Java $\longrightarrow$?

Programming Language

Java $\longrightarrow$ OOP's $\longrightarrow$?? $\frac{}{}^n$

C++
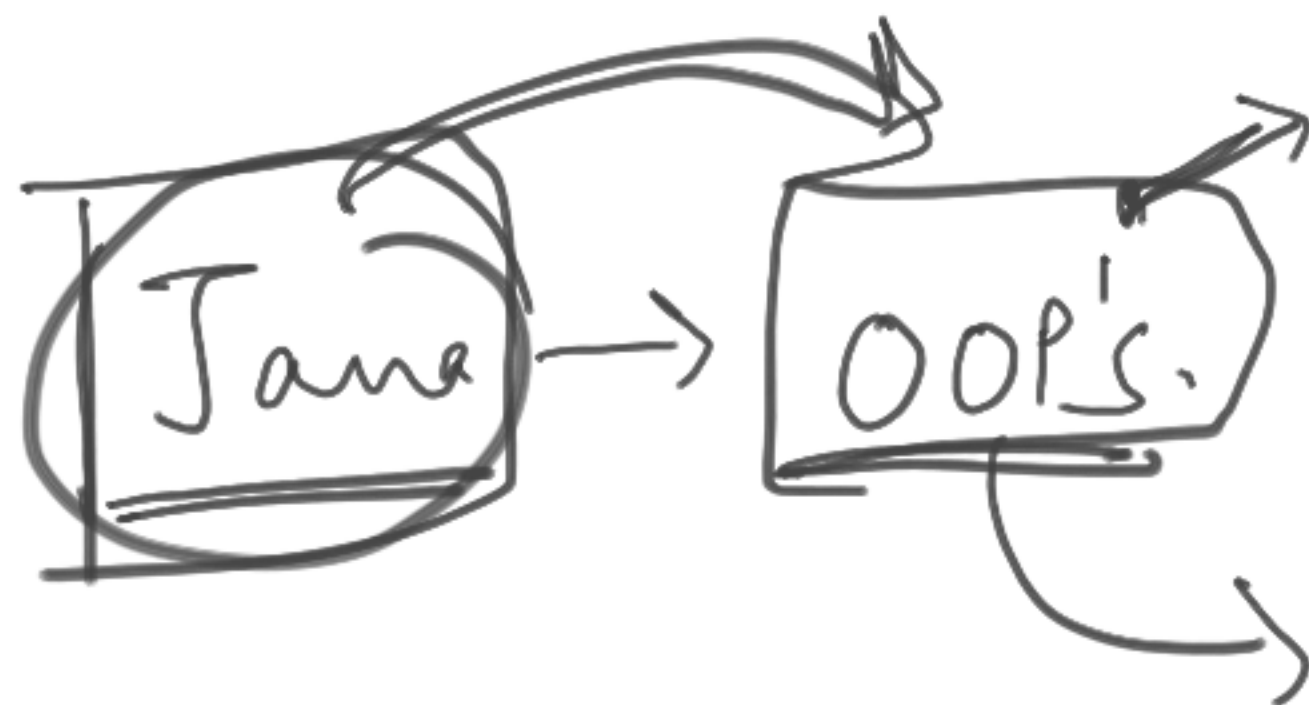
OOP's

Paradigm

Approach

OOP's

4 Pillars ⟶ Abstraction

Object
Oriented
Programming

⟶ Encapsulation

⟶ Inheritance

⟶ Polymorphism.

JAVA

Core $\rightarrow$ Object — ??

C

Core $\rightarrow$ Functions

Real World

Entities

Relation in a smooth.

Students

Mentor —

Admin —

ATM $\rightarrow$

ATM

1. (Card)
2. (User)
3. (Bank Account)
4. (ATM Machine)
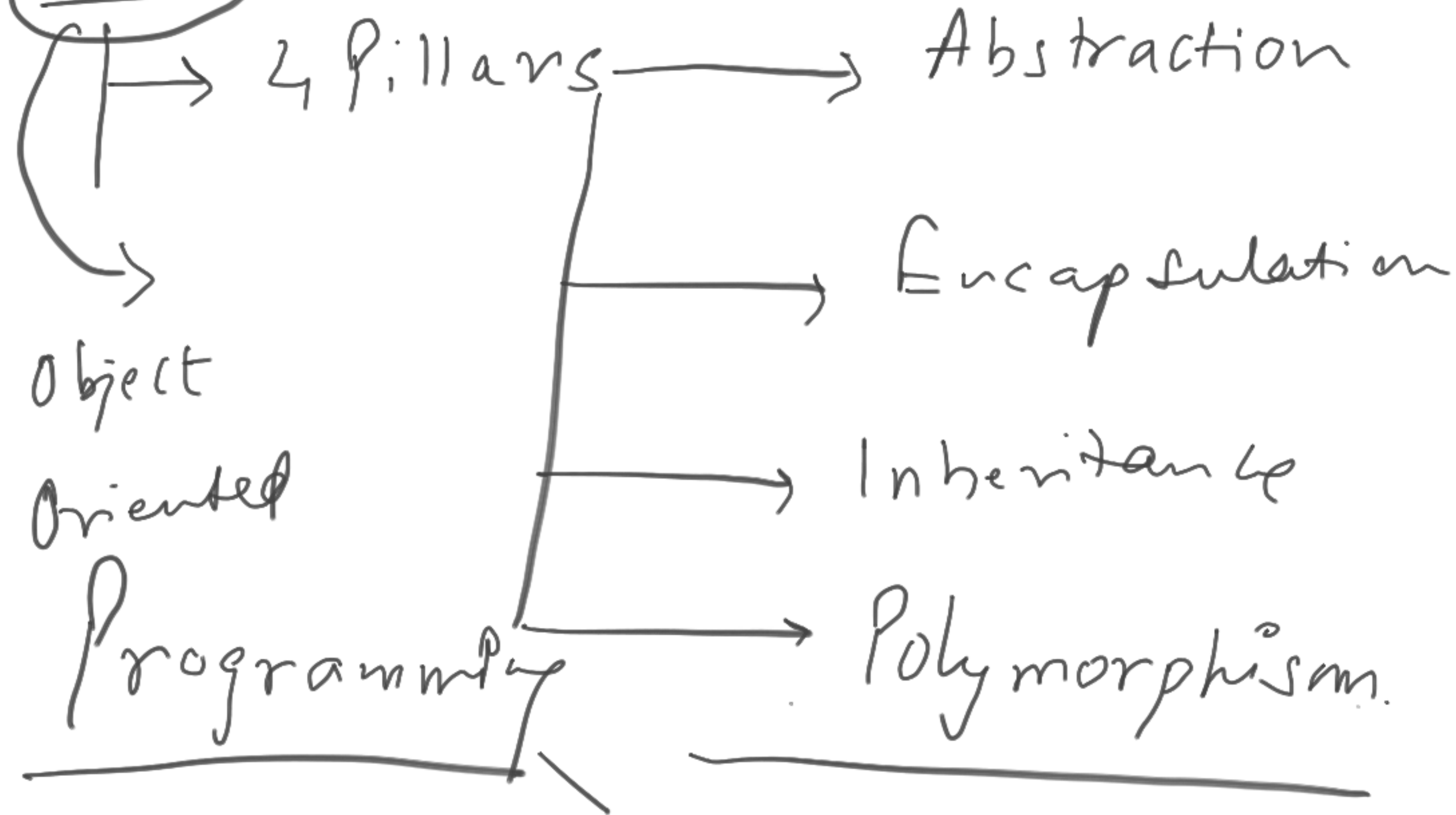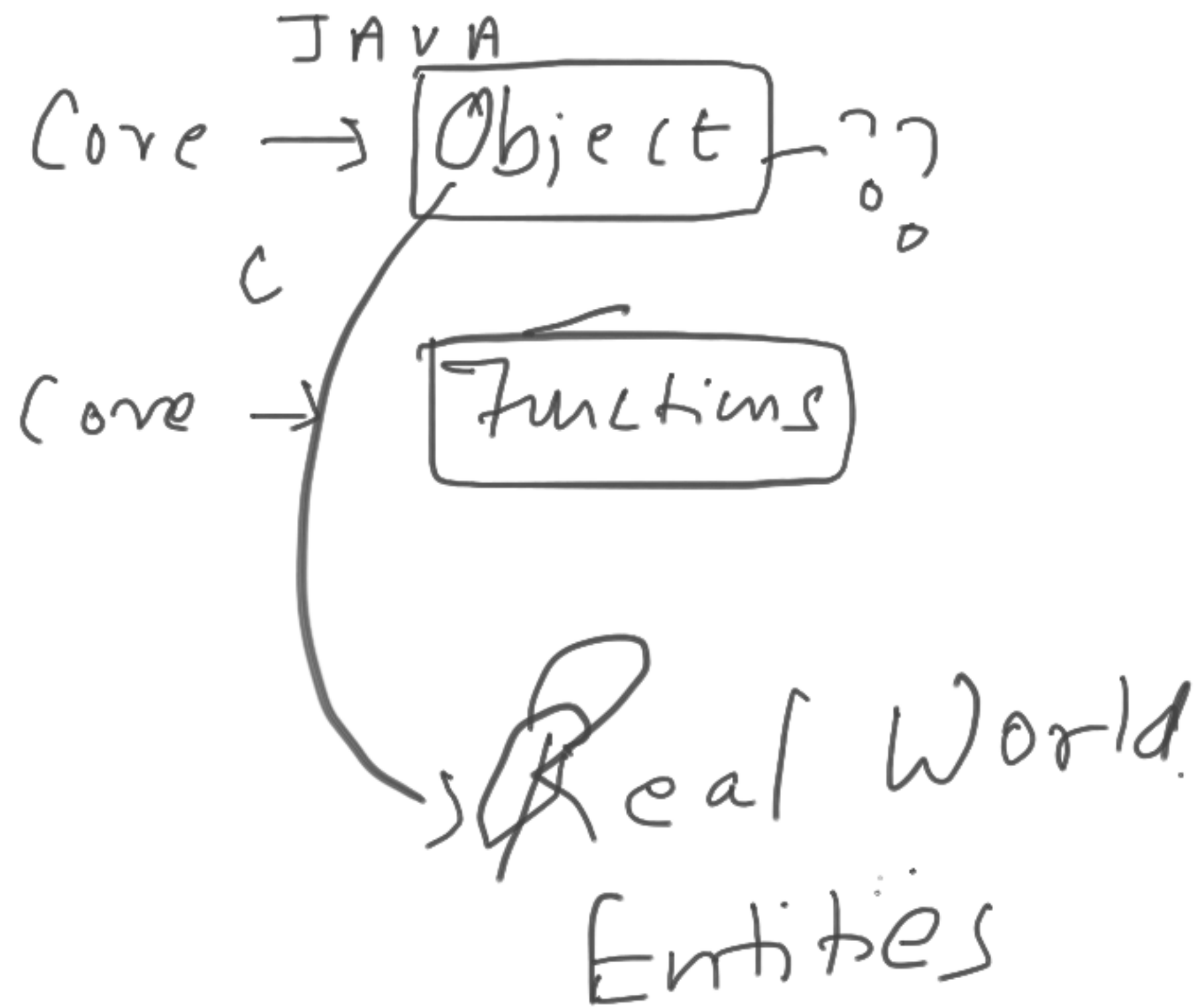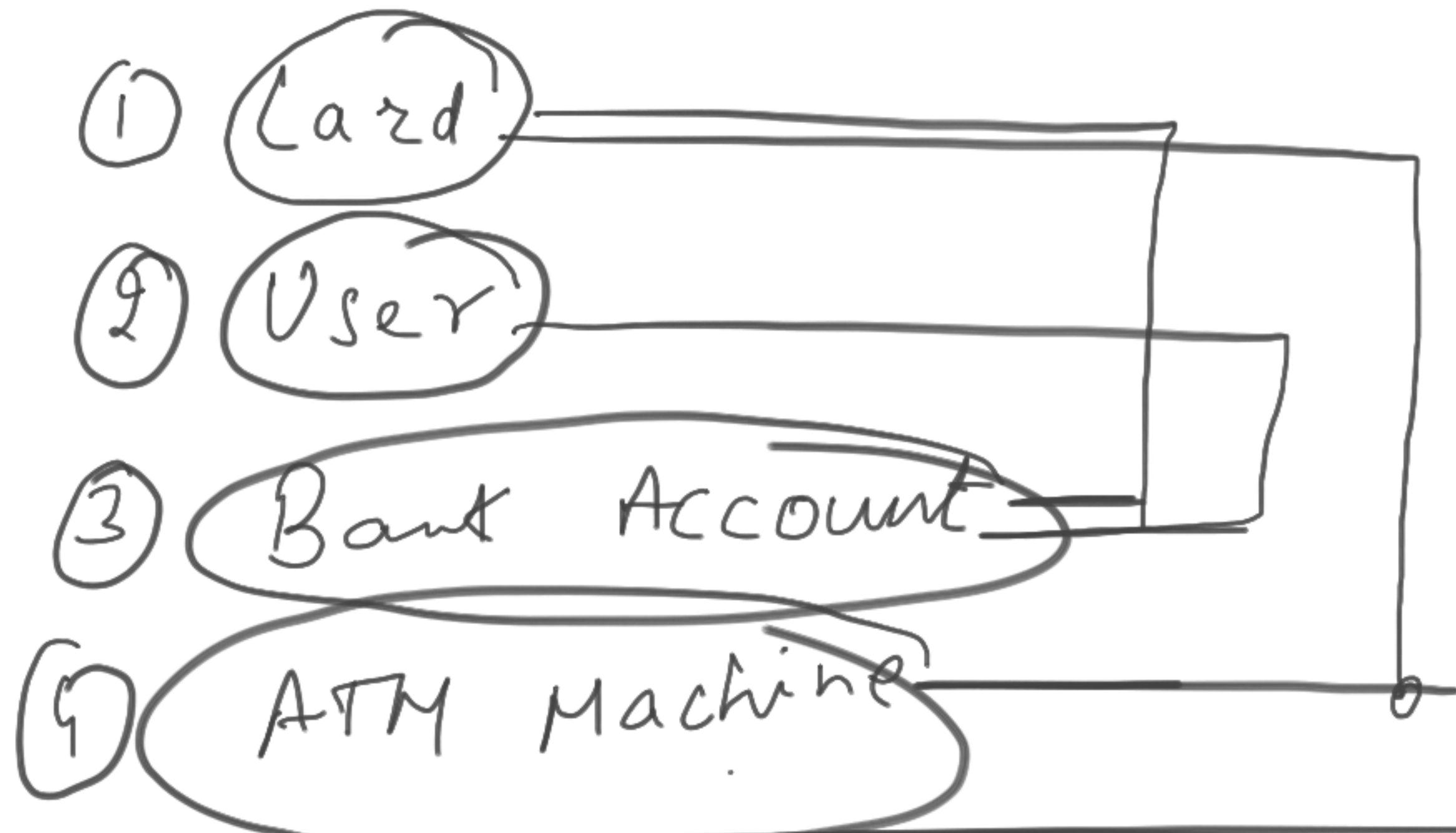
(Behaviour) | 3 Close 4 Open

Bank Account

1. Number — (Identity)
2. Branch =
3. Type
4. Name / ID
5.

A
T
T
R
I
B
U
T
E
S
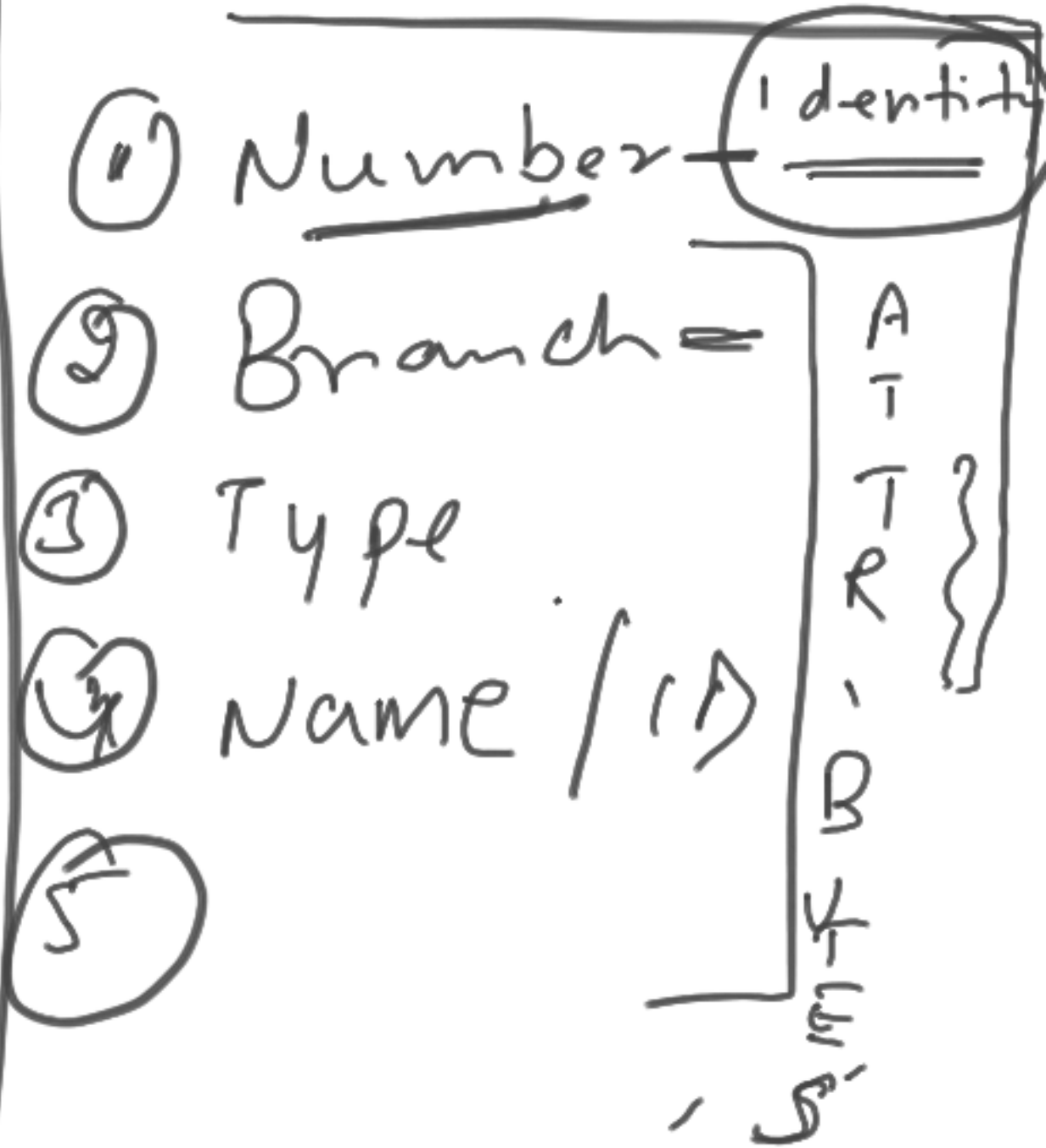
1. generate Statement
2. Withdraw Money

Blueprint or map ~ $\longrightarrow$ CLASS.

prototype

ATTRIBUTE

(Behaviour)

$\downarrow$ Attribute

$\downarrow$ Behaviour

(Function)

data member

member function

OP's

Abstraction :- → NOT VISIBLE

ATM → NOT KNOWN.

Data Hiding.

Encapsulation ⟶ Binding

D.M.

M.F..

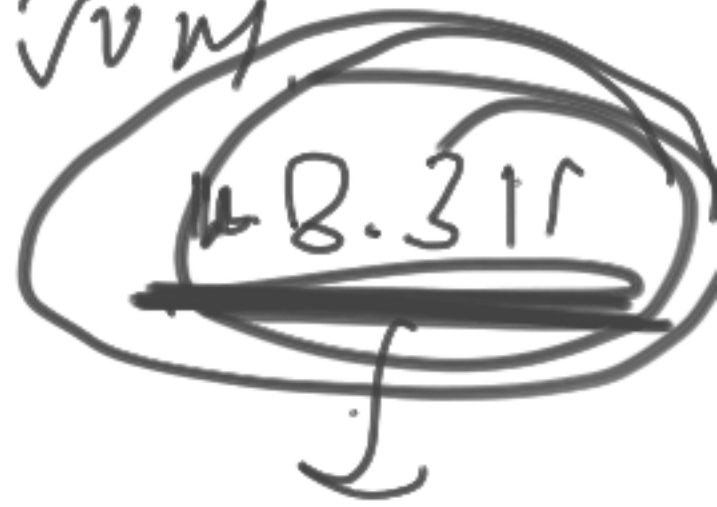Getter/Setters

class →   Keyword

A.S class C.N {

BODY

}

Blueprint

```
public Account {
    int id;
    int cid;
    int bal;

    withdraw();
    deposit();
    :
    :
}
```

C
L
A
S
S.

1 | 3

JVM **Platform Indep**

Java

.java $\longrightarrow$ .class

Java $\longrightarrow$ Windows
Linux

C | B |

onetype

Run many JVM
API version

1.8.311

JDK
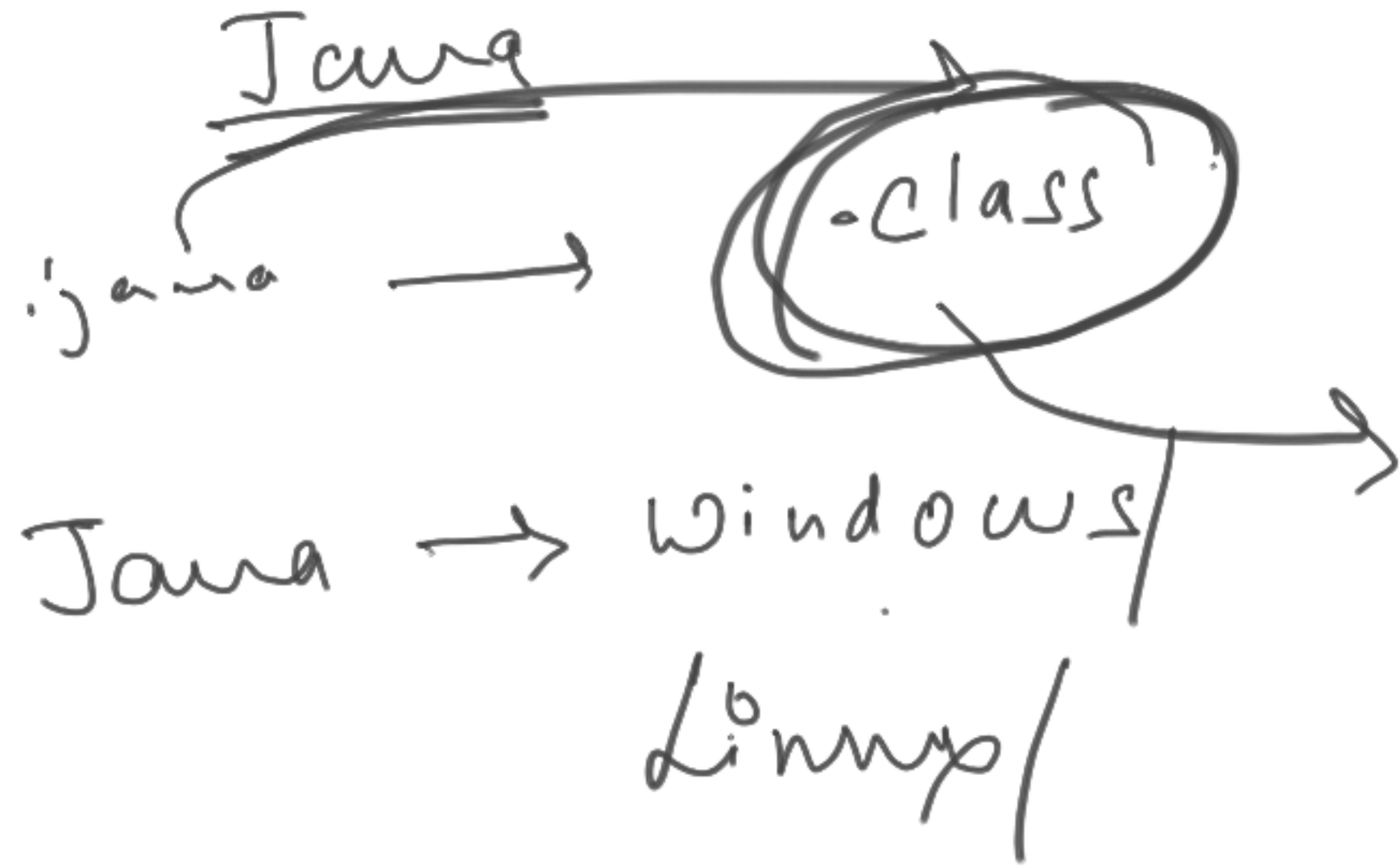
classes
+
JRE

Main.java
JDK

javac Main.java
JDK

JRE

Java
Runtime
Env.

Main.class

java Main

JVM (Running)

Java Virtual Machine

→ chrome

on any **JRE**

JVM

Psvm

Main.main → Runs.

Public class name

Static

void
doesnot
return
anything function

main.
name
of

AS
this can be
directly outside
the package.

Any object/
variable/fⁿ.
which static
can be accessed
without creating directly
by the class Name.