Control Statements:

Control flow of the code

1. Conditional control statements

If, if else, switch

2. Unconditional control statements

Break, continue, goto

User input: number, decimal,string, char, Boolean

Int maths=24

Int sci

Arrays:-

Naming Conventions:-

Classname:-

Class name must be a noun

Class name should be camel case letter

If class name is single name –

class name starting letter must be capital

If class name has multiple words –

Each word starting letter must be in capital

Method name convetions:

Method name should verb

If the method name is one word – all letters must be a smaller case

Followed by ()

If the method name is multiple word – first word all letters must be a smaller case

Then second word onwards each word-starting letter must be capital case

Followed by ()

Variable

If the variable is one word – all letters must be smaller case

If the variable is multiple word – first word all letters must be smaller case

Second word onward each word starting letter must be capital

If the variable is constant – all word must be capital with underscores

Package name conventions:

All letters must be smaller case

OOP:-

It’s a concept can be applied any of the programming Language

Since Java has incorporated OOP concepts into it that’s the reason java is called OOP language

Classes

Objects

Inheritance

Polymorphism

Abstraction

Encapsulation

Variable:-

Syntax:

AccessModifier Non-AccessModifier Datatype variablename;

Public,private,default,protected

Static

Local Variable:-

A variable declared inside the block method, conditional statement, loop

{

Int a;

}

Instance Variable

A variable declared inside the class outside any method it is called as instance variable

Public class Demo{

String name;// instance variable

int accountNumber; // instance variable

int age; // instance variable

int aadharNumber;//instance variable

public void transaction() {

System.out.println(name);

}

}

Static Variable

A variable declared inside the class outside of any method with static non access modifier keyword it is called as static variable

Class Members:-

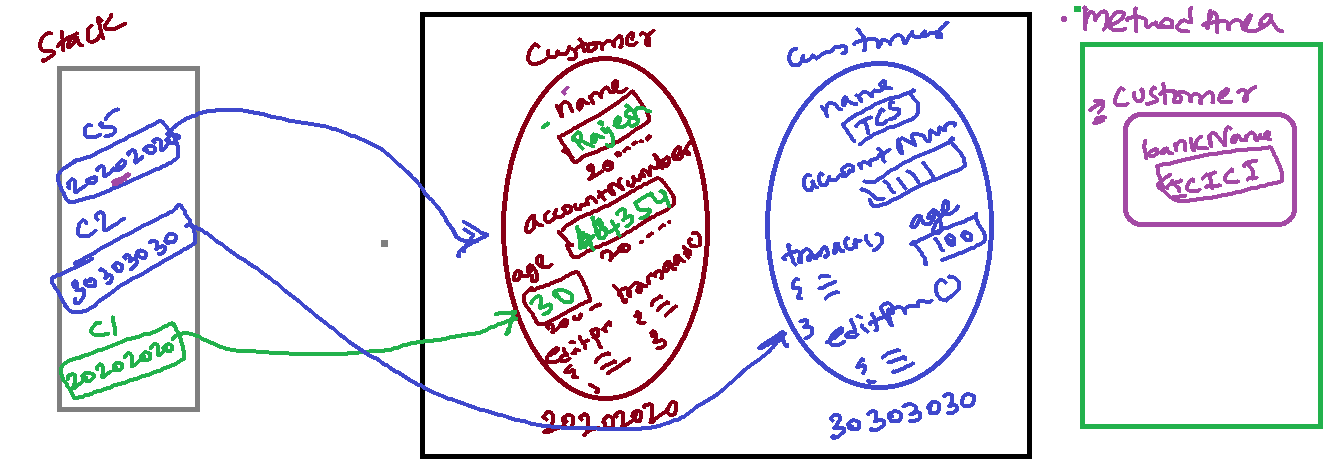
Class Customer{

Variable – instance -static

constructor

method – instance or static

}



Inheritance:-

Class Phone{

Public void call(){

}

Public void sms(){

}

}

Class SmartPhone extends Phone{

Public void camera(){

}

Public void facebook(){

}

}

Categories:-

Is-a Inheritance – by using extends

Lion is a Animal

Has-a Inheritance – By creating object of one class to other class

Car has a Engine

Company Has Employee

Encapsulation:-

* Capsuling data and method together
* Class –
* Access Specifiers –
* 100%

Access Modifiers:-

How long it can be accessible

Visibility

Private , default, protected, public

* + Before the instance or static variable
  + Before the method name
  + Before the class

Non-Access Modifiers

Static

Final

Variable Syntax:

Accessmodifier Nonaccessmodifier Datatype variablename;

Method Syntax:

Accessmodifier nonaccessmodifier returntype methodname(){

}

Packages:-

Collection of classes

Package name:-

IndianEmbessay

PassportIssue.java

PassportVerification.java

Emigration

Visa

Passenger

Payment

Officer

IciciBank

Account

AccountOpening.java

AccountClosr.java

UpdateAddress.java …

Transaction

Withdrawl

Deposit

Cheque

Card

Debitcard

Creditcard

Loan

Insurance

[www.abcbank.com](http://www.abcbank.com) -> reverse the domain

com.abcbank.account

com.abcbank.card

com.abcbank.tranasaction

com.abcbank.insurance

Constructor:

It is a special function

Constructor does not return anything

Constructor name should be same as your class name

Polymorphism

Final

Abstraction

Polymorphsim:-

Poly – many

Morphism – forms

Doing a task in different forms is called as polymorphism

Task – sendParcel

sendParcel(IndiaPost){

statements;

}

sendParcel(courierservice){

statements;

}

sendParcel(myfriend){

statements;

}

sendParecel(myself){

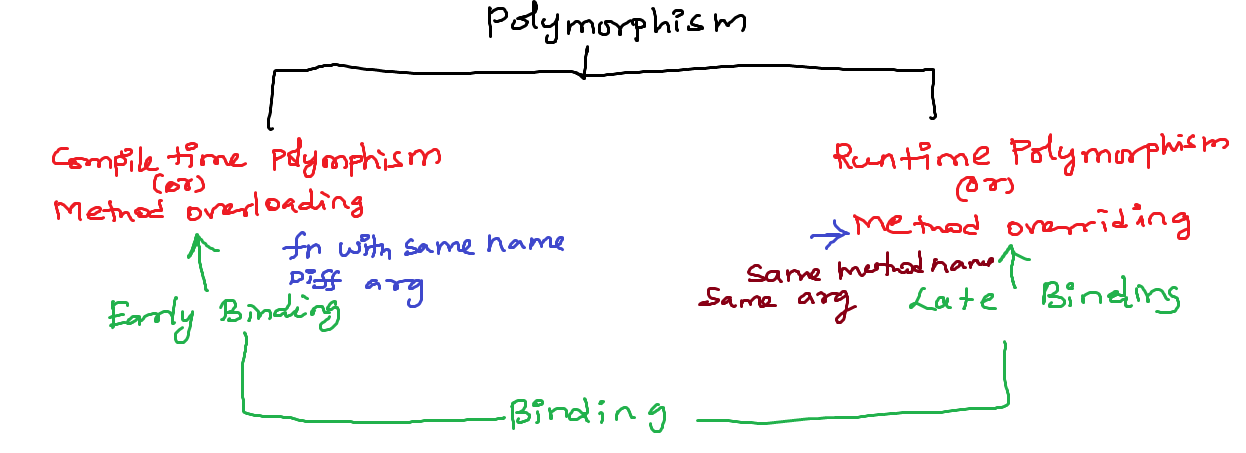
statements;

}

There are type’s polymorphism:

1. Method Overloading

2. Method Overriding



Abstraction:-

Showing essential details hiding complex details

Car – Driver

ATM – customer

1. Abstract class

2. Interface

Implemented method vs abstract method

Public void display(){/x/ implemented or non-abstract method

}

Public void display(); // abstract or unimplemented method

Exception Handling:-

* Is an abnormal condition it arises during program execution.
* If Exception happens in your during execution then its stops its execution abruptly
* If u handle the exception with try catch it terminates the program smoothly and rest of the code also continues

Syntax:

try{

//code possible to create an exception

Scanner s=new Scanner(System.in);

}

catch{

//executed when exception happens

}

finally{

//always executes

//to clean up the resources

}

Types of exception

1. Checked exception

2. Unchecked exception

Collections Framework:-

Collection is a group of objects

Collections used to store and manage group of object by single reference variable name

Collections framework contains set of interfaces, abstract classes and concrete classes

Each collection classes designed with one data structure

Collection framework from java.util package

Data – a useful information

Database – a collection of useful information

tablename

|  |  |  |  |
| --- | --- | --- | --- |
| Columnname1  Datatype(size)  [Constraint] | Columnname2  Datatype(size)  [Constraint] | Columnname3  Datatype(size)  [Constraint] | Columnname4  Datatype(size)  [Constraint] |
| Data | Data | Data | Data |
| Data | Data | Data | Data |
| Data | Data | Data | Data |

employee

|  |  |  |  |
| --- | --- | --- | --- |
| name  varchar (30) | id  int(6) | salary  int(10) | email  varchar(25) |
| Rajesh | 1234 | 20000 | Rajesh.kit@gmail.com |
| Ajay | 56767 | 24555 | ajay@gmail.com |
| Data | Data | Data | Data |

Database management System – It is a software which manages the databases

Relational Database Management System - It is a software which manages the databases, Table inside the database involved in relation (primary key foreign key) then it is called as rdbms

SQL:-

DDL

Create

Alter

Rename

Drop

DML

DCL

TCL

Create a database:

Syntax:

CREATE DATABASE databasename;

Select a database:

USE databasename;

Delete the database:-

DROP DATABASE databasename;

List the database:-

SHOW DATABASES;

List the table from the database selected:-

SHOW TABLE;

Creating a table:-

CREATE TABLE tablename(columnname1 datatype(size),columnname2 datatype(size),….);

To see the table structure:-

DESC tablename;

Read the data from the table:-

SELECT \* FROM tablename;

Adding a row into the table:-

INSERT INTO tablename(columnname1,columnname2,….) VALUES(column1data,column2data,….)

INSERT INTO tablename VALUES(column1data,column2data,….)

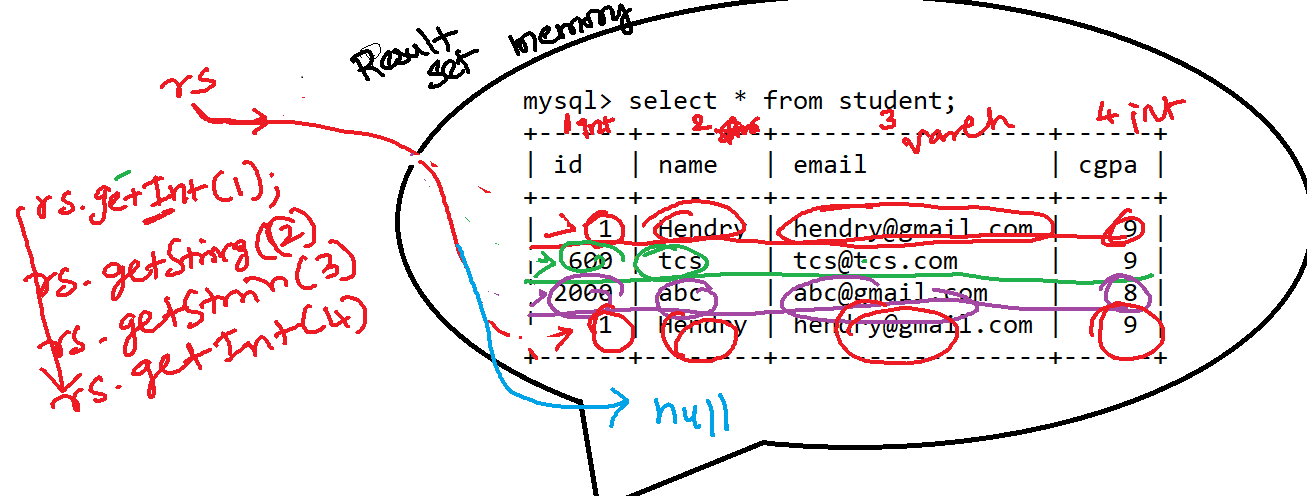
Updating a row data:-

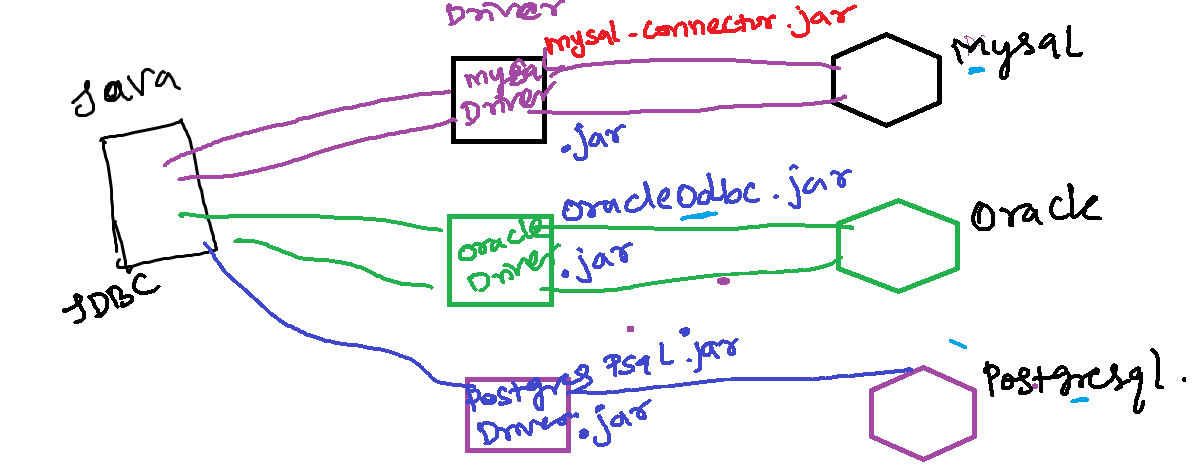
UPDATE tablename SET columnname=newvalue;

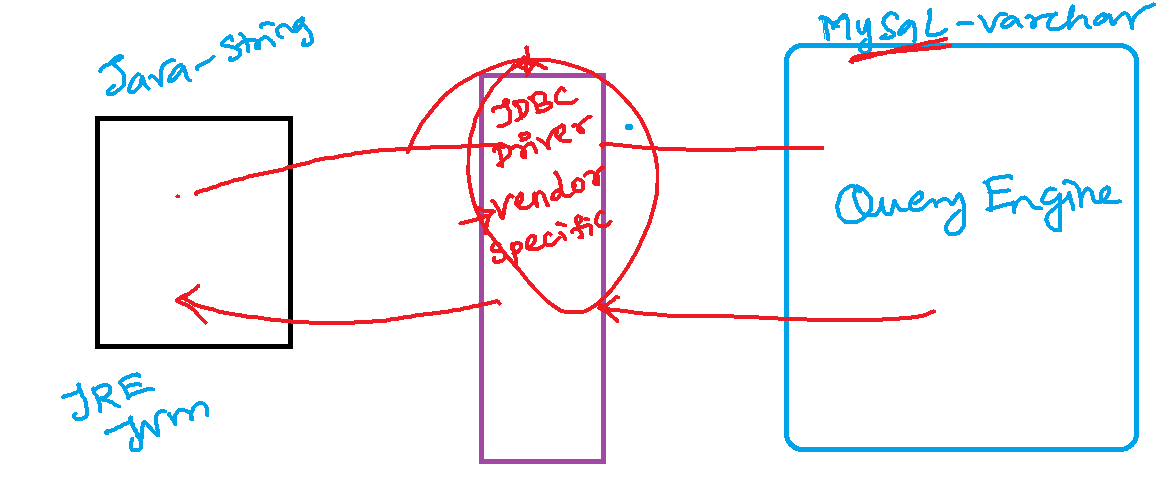
UPDATE loan\_details SET loan\_amount=90000;

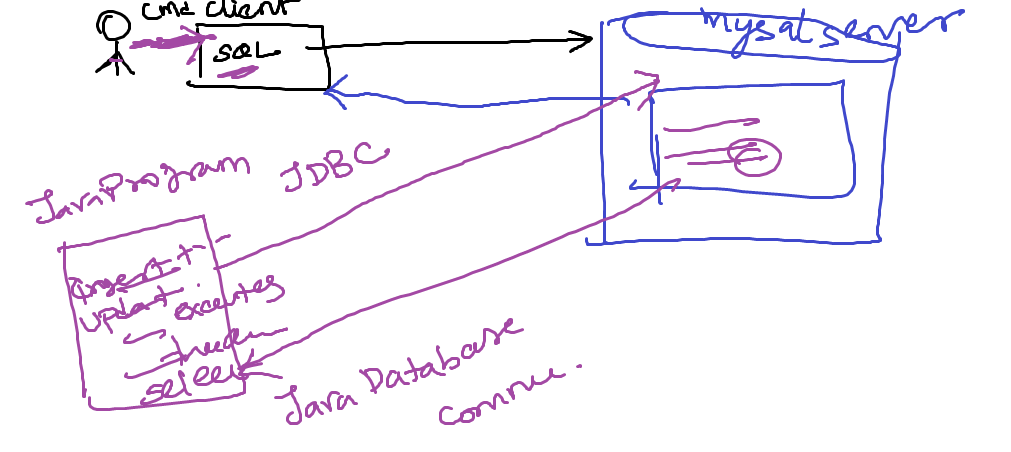
Deleting a row:-

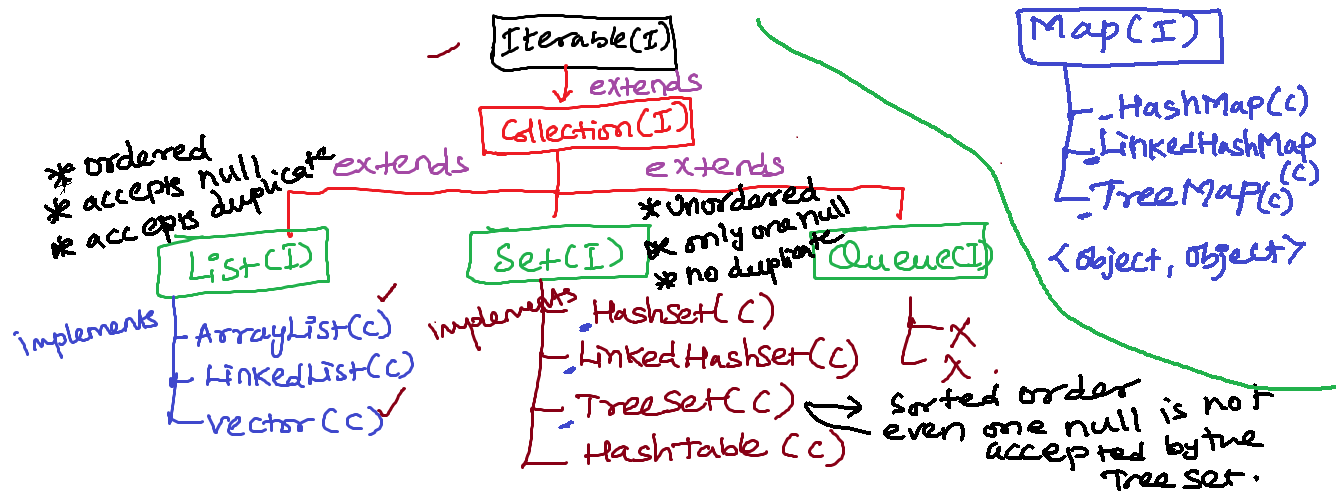
DELETE FROM tablename where columnname=value;

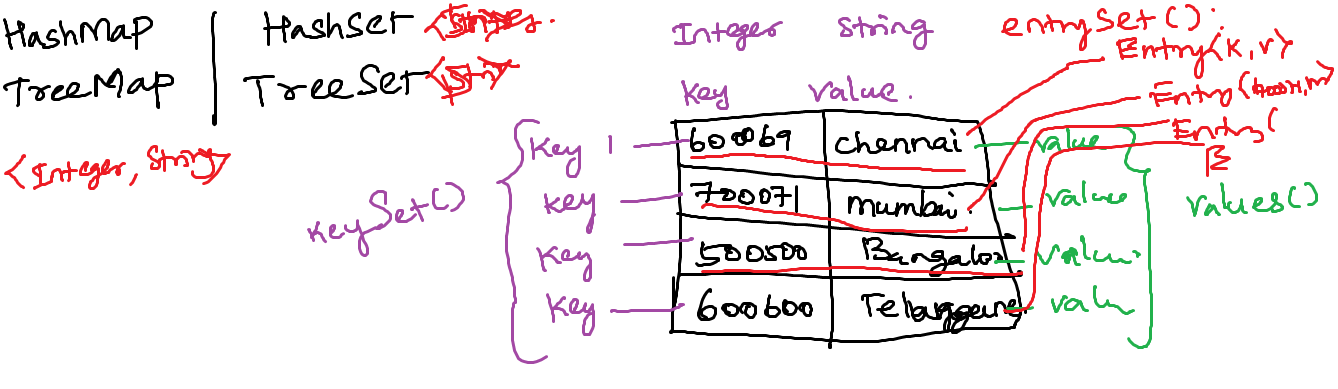


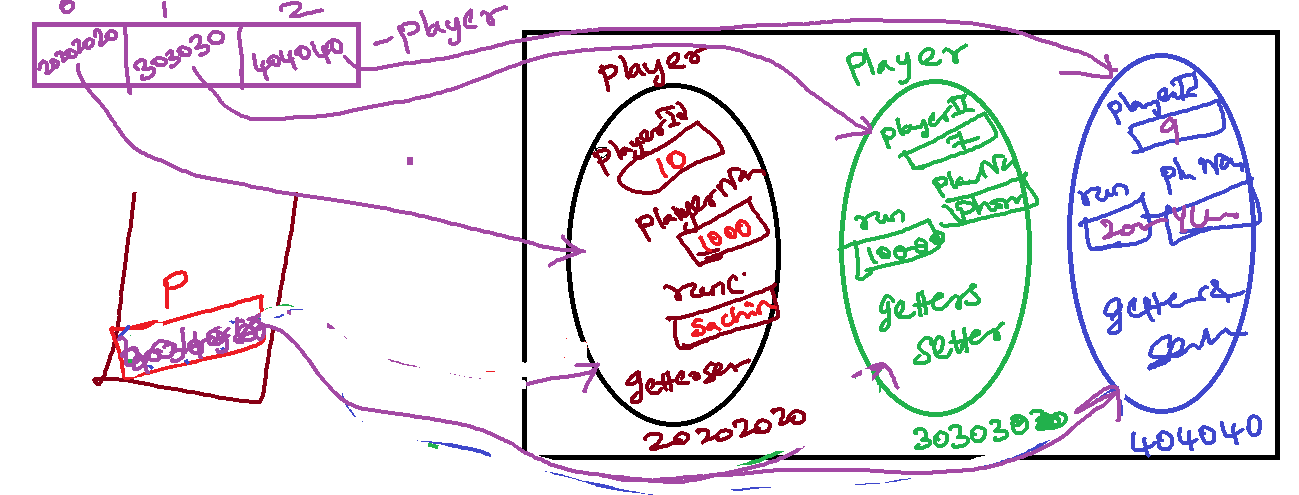


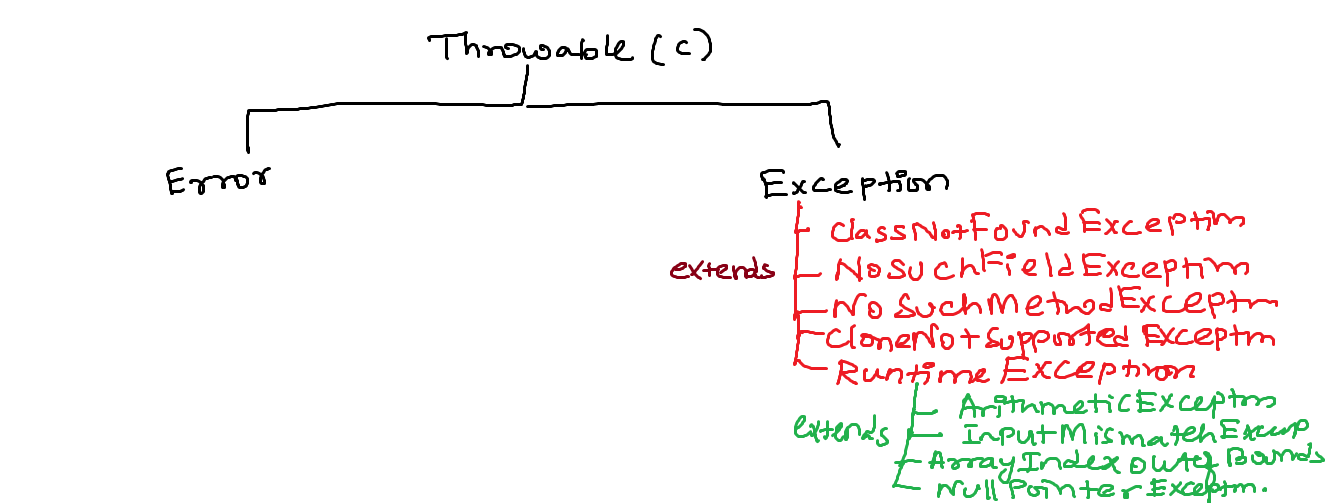


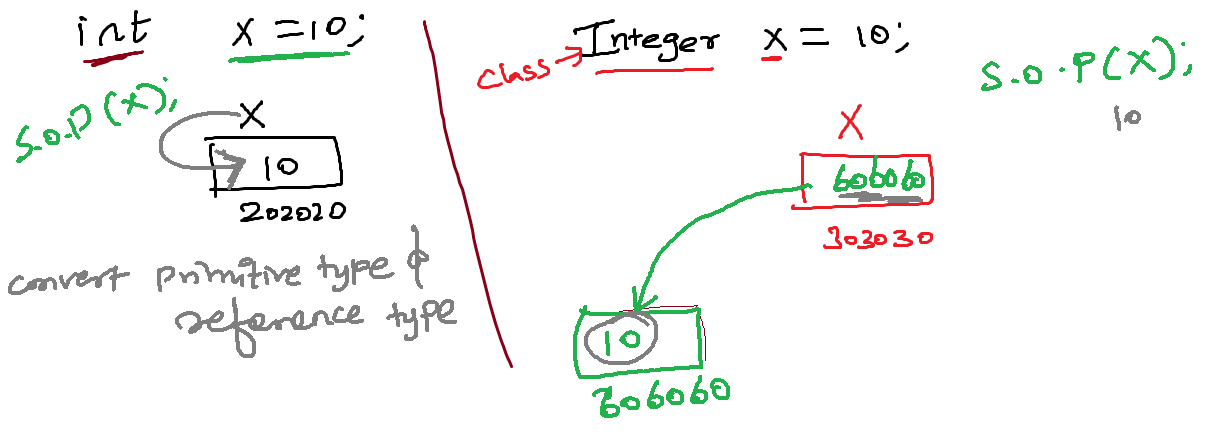












Static query ->

insert into tablename values(‘dfdsf’,’sdfsd’,’fsdf’,’sdf’);

query 🡪 syntax 🡪compiled🡪 Query execution plans🡪least cost execution plan🡪optimaztion🡪 execute

insert into tablename values(‘yyyy’,’ttt’,’www’,’6hfgh’);

query 🡪 syntax 🡪compiled🡪 Query execution plans🡪least cost execution plan🡪optimaztion🡪 execute

Dynamic Query – precompiled query

insert into tablename values(?,?,?,?);

query 🡪 syntax 🡪compiled🡪 Query execution plans🡪least cost execution plan🡪optimaztion🡪 execute

insert into tablename values(?,?,?,?);

Query execution plans🡪least cost execution plan🡪optimaztion🡪 execute

SQL Query Engine:-

query 🡪 syntax 🡪compiled🡪 Query execution plans🡪least cost execution plan🡪optimaztion🡪 execute

Statement -> static query

boolean execute() – DML or DDL commands

int executeUpdate() – INSERT, UPDATE, DELETE

ResultSet executeQuery() - SELECT

PreparedStatement ->

boolean execute() – DML or DDL commands

int executeUpdate() – INSERT, UPDATE, DELETE

ResultSet executeQuery() – SELECT

setInt

CallableStatement ->