,
ArithmeticException, AIOOBException,
ClassCastException, NullPointerException, ...

subclasses of this are known as
checked exception
(including Exception, Throwable, Error)

ex: FileNotFoundException, IOException,
SQLException, ...

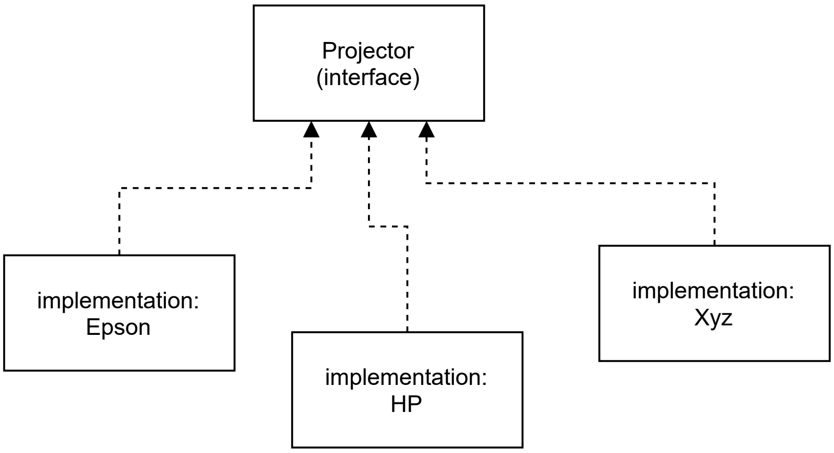


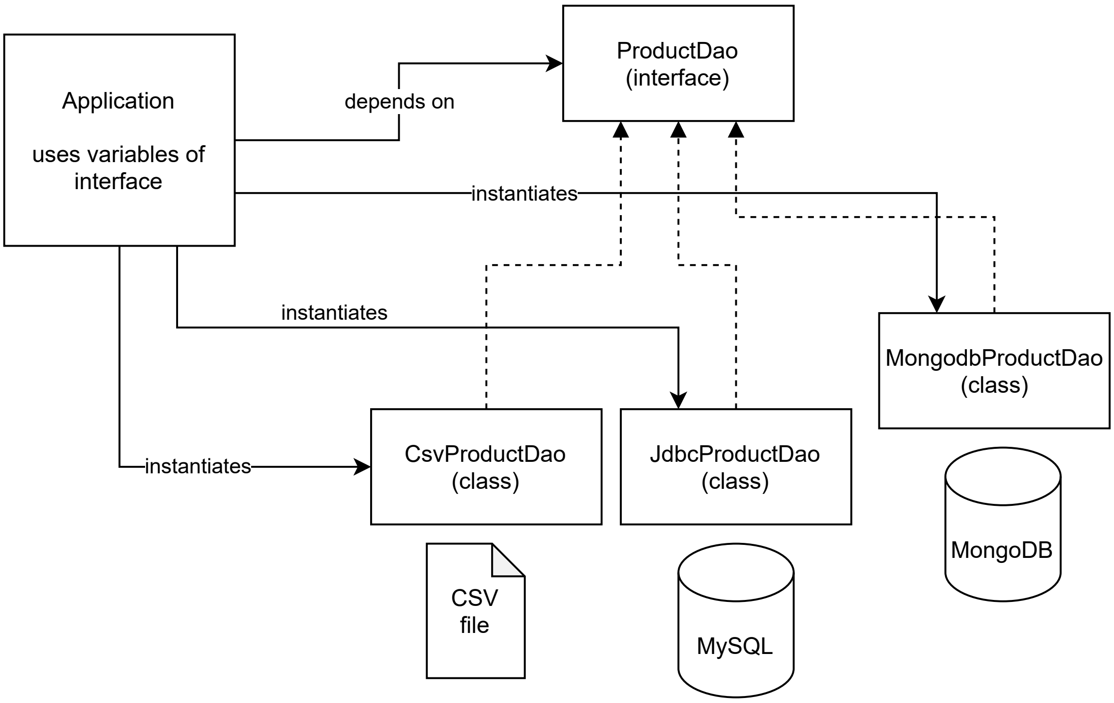
Inheritance:

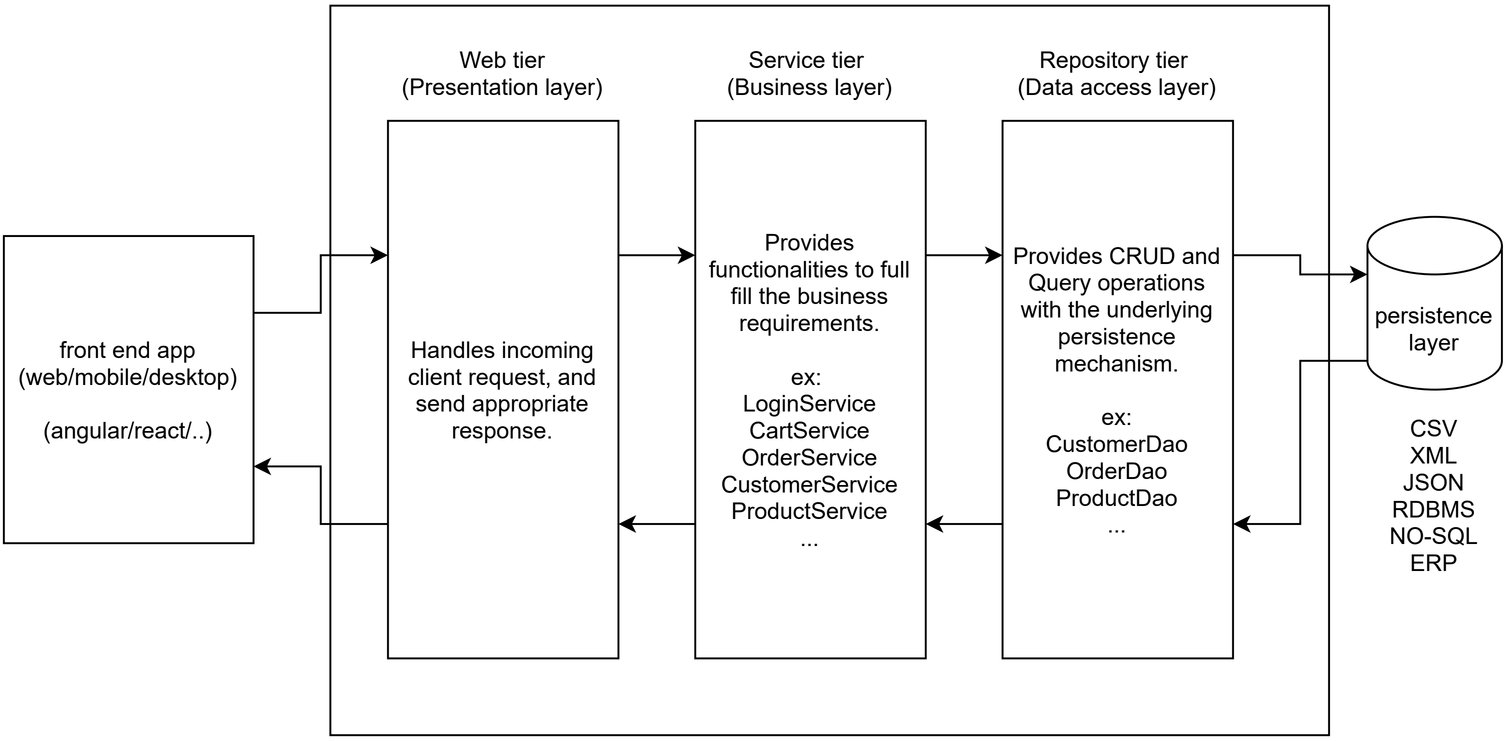
1. Members of super class are inherited
2. Primary goal --> code reusability
3. Super class may have 0 or more abstract methods, 0 or more concrete methods
4. Super class reference can point to sub class objects --> Polymorphism

Interfaces:

1. Implementation of contract methods
2. Members of interface are inherited to the implementing class; members: static final variables and abstract methods
3. zero code reusability (except for default methods; version 1.8+)
4. Primary goal --> polymorphism; realization of interface objects via concrete classes
5. Loose coupling between different layers of application






front end app
(web/mobile/desktop)
(angular/react/..)
Web tier
(Presentation layer)
Handles incoming client request, and send appropriate response.
Service tier
(Business layer)
Provides functionalities to full fill the business requirements.
ex:
LoginService
CartService
OrderService
CustomerService
ProductService
...
Repository tier
(Data access layer)
Provides CRUD and Query operations with the underlying persistence mechanism.
ex:
CustomerDao
OrderDao
ProductDao
...
persistence layer
CSV
XML
JSON
RDBMS
NO-SQL
ERP