Static vs dynamic memory management::

class theke amra j object create korchi setai dynamic memory location,

static use kore j variable declare korchi setai static memory location.

Static memory allocation hoy compile time e but dynamic memory allocation hoy runtime e.

Static r size fixed but dynamic r size variable depends on the programmer basically.

Programmer analyst er kaj holo define, develop, test, analyze, maintain new software application in support of the achievement of business.

Delete and truncate both use kora hoy database e table theke records delete korar jnno.

TRUNCATE TABLE table\_name use kore table er structure e kono change asena but table e thaka sob records delete hye jay, mainly er kaj ta hche eta first table r structure r ta k copy kore then complete table ta k drop kore dey.

But DELTE FROM table\_name; each record k directly delete kore.

And drop puro table along with it’s structure delete kore dey.

Delete maily where condition r sathe use kora hoy.

DELETE FROM table\_name WHERE condition;

DELETE FROM table\_name; records gulo k delete kore dey.

DROP TABLE table\_name; puro table k delete kore dey with it’s structure.

TRUNCATE TABLE table\_name only records delete kore.

Static keyword:

Static variable k class variable o bola chole as it’s charactarestics is quite similar to class. Mane amra jokhn static variable create kori tokhn sei variable r ekta copy create hye jay and er sob instance e ei copy ta kei use kora hoy. Static variables or method class er modhyei use kora hy generally rather than it’s instamce mane onject e.

Static method k amra classer baire theke call korte pari without creating an instance of that class.

Mane amra Jodi emni ekta instance variable call kori sei variable tar scope holo within that class but static hole classer baire thekeo call korte pari even without creating an object of that class.

Static variable k different place e use kora geleo only ekta memory tei eta store thake.

The value of a static variable can be changed anytime during the program's execution.

PSVM:

public is a access modifier in java, used while creating any methods, variables etc. public k amra jekhan theke khusi access korte pari mane same class, same package, different class, different package. Void means it does not returns any value. Main is the methid that jvm looks for as the starting point of a java program.String[] args basically command-line programming mane git e “javac CommandLineExample.java “and “java CommandLineExample arg1 arg2 arg3” eta run korale and then

ei agrs gulo k chaile amra print korate pari

public class CommandLineExample {

public static void main(String[] args) {

if (args.length > 0) {

System.out.println("Command-line arguments:");

for (int i = 0; i < args.length; i++) {

System.out.println("args[" + i + "]: " + args[i]);

}

} else {

System.out.println("No command-line arguments found.");

}

}

}

eivabe. Ei arguments gulo string hisabe receive hche.

Mane jokhn java program run korachi tokhn ei args gulo k pass korano hoy. Most of the time amra kono args pass korai na.

Private::

Private instance variable or method k same class er modhye jekhan theke khusi mane kono method r vitor theleo access kora jay. But private k other classes theke kono vabei access kora jay na. basically private variable er jnno public getterm setter use kora hoy. Classer baire theke access korte gele compile-time error die debe.

Encapsulationer jnno private is a great weapon. Private k different package theke, instance of a class theke access kora jay na.

Protected::

Different package er non-subclass chara sob jayga thekei aceessable.

Default::

Same classer modhye sob jayga thekei accessible. CPP te default thake na.

JVM(java virtual machine)::

Runtime environment mane jre provide kore in which java bytecode can be executed making java a platform-independent language because the same java program can run on any machine that has jvm installed in it.

Memory te thaka .class files k jvm java bytecode e convert kore, then check kore ete kono error ache kina, then JIT(Just-In-Time) compiler use kore bytecode k machine code e convert kore execute kore.

Pre-installed garbage collector use kore memory management e help kore. Runtime environment mane jre provide kore.

JRE(java runtime environment):

Jvm+libraries= jre+development tools= jdk(jaba development kit)

 **JVM**: Executes Java bytecode and provides a platform-independent execution environment.

 **JRE**: Includes the JVM and libraries necessary to run Java applications.

 **JDK**: Includes the JRE and tools needed for developing Java applications.

Final::

Final variable is like a constant. Mane ekbar ete value initialize korle change kora jay na.

Declare korar time e eke initialize kore deoia uchit.

Final methods k subclass e override kora jay na.

Final class er subclass thake na,

Try-Catch::

Exception handling r kaj e use kora hoy. Unexpected conditions that occurs during the execution of the program. J code ta te error thakte pare mane j code ta exceptoion throw korte pare seta try r modhye thake and catch block r modhye code thake ja exception k handle korte pare.

try {

// Code that may throw an exception

} catch (ExceptionType1 ex1) {

// Handle ExceptionType1

} catch (ExceptionType2 ex2) {

// Handle ExceptionType2

} finally {

// Optional block, always executes

}

Multiple catch block use kore amra multiple exception handle korte pari.

Try{

Int[] arr=new int[5];

arr[10]=30;

}catch(ArrayIndexOutOfBoundsException e){

System.out.println(“Array index is out of bound”);

}finally{

Sysot(“checked”);

}

Using ‘throw’ keyword we can throw an exception explicitly.

throw new ExceptionType("Error Message");

Compile time e je exception gulo k check kora hoy setai checked exception.

IOException, SQLException, ClassNotFoundException

Compile time e jegulo k check kora hoy na setai unchecked exception. They are subclasses of ‘RunTimeException’ and these are thrown show programming logic errors.

NullPointerException, ArrayIndexOutOfBoundsException.

‘Throw’ die explicitly exception throw kora hoy but ‘throws’ k method er sathe declare kora hoy to say that method may throw an exception.

‘throw’ method bodyr vitore but ‘throws’ method r signature e.

‘throw’ has both checked and unchecked exception, ‘throws’ have only unchecked exception.

*  If System.exit() is called in the try or catch block, the JVM will terminate, and the finally block will not be executed.

 ACID(Atomicity, Consistency,Isolation,Durability):

These are the four key properties of data transaction in DBMS.

Atomicity:: Mane suppose amra 100 records k pathachi , so 45 records jaoar por for any reason transaction cancel hye gelo then sei 45 records roll back korbe sender side e tar mane abar prothom theke transaction start hbe.

Money sending is an example.

Consistency:: mane transaction hoar age and pore both sender and receiver side e rules and constraints(NOT NULL, PRIMARY KEY, FOREIGN KEY) must remain unchanged.

Isolation:: 2 to transaction jokhn hbe tokhn eke opor k jeno kono vabe interfere na kore.

For example, suppose 2 different persons are trying to book the last available ticket of a plane , isolatin ensures that one transaction is complete before the other begins.

Durability:: Jodi kono transaction commit kora deoa hoy then system failure(power loss, crash) holeo changes made will be permanently recorded in the databse.

 **Inner Join**: Returns only the intersection of both tables.

 **Left Join**: Returns all records from the left table and the matched records from the right table (NULL if no match).

 **Right Join**: Returns all records from the right table and the matched records from the left table (NULL if no match).

 **Full Join**: Returns all records when there is a match in either left or right table (NULL if no match).

 **Cross Join**: Returns the Cartesian product of the tables (all combinations).

 **Self Join**: Joins a table to itself, typically using aliases to represent different instances of the same table.

Normalization::

Redundancy minimum korar jnno normalization kora hoy.mane divide large tables into smaller, more manageable pieces without losing any data.

1NF::(1 NORMAL FORM)

Atomic valued or single valued attribute, no multi valued attribute.

Mane suppose course column er modhye Jodi kono row te c,c++ thake then course column ta 1nf r modhye porbe na.

2NF::

1nf, no partial dependency, only full dependency.

AB->C(RIGHT)

A->C(WRONG)

B->C(WRONG)

3NF::

2nf, no transitive dependency

x->y->z(wrong)

BCNF(Boyce-codd normal form)::

3nf, L.H.S must be candidate key or super key

4NF::

Bcnf, no multivalued dependency

5NF::

4nf, lossless decomposition(mane suppose amra kono tables records and columns gulo k 2 to different tables e divide korlam then abar Jodi sei tables gulo k combine kori tahole original table r sathe jeno kono difference na ase).

Primary key:

Unique values, can’t contain null values, only one primary key per table.

Candidate key:

Unique and not null values, can be used as primary key, only dfference to primary is that a table can have multiple candidate keys.

Super key:

Unique identification attributes charao unnecessary attributes contain kore.

Every candidate key is super key but not every super key is candidate key as it may conatain not unique values.

Composite key::

Jokhn ekta single column kono row k uniquely identify korte pare na, combinations of two or more columns are required, ei combinations kei composite key bole.

Foreign key::

2 to table r modhye relation establish kore. J table e ei foreign key thake take child table and jate reference key tat hake take parent table bole.

Alternate key:

An alternate key is a candidate key that is not choosen as primary key.

Mane ja bujhlam ta holo ekta table k uniquely identify korar jnno onek gulo unique and not null wala column thaktei pare, eder candidate key bole. tar modhye theke j column ta choose kora hbe uniquely identifying er jnno seta primary key baki gulo alternate key.

Kichu column thake jate uniquely identifying r chara o unnecessary extra rows thake sei column gulo super key and we can say that every candidate key is a super key but not every super key is a candidate key.

Partial key:

Entity relationship model(ER model) e kono entity te nijer primary key na thakle seta k week entity set bole and thakle take string entity set bole, then ei weak entity set partial key r help e strong entity set r primary key r sathe relation establish kore.

1. **Strong Entity Set**: Library
2. **Weak Entity Set**: Book

**Strong Entity: Library**

Attributes:

* library\_id (Primary Key)
* library\_name

**Weak Entity: Book**

Attributes:

* book\_id (Partial Key)
* title
* library\_id (Foreign Key referencing Library)

===========================

Functional Dependency:

How one attribute uniquely determines another attribute.

| **StudentID** | **StudentName** | **ClassID** | **ClassName** |
| --- | --- | --- | --- |
| 1 | Alice | 101 | Math |
| 2 | Bob | 102 | Science |
| 3 | Charlie | 101 | Math |

### Functional Dependencies

* StudentID uniquely determines StudentName:
  + StudentID→StudentName
  + Knowing a student's ID, you can find out their name.
* ClassID uniquely determines ClassName:
  + ClassID→ClassName
  + Knowing the class ID, you can find out the class name.

Kernel::

Core part of OS. It manages system resources and facilitates communication between hardware and software.

Hotel manager er moto kaj kore jekhane guests holo software application and hardware holo hotel stuffs.

Kernel cpu and memory assign kore software application k and ensure they have access to hardware devices.

SDLC(Software Development Life Cycle)::

Planning& requirement, Requirement Analysis, Designing, Coding, Integration & testing, Deployment.

Stack(LIFO-Last in fast out): a stack of book

Queue(FIFO): fast in fast out: railway ticket counter

Class name capital na die start korle kono error raise korbena but it’s widely convensional.

Second highest salary::

**SELECT** max(salary) **as** SecondHighestSalary

**FROM** employee

**WHERE** salary <

(**SELECT** max(salary)

**FROM** Employee)

Singleton pattern ensures that there is only one instance of the class throughout the application's lifecycle.global access peovide kora hoy by using a static method.ete ekta private constructor thake jate class er baire theke instantiation kora na jey.