Good Morning :)

Revise -
- DML commands - Insert Update Delete
- DDL commands - Create table integer , varchar , jsonb ,boolean
  - SQL = PostgreSql = DIALECTS
  - Drop
  - Alter
  - Foreign Keys = Purpose = relate two tables

Normalization = reduces data redundancy = Storage better, lesser conflicts
1NF
2NF
3NF

JOINS = Inner Join ,Outer Join
Cartesian Product - CROSS JOIN
------ PAIRING each row of one table with each row another table

a	B	C
e	F	g

1	2	3
4	5	6
7	8	9

a	B	C	1	2	3
A	B	C	4	5	6
A	B	C	7	8	9
E	F	G	1	2	3
E	F	G	4	5	6
E	F	G	7	8	9

Subset of CROSS join records would be INNER JOINS and OUTER JOINS
That will be done based on some condition = FK condition

Indexing = A Table can have an index column !!!
WHY to have index ? **It is useful for quick search**
AFTER we create a table
AFTER we have inserted rows
We observe the table access pattern
We can decide upon a column that should be made as index column

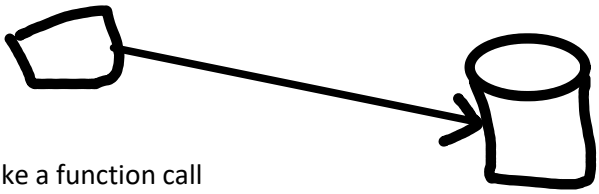
Views : Logical
: is it good for Query statements or DML statements ?
Select queries are simplified

PLSQL---

PRE-COMPILED query

Select queries are simplified

PLSQL---



PRE-COMPILED query
Is already here
Runs here
Returns the result (scalar/table

Client
make a function call

PL/pgsql is on the Server side

Insert into book….
Query

Query is compiled and executed
Result set is sent back

When I think of creating a TRIGGER

A. Name of the trigger = CREATE TRIGGER xyz
B. Condition of the trigger = AFTER UPDATE on employees
C. HOW MANY TIMES = FOR EACH ROW
D. EXECUTE FUNCTION plpgsql_func() *****MAPPING the code with TRIGGER

```
CREATE TABLE orders (
   order_id SERIAL PRIMARY KEY,
   customer_name TEXT NOT NULL,
   dish TEXT NOT NULL,
   quantity INTEGER NOT NULL CHECK (quantity > 0),
   price NUMERIC(10, 2) NOT NULL CHECK (price >= 0),
   order_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);


CREATE TABLE order_log (
   log_id SERIAL PRIMARY KEY,
   order_id INT NOT NULL,
   log_time TIMESTAMP DEFAULT NOW()
);


CREATE OR REPLACE FUNCTION log_order_insert()
RETURNS TRIGGER AS $$
BEGIN

RAISE NOTICE 'New order inserted with order_id = %', NEW.order_id;

   INSERT INTO order_log(order_id, log_time)
   VALUES (NEW.order_id, NOW());
   RETURN NEW;
END;
$$ LANGUAGE plpgsql;



CREATE TRIGGER after_order_insert
AFTER INSERT ON orders
FOR EACH ROW
```

EXECUTE FUNCTION log_order_insert();

INSERT INTO orders (customer_name, dish, quantity, price) VALUES ('Jane Doe', 'pizza', 3, 999);

prach1g/Dbank2025
https://github.com/prach1g/Dbank2025/tree/main

psql
\i ~/triggers.sql

Core Java -----
SANDBOX

JAVA programming Language = Object Oriented Language
= Platform independent Language

Platform = OS + Microprocessor

SOURCE CODE = The human readable/Writable code = XYZ**.java**

SOURCE CODE is platform independent = **no platform specific API calls in the source code**

EXECUTOR PROCESSOR (execute the instructions)
Will the processor understand the XYZ.java source code ? NO

SOURCE CODE ----->Translated -----**COMPILER** -----> CLASS FILE (Bytecode= Target Lang)
XYZ.java ----------XYZ.class

Will the processor understand the XYZ.class ? **NO**

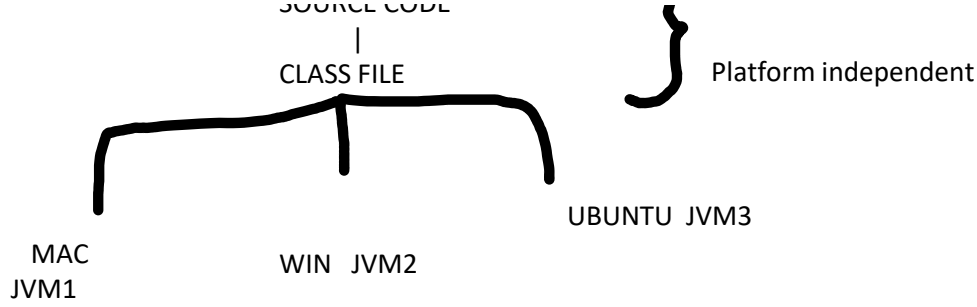
XYZ.java and XYZ.class both are PLATFORM INDEPENDENT = no platform understands it

XYZ.class ------> Translated ---**INTERPRETER (JVM = JAVA interpreter**) ---> platform specific code

EXECUTOR understands the platform specific code and EXECUTES it !!!

Compiler	**Interpreter**
Translator	Translator
Save the translation in an output file XYZ.class	NEVER saves the translation to any file
Never executes only translate and save	It executes every line it translates

SOURCE CODE	XYZ.java	Platform IN-dependent
Compliled Code	XYZ.class	Platform IN-dependent
Interpreted code	Not saved , executed directly	Platform dependent



JDK --------- Java development Kit === download it
Compiler
JVM interpreter

STEP 1 = check your SANDBOX whether it has JDK
c:\program files \java\jdk

STEP 2 = Run a hello world java program on windows command line
javac = to compile
java = to run

NAMING CONVENTION :-

All keywords	small case
Class names	CONVENTION : First (begin with a caps), **F**irst**E**xample
Function names	**d**o**J**ob() : begin with small
Property/variable names	begin with small - fname , first**N**ame
constant	CAPS MAX, MAX_VALUE

Starting point of ANY java program --
public static void main(String[] args)

Compile Time **First.java ===> First.class**

Run Time **First.class ===>output**

java First }}} SEARCH for First.class
JVM will Call the public static void main (String[])
run the code { }

IDE = Integrated Development Environment
What to integrate ?
Tools = Editor + CMD + javac + java

Intelliji or Eclipse IDE

Should the name of class and name of the file be always Same ?
It MUST be same if class is public
otherwise it could be different

Can we have more classes in a XYZ.java file ? YES
can all the classes be public ?
No - only one public , other classes will be non public

If Main.java has 3 classes
public class Main
class Two
class Three
Q. How many .class files will we have ?
One .class file is created per class !!!!! JAVA MODULARITY
Q. If each class has a main
Then the main passed to java command will run
Java Three } starting point in class only runs

Program - start point and end point
Java program - p.s.v.main(String[] args)
{ ---- start
.....
.....
....
} -----end

Ex - Add two numbers and print the sum

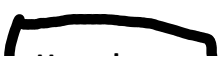
Data types of all possible variables

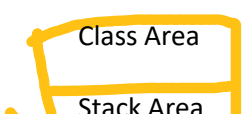
4 integer	byte (1 byte), short (2bytes) , int (4 bytes) long(8 bytes)
2 decimal point	float , double
boolean	
char	

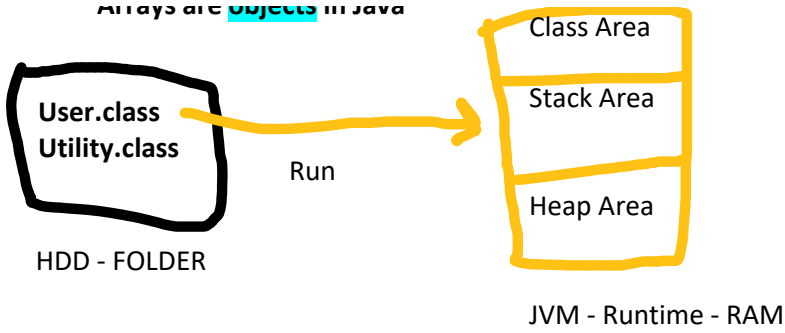
Classes have functions
1. Static function
2. Non static function

User class ,
---- Starting point main
Utility class
--- static calcPower
--- static calcFactorial

Arrays in Java -
Arrays are objects in Java







Class Area = .class files and the class data (static)
Stack Area = Stack of Activation Frames which are PUSHED and POPPED
PUSH = append to the top
POP = remove from the top
LIFO

Pushed = when a function is called the FRAME is PUSHED
Popped = when the function returns the FRAME is POPPED

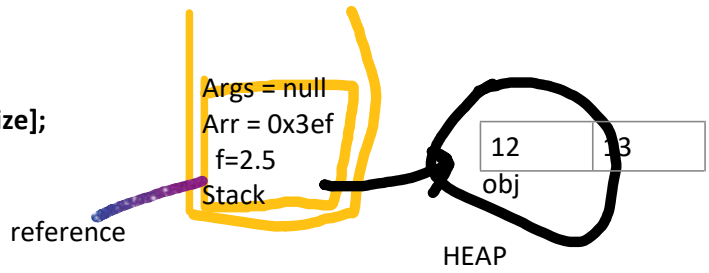
FRAME =the localvariables , parameter, return values

Heap Area = Where all the objects are stored

In java any object is created using new keyword !!!

```
p.s.v.main(String[] args)
{
    float f = 2.5f;
    int[] arr = new int[size];

}
```



Parameter passing in Java
---
if a Primitive variable is passed = its copy goes to the function
Changes are done in the copy
Original remains intact

If an object is passed = the copy of the reference is creaed

Arr caller reference x callee reference

main uses it as arr populate array uses it as x

---

public static void main(String[] args)

= starting point of the JAVA program
= called by the JVM
= the parameter that is the String[] will be passed by theJVM
= The values given on the command line are passed as a String [] to main

_____

A class
STATIC fucntion ------
add()
Utility.calculate()
Boolean.parseBoolean()

A class is a blueprint - design of an object = Description of an object
Description includes - properties , behaviour , how to initialize the object !!!

Pen class
Inkcolor
Cost
Brand

Write()
displayInfo()

Constructor