Day 1 :

06-01-2023

Full Stack :

Java

Python

.net

MEAN Stack

MERN Stack

Java Full Stack

Phase1

SDLC

Software Development Life cycle.

SDLC Model

Water fall model

Increment model

V model

Agile

Etc

Git : DevOps

Login (SignIn and SignUp) : HTML

Dev1 SignIn

Dev2 SingUp

Git

Manager -🡪

Java Programming

Basic Program

OOPs

Procedure

Object oriented programming

Functional programming

Aspect oriented programming

Phase 1 :

SDLC and Agile

Git

CI/CD

Core Java and data structure

Phase2

Intro to cloud

AWS : Overview : install Jenkin etc.

Into to database mysql or oracle or mongo db.

Phase3

Spring framework or spring boot

Phase4

UI html, css, js angular / react

Phase5

Testing selenium, docker, Kubernetes,

CI and CD

AWS Overview

Program : set of instruction to perform a specific task.

Input read a=10,b=20,sum

Process sum = a+b;

Output write sum

UML

ER-Diagram

SDLC

STLC : Software Testing Life cycle

JSON : Java Script Object Notation

XML :

RDBMS : MySQL table format

No sql : Mongo DB json format

Neo4J graph format

Day 2 :

06-02-2023

Git

Maven

Gradle

Docker

Cloud tech

Etc





Day 3 :

06-05-2023

DevOps

Day 3

Git

Day 4

CI and CD Tool Using Jenkin

Git : Git is distributed sub version control tool which help to record the flow the application.

Git is an open source software.

Dev1 login page.

Dev2 Remote Repository

Dev3

Please login to virtual lab.

Then create the Folder

Then create the file with some contents

Inside a open the terminal to verify git version

git --version

git init This command is use to local folder as local git repository.

ls -a this command display all files and folder including .git folder. (Unix command).

git status This command is use to check current status of local repository

git add filename This command is use to add the file from local machine or folder to

stagging area.

Or

git add . we can add all file and folder present current location.

git commit -m “commit message” this command is use to push the file

from git stagging area to local repository.

git config --global user.email “[akash300383@gmail.com](mailto:akash300383@gmail.com)”

git config --global user.name “akash”

In Same folder create another file with simple message.

In terminal

git status

git add .

git status

git commit -m “task2”

git branch : Branch is like a pointer which hold more than one commit details.

Using git branch command we need to check default branch details

By default branch name can be master or main.

Command to create user defined branch

git branch branchname;

git branch : this command to check all branches present in local repository

git branch A This command is use to create user defined branch

git checkout branchname: This command is use to switch from one branch to another branch

git merge branchname : this command is use to merge user defined branch code to current branch

git merge A

git branch -D branchName

git branch -D A This command is use to delete the branch

if we want to share the code to other team then we need to configure our local repository to remote repository

Remote repository can be

Git hub

Git lab

AWS

Azure

git branch -M main : rename local branch name

git remote add origin URL : this command is use to link local repository to remote repository.

Day 4 :

git remote add origin <https://token@github.com/Kaleakash/test_repository.git>

ghp\_MRDeF2TlL5gakBPq7c1WLIKkVucHxh1Q5iuq

git remote remove origin this command remove remote repository from

local repository

git remote add origin <https://token@github.com/Kaleakash/test_repository.git>

please create separate folder

open the terminal and write the command as

git clone <https://github.com/Kaleakash/test_demo.git>

once you clone test\_demo folder created in your local machine

that folder contains test file.

Next file if you need any new changes then open the terminal inside a folder and write

The command as git pull

git branch -M main

CI and CD : Continuous Integration and Continuous Deployment / Delivery

Remote Repository : GitHub /GitLab/ AWS / Azure etc

Manager

Login page

Dev1 Shared Repository (structure of project present in main or master branch).

application page

Dev2

Feedback page

Dev3

Dev1 create user defined branch with some rules and start the working inside that branch.

git add .

git commit -m “login page created”

git push -u origin branchName;

After dev1 or dev2 or dev3 when we merge the code to actual working code

We need to compile, run , test (build the project).

May be we can build success fully or we can get the error.

We are combining more than one developer code ie integration phase.

Jenkin is an Open source CI and CD tool created using Java technology or language.

Java is an open source. Jenkin is a plugin base CI and CD.

Java : Java is an open source, pure object and platform independent programming language.

Java introduce in nov 1995. Initial name of the Java is Oak.

OOPs : Object Oriented programming

Object : object is any real world entity.

Property or state -🡪 have (data type int, float, char, string,boolean)

Person

Behaviour 🡪do/does 🡪 action 🡪 teaching(), listening(), sleeping(), eating() etc

Bank

Wheel(int), colour(string), price(float) etc -🡪 have

Car

Start(), appliedGear(), moving(), stop() etc 🡪 behaviour

name,

Employee

Customer

Etc

Online application :

Pid,pname,price,qty etc.

Product

addProject, updatePrice, deleteProduct, viewProduct etc

viewProduct , orderTheProduct etc.

Order

Account

Login

Manager

Customer

etc

object is concept.

class : class is known as blue print of object or template object or it is user defined data type which help to create the object.

Class syntax :

class Car {

property

behaviour

}

class Car {

int wheel;

String colour;

float price;

void start() {

}

void stop() {

}

}

Car innova = new Car(); car class object created.

Car bmw = new Car(); another class object created. (memory created);

06/12/2023

Eclipse is a type of Open source IDE (Integrated development Environment ) . it is open source

Tool which help to write java program.

Package is a collection of classes and interfaces.

Package is just like a directory or folder which contains more than one class as well as interface.

Data types

Data type is a type of data which tells what type of data it can hold.

In java data types are divided into 2 types.

1. Primitive type : it is use to store only value.

8 primitive types

byte 1 byte -127 to 128 (range number)

short 2 byte

int 4 byte

long 8 byte without decimal

float 4 byte

double 8 byte with decimal

char 2 byte any single character

boolean 1 bit true or false value

1. Non primitive type or reference type. :it is use to store value as well as reference of another data types.

Variable : variable is a name which hold some value.

Syntax

datatype variableName;

int a; declaration

int b=20; declaration with initialization.

double b;

boolean result;

a=10;

a = true;

a=”akash”;

Operator : operator are use to do some operation on variable.

Arithmetic Operator : +, -, \*, /, % modules (remainder)

Assignment : int a=10;

Conditional operator : >, >=, <, <=, ==, !=

int a=10;

int b =20;

boolean res;

res = a>b;

increment and decrement

++ --

int a=10;

a=a+1;

System.out.println(a);

a++;

System.out.println(a);

a--

System.out.println(a);

Conditional statement : it is use to execute set of statement base upon condition.

1. If statement

If(condition) {

True block

}

1. If else

If(condition) {

True block

} else {

False block

}

1. If else if

if(condition1) {

true block

} else if(condition2) {

True block

}else if(condition3) {

True block

}else {

False block

}

Switch statement

In switch statement user can take the decision which block we need to execute.

Syntax

switch(variableName) { variable can be int family or char no float/double

case value1: block1;

break;

case value2: block2;

break;

case value3: block3;

break;

case value3: block3;

break;

default : defaultblock;

break;

}

Taking the value through keyboard in Java

In java we can take help of Scanner class.

Scanner is a pre defined class which provide set of method

Which help to take or scan the value through keyboard.

Syntax to create Scanner class object

Scanner sc = new Scanner(System.in);

This Scanner pre defined class part of util package.

We need to import that package while using scanner class.

Outside a class

We need to import

import java.util.Scanner;

Looping

Looping is use to execute the task again and again till the condition become false.

Initialization : start and end position i=1,n=10

Condition it must be true i<=n or i>=n

Do the task System.out.println(“Hello”);

Increment or decrement i++ or n--

While loop

int i=1,n=1000;

while(i<=n) { entry loop

System.out.println(“Hello”);

Syste.out.println(i);

i++;

}

Do while loop : this loop execute at least once doesn’t matter condition true or false.

int i=1,n=1000;

do{

System.out.println(“Hello”);

Syste.out.println(i);

i++;

}while(i<=n); exit

For loop

Array : Array is reference or non primitive data type which help to store more than one

Value of same type.

int a=1,b=2,c=3,d=4;

array declaration

datatype arrayName[];

int num[]; it allow to store more than one value of type int.

int n;

int num[]={10,20,30,40,50}; this num variable hold five values.

int a=10;

array follow index concept. The index value start from zero position.

System.out.println(num); // That variable address.

System.out.println(num[0]);--🡪10

System.out.println(num[1]);--🡪20

a=20;

sid,sname,sage

int sid[];

int sid;

to retrieve the value from array we use for loop

syntax

1 2 4

for(initialization ; condition; increment/decrement) {

body of the loop; 3

}

Array memory creation

datatype arrayname[]=new datatype[size];

int abc[]=new int[10]; default value for int array is zero.

System.out.println(abc[0]); 0

System.out.println(abc[9]); 0

System.out.println(abc.length); 10

Method or functions

Method or function is use to write set of instruction to perform a specific task.

Using function or method we can do re-usability of code.

Syntax

accessSpecifiers returnType methdoname(parameterList) {

}

Access specifiers optional in method.

No passing parameter as well as no return type.

void display() {

set of code.

}

void is a keyword no return any value to caller method.

char c=’b’;

String means combination of more than one character enclosed in double quote.

In Java store more than one character java provided pre defined class ie String.

String :

Syntax to store the value in String class

String name1 = “Akash”; //literal style

String name2 = new String(“Akash”); // memory creation style using new keyword.

object : object is any real world entity which have properties and behaviour

Car

Employee

Person

Bank

Animal

class : class is blue print of object or template of object.

``

Day

Date : 06/15/2023



Constructor : Constructor is a type of special method. Which help to create the object in all OOPS language.

Pts

1. Constructor have same name as class itself.
2. Constructor doesn’t contain any return type not even void also.
3. Constructor get call automatically when we create the object of that class. But for method we need to call explicitly to execute that code.

If we not write any constructor in program inside a class by default JVM (Java Virtual Machine)

Provide default constructor which doesn’t contains any logic it only help to create the memory. If we want to do any custom logic when memory created like initialization then we need to write empty constructor. If we write explicitly any constructor then JVM doesn’t provide any default constructor.

Parameterized constructor : constructor can take 1 or more than one parameter.

In the life of the object if we want to do any task only one time that type of task we need to write inside an empty or parameter constructor.

If the life of the object if we want to perform any task again and again that type of task we need to write inside method.

JDBC : Java Database Connectivity : using this concept with core java as well web application we can store as well as retrieve data from database ie mysql or oracle or mongo db etc.

ORM : Object Relation Mapping : Hibernate or JPA.

Encapsulation : Binding or wrapping data (variable) and code (function or methods) in single unit is known as Encapsulation.

In Java or all oops language we bind variable and method inside a class. no one access our variable and method directly without creating object of my class.

class is good example of encapsulation.

To make our data secure we need to add private key for all instance variable.

If we declare the instance variable as private we can’t change as well as can’t access instance variable

Value outside class through object.

vodafone.banking.com.bean

com.bean

Inheritance : Inheritance is use to inherits properties and behaviour of old class to new class.

class OldClassName { super class or base class or parent class

properties

behaviour

}

class NewClassName extends OldClass { // sub class or child class or derived class.

property

behaviour

}

class A {

void dis1() {}

}

class B extends A {

void dis2() {}

}

A obj1 = new A();

obj1.dis1();

B obj2 = new B();

obj2.dis2();

obj2.dis1();

Types of Inheritance

1. Single inheritance : one super class and one sub class

class Employee { } oner super class

class Manager extends Employee { } one sub class

1. Multilevel Inheritance : one super class and n number of sub class connected one by one

class Employee { } super class

class Manager extends Employee { } sub class

class ProjectManager extends Manager { }

1. Hierarchical Inheritance : one super class and n number of sub classes directly connected to super class.

Class Employee { }

Class Manager extends Employee {}

Class Developer extends Employee {}

Class Tester extends Employee {}

1. Multiple inheritance : more than one super class and one sub class.

Class Father {}

Class Mother {}

Class Child extends Father, Mother {} Error in this line multiple inheritance java doesn’t support using class it will support using interface.

class Employee {

id,name,salary with methods.

}

class Manager extends Employee {

numberOfEmp with methods.

}

class ProjectManager extends Manager {

clientDetails and methods

}

class Developer extends Employee{

project name and with some methods .

}

Day

Date : 06/22/2023

Polymorphism : One name many forms or many implementation.

2 types

Compile time polymorphism : javac Demo.java

Static binding or early binding

Example : Method Overloading

Method Overloading : Method have same name but different parameter list (ie type of parameter list or number of parameter list must be different)

Run time polymorphism : java Demo

Dynamic binding or late binding

Example : Method overriding

Compile and interpreter

Compiler convert whole code in another format like byte code in java.

Interpreter check covert the code line by line

Method overriding : Method have same name and same method signature number of parameter list, type of parameter list and return type must be same.

To achieve method overriding we need to use inheritance. Without inheritance we can’t achieve method overriding.

void info() {

logic different

}

void info() {

logic different

}

Super class must be generic

Sub class must be specific

Class Bike

Class Honda extends Bike

Class Activa extends Honda

Car

Bank

abstract , final and static keyword

abstract keyword :

1. abstract is a keyword we can use with method and class but not with variable.
2. abstract method : method without body or incomplete method or without curly braces is known as abstract method.

abstract returnType methodName(parameterList);

abstract void withdraw(); // method declaration

1. if class contains abstract method ie one or more that class we need to declare as abstract class.

abstract class className{

}

1. whichever class extends abstract class that class must be provide the body for all abstract method belong to super class.
2. abstract class can contains normal as well as abstract method.(it can contain 1 or many or zero).
3. If class is abstract we can’t create the object of that class.

final keyword :

1. Final keyword we can use with variable, method and class.
2. Final variable : to declare a constant value we use final keyword.

final double PI=3.242;

PI=12345; error

final int A=10;

A=20; error

int b=30;

b=50;

1. final method : if method is final sub class can’t override that method. Final method in sub class we can use it but can’t override it.
2. final class : if class is final we can’t inherits that class means we can’t make sub class for final class.

static : static keyword we can use with variable and method but not with class.

1. if variable is static we can call or access that variable with help of class name. object is not required.
2. If method is static we can call that class method with help of class name object is not required.
3. Static variable we can access through object also.
4. Static metho we can access through object also.



06/23/2023

Abstraction : hiding the internal implementation without knowing background details.

String name = “Welcome to Java Training”;

name.toUpperCase();

interface : interface is a reference data type. Interface also known as 100% pure abstract class.

syntax

interface interfaceName {

variables;

methods;

}

In interface by default all variables are

public static and final

by default all method interface are public and abstract.

interface Abc {

public static final int A=10;

int B=20;

public abstract void dis1();

void dis2();

}

interface Abc {

int A=10;

void dis1();

}

interface Xyz {

int B=20;

void dis1();

void dis2();

}

interface Mno extends Abc,Xyz{

int C=30;

void dis3();

}

One interface can extends another interface but interface can extends more than one interface. But class can extends only one class.

class Demo implements Abc,Xyz {

dis1() and dis2() provide the body.

}

We can’t create the interface object. Because it contains abstract method.

Interface is use to provide to specification and class provide implementation for that interface.

Package :

Package is a collection of classes and interfaces which have same name but different purpose use.

Package is just like directory or folder which we are creating more than one classes or interfaces which have same name.

User defined package

Education

School College

Attendance Attendance

education.school education.college

Attendance.java Attendance.java

Access Specifiers

Java provided 4 types of access specifiers which help to expose the visibility or accessibility of

Variable, method and class from same package or other package.

private : we can use with instance variable, static variable, non static method, static method, constructor but not with local variable and class.

scope : within a same class.

default : we can use with all (no access specifiers means default access specifiers)

scope : within a same package

protected : we can use with instance variable, static variable, non static method, static method, constructor but not with local variable and class.

scope : within a same package other package if sub class.

public : we can use with instance variable, static variable, non static method, static method, constructor, class but not with local variable.

Scope : we can access in same package as well as other package

By default every Java program it can be pre defined or user defined. Internally extends Object class.

Exception handling

Exception : In Java Exception is pre defined class part of lang package.

By default every java program imported lang package.

Exception is an object or memory which get created when unexpected or abnormal things

Happen during the execution of program.

IDE : Integrated Development Environment like Eclipse , My Eclipse, JDeveloper etc.

Java program

Compile program Run the program

Compile time error run time error

Syntax error

class Demo {

void dis() {

System.out.println(“Welcome”)

}

}

class Test {

}

Run time error

Error Exception

Error and Exception both are pre defined classes part of lang package.

Error :The error which generated at run time which we can’t handle it.

JVM Crash, software or hardware issue, out of memory etc.

Exception : it is a type of run time error which we can handle it.

Divided by zero. 10/0;

Array index out of bounds exception.

int num[]={10,20,30,40,50}; num[0], num[1] till num[4] num[6]

Exception

Checked Exception Unchecked exception

RuntimeException

SQLException ArrayIndexOutOfBoundsException

IOException ArithmeticException

FileNotFoundException NumberFormatException

All unchecked exception classes internally extends RuntimeException class.

All checked exception directly extends Exception class.

To handle both the type of exception java provided 5 keyword.

try

catch

finally

throw

throws

try – catch block

try {

try block

}catch(Exception e) {

Catch block

}

7/10/23

In Virtual Lab

Create Empty folder

Then open terminal

git clone URL

Please Create Folder with names as JavaTrainingNotes in Virtual lab.

Then open the terminal inside that folder

Inside this folder you keep your java programs, notes files other files.

Examples

Copy workspace folder as well as others

from akash git clone repository into JavaTrainingNotes.

git init (this command is use to make JavaTrainingNotes folder as local repository folder)

git add . (this command is use to add all files and folder into stagging area)

git commit -m “akash notes copied” (send all files and folder from staging area to local repository folder)

git status

In

please logic in git hub account using browser.

git branch -M main this command is use to rename branch from master to main

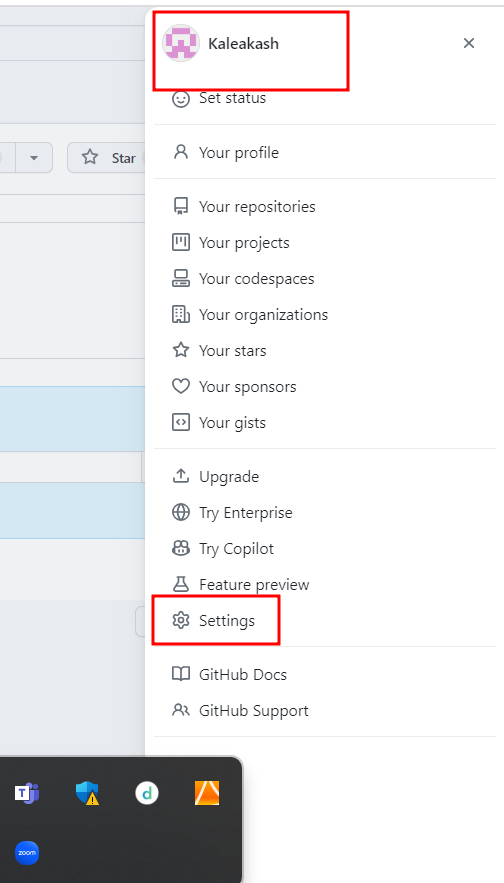
copy this command from remove repository screen and paste it but don’t hit as of now.

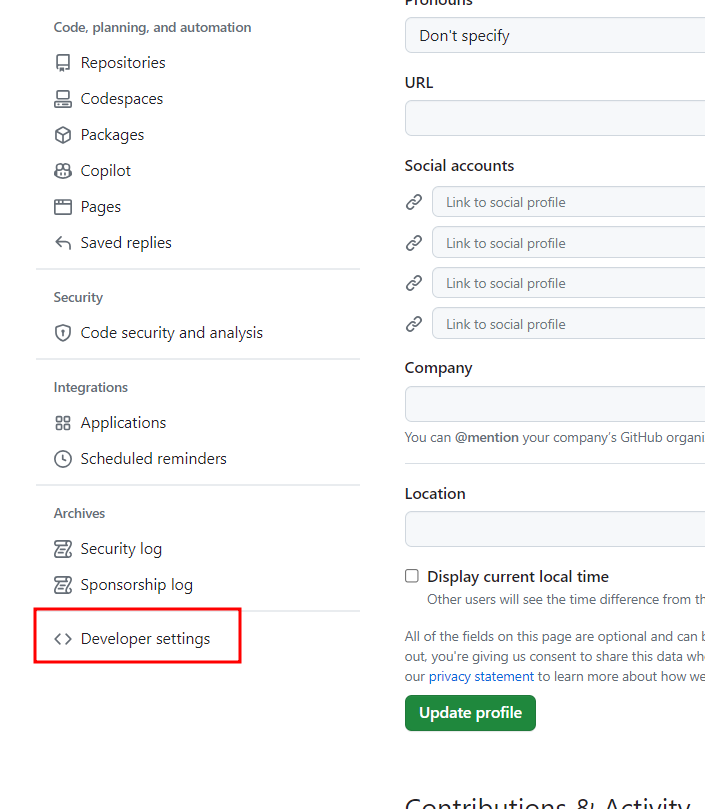
git remote add origin https://github.com/Kaleakash/java\_notes.git

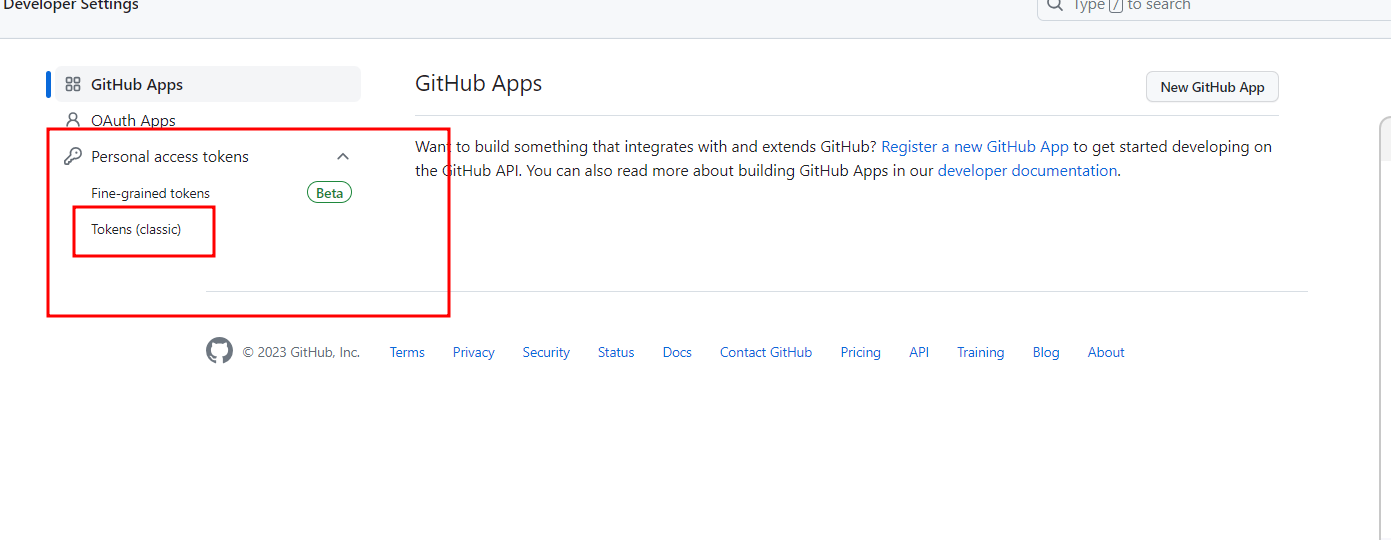
creating token

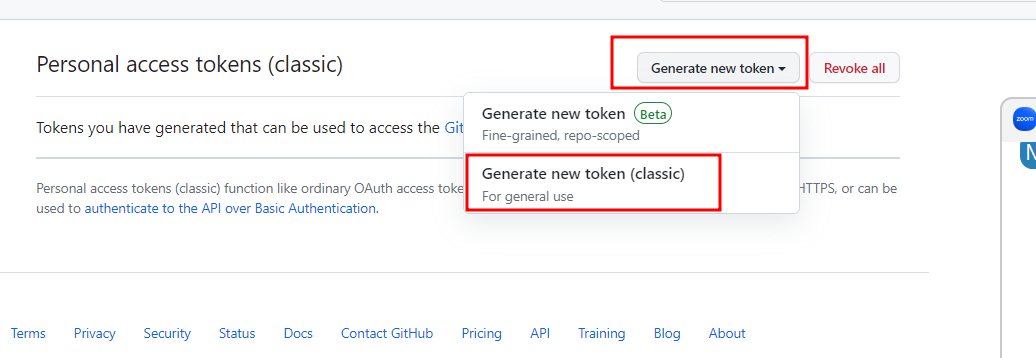
Exception Handling

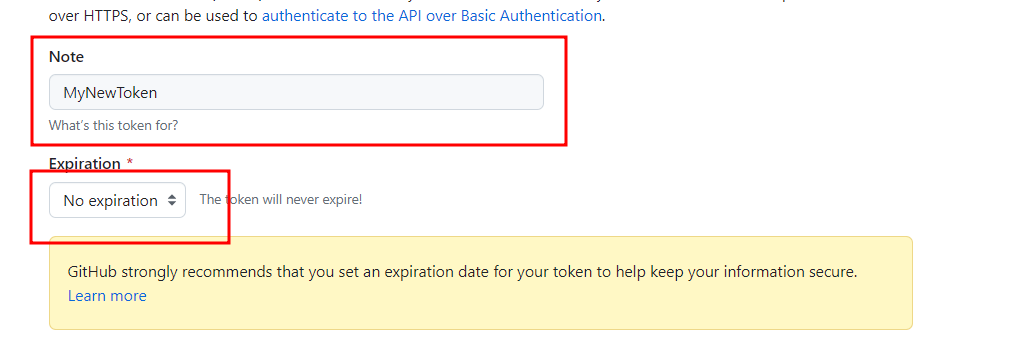
Right corner in git hub dashboard

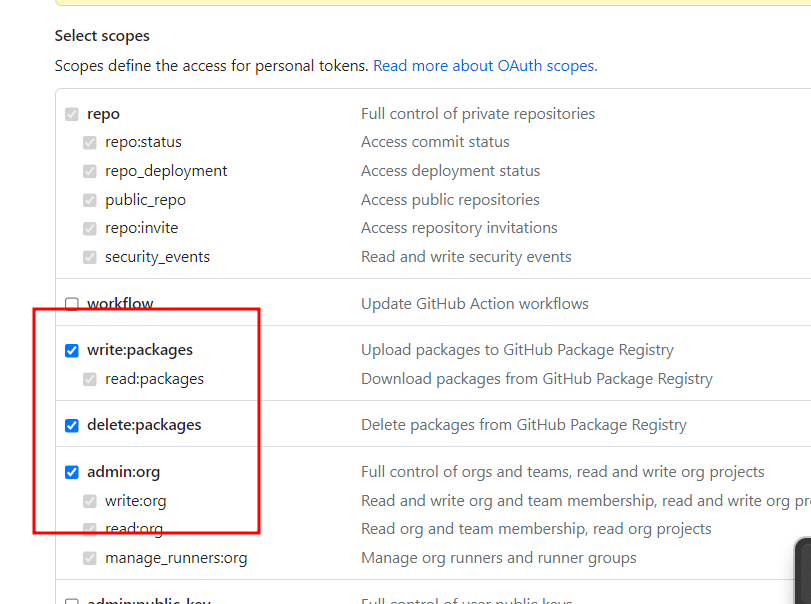


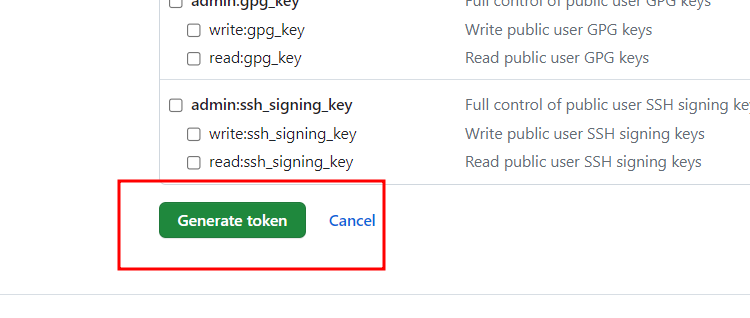


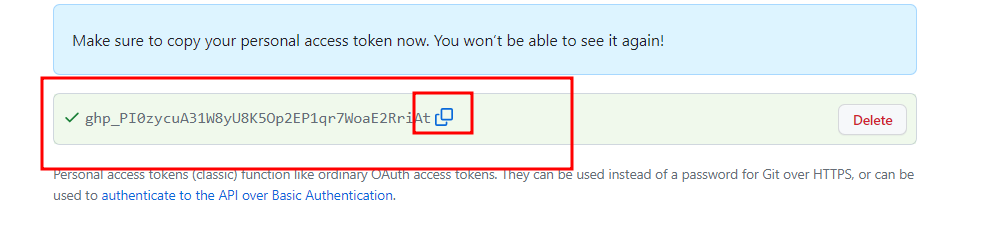












Copy the token

Open the terminal and paste the token after https://

Git remove add origin https://token@github.com/Kaleakash/java\_notes.git

git remote add origin https:// ghp\_PI0zycuA31W8yU8K5Op2EP1qr7WoaE2RriAt @github.com/Kaleakash/java\_notes.git

this command link your local machine terminal to remote repository.

git push -u origin main (this command is use to push local repository code to remove repository).

**Exception handling**

To handle the exception using try catch block

try {

}catch(Exception e) {

}

The code one line code or more than one line code which generate the exception that code we need to keep in try block

Try with single catch block.

If any exception generate we want to do common task need to perform

If we don’t know exception class name

That time we need to use try with single catch block.

try{

code

}catch(Exception e) {

}

Try with multiple catch block

Base upon exception we need to perform different task then we need to use try with multiple catch block.

try{

}catch(ArithmeticException e) {

Specific for divided by zero

}catch(ArrayIndexOutOfBoundsException e) {

Wrong index position

}catch(Exception e) {

Apart from top 2 any other exception then this catch block.

}

07/11/2023

try : the code which generate exception ie one line code and more than one line code we need to keep in try block.

catch : This block is known as exception handler. If any exception generate then this block get execute. No exception no catch block.

finally block : This block execute 100% sure if any exception generate or not.

try {

code that can generate the exception

}catch(Exception e) {

It will execute only if any exception generate.

}finally {

This block execute compulsory doesn’t matter if any exception generate or not.

}

We can write try with finally without catch block.

try{

}finally {

}

Finally block is use to close the resources doesn’t matter exception generate or not.

File handling

Database resources

Security resources

throw : throw keyword is use to generate or raise pre defined or user defined (custom exception)

base upon the conditions.

Syntax

throw new Exception();

or

throw new ExceptionSubClass();

Checked exception check at compile time as well as run time.

Checked exception we can’t avoid it but un checked exception some extend

We can avoid it.

Un checked exception check only at run time.

Throws keyword we write with method signature.

Syntax for throws

void display() throws Exception{

}

Throws keyword we use with method to throw the exception to caller method.

Introduction to Multi threading

Program : set of instruction to perform a specific task.

Processor : processor is responsible to execute the code or run the code.

Process : program in execution or time taken to execute the code.

Thread : thread is a small execution of code within a process.

By default in process in java one default thread get execute.

Thread

Process

Process generally heavy weighted. Thread are light weighted.

Heavy weighted means it takes more time or more resource to execute the task.

Thread take less resources or less time to execute the code.

Thread also known as light weighted process.

First program to check default thread details in java program.

In java to working threading concept java provided pre defined class Thread. Part of lang.

By default every java program import lang package.

By default java is thread base programming language by default one thread always run inside a main method.

Thread t = Thread.currentThread();

currentThread is a pre defined method part of Thread class and it is a static method.

And method return type is Thread class reference.

System.out.println(t); Thread[main,5,main];

main -🡪 name of thread by default is main.

5🡪 priority of thread by default 5.

main 🡪group of the thread main.

We can set the priority within a range to 1 to 10

We can’t set less than 1 or more than 10.

By default 5.

Multi threading :

Creating more than one thread to do different task.

Multi tasking

1. Process base: By default, C or C++ are process base
2. Thread base : By default java is thread base

Printing the paper must one by one.

Copy set of files or folder from one location to another location.

Thread will execute very fast and they share same memory

Multi tasking using thread base is faster than multi tasking using process base.

In java we can create more than one thread using two ways

1. Extends Thread class
2. Implements Runnable interface

Extends Thread class

1. We need to create user defined class and that class must be extends Thread class.
2. In main class create user defined class object ie consider as Thread class reference indirectly.
3. With help of reference call start() method. Start() is pre defined method which is to make the thread to run. This method is part of Thread class.
4. Start() method internally call run() method of Thread class. run() method of thread class contains empty body.
5. So if we want to do custom logic then we need to override run method part of thread class in A class as well as B class.

Implements Runnable interface

1. We need to create user defined class and that class must be implements Runnable interface.
2. Runnable interface contains one abstract method ie run() methods.
3. When class implements Runnable interface we need to override run method mandatory.
4. Now create the user defined class object. Using A or B class we can’t call start() method

but to give the life for the thread we need to call start method.

1. Create Thread class reference with pass the parameter of that class which implements runnable interface.

07/12/2023

Synchronization

It is use to lock or block the thread. it allow to use all resources for only one thread at time.

To achieve synchronization in java we use synchronized keyword.

This keyword we can use with method or inside a method we can use more than one synchronized block.

Inner thread communication

In Java using wait(), notify() and notifyAll() we can make the thread to wait (suspend)

Notify(resume) with conditions.

Consumer and Producer

Consumer and producer are two thread.

Producer

Consumer

Producer

Consumer

When we want to use wait(), notify() and notifyAll() method

These three methods throws checked exception. So we need to use try -catch mandatory.

Method must synchronized.

Data Structure : Collection Framework

Data structure is use to store the data in proper manner so we can access those data very easily

According data structure while accessing the data from a container. We need to keep the track

Time complexity and memory.

In Java Collection framework provided set of classes and interfaces which help to store the data, search, sorting etc.

Algorithms : set of rules to perform a specific task.

Variable

int a=10;

a=20;

array : it is use to store more than one value of same type.

Int abc[]=new int[10];

In this abc we can store 10 number of type integer.

Limitation of array memory.

Array is type of data structure which is known as fixed in size.

After stored element in array if we want to insert any element in array in between more complex

Int num[]={100,200,300,400};

Num[0], num[1],num[2],num[3]

100,1000,300,400

Structure : it is use to store more than one value of different types. Java doesn’t support structure. It support by C or C++ etc.

class Employee {

int id;

String name;

float salary;

}

Employee emp = new Employee();

emp.id=100;

emp.name=”Ravi”;

emp.salary=12000.50;

array object

Employee []employees=new Employee[100];

Employees is a reference name which can hold 100 object of employee type.

Int abc[]=new int[100];

Array of object or array of integer, float,string

Array is known as fixed memory size, adding element, remove element more complex.

Collection Framework : it contains lot of classes and interfaces which internally connect to each others which allow to store any types of value, it provides set of methods which help to do

Add, remove, search, sort, iterate very easily with help pre defined methods.

Collection framework is part of util package.

Collection Framework hierarchy

Collection -🡪 interface top interface

Set List Queue Map

All are interfaces

Set, List, Queue internally extends Collection interface but Map doesn’t extends.

Set : it is use to store set of items(number types, string, float, Boolean, user defined object).

Set classes

HashSet Unorder same or different

LinkedHashSet Order same or different

TreeSet Sorted by default ascending order. In tree set we can store

Same type of values.

These three classes directly or indirectly implements Set interface.

Set doesn’t allow duplicate. In set elements can be unorder, order or sorted.

List : it allow duplicate. It maintains the order using index position.

Stack

ArrayList

LinkedList

Vector

These four classes type of list class mean directly or indirectly implements List interfaces.

Queue : FIFO :First In First Out. It allow duplicate and maintain order as well as unorder.

LinkedList

PriorityQueue

These two classes type of queue classes internally implements Queue interface.

Map : it allow to store the information in key-value pairs. Key must be unique and value can be duplicate.

HashMap ­

LinkedHashMap

TreeMap

Hashtable

These four classes type of Map which internally implements Map interface.

Set doesn’t allow duplicate.

HashSet class : it is a type of Set class HashSet maintain the elements in unorder manner.

LinkedHashSet : LinkedHash internally extends HashSet and it maintain orders. It doesn’t provide any extra methods.

TreeSet : TreeSet class internally implements SortedSet interface. That interface internally extends Set interface. SortedSet interface provide the logic to do sorting by default in ascending order.

In TreeSet we need to store only same type of data types values.

TreeSet provided few extra methods which is not present in HashSet or LinkedHashSet.

List API(Application programming interface)

Stack : Stack is a type of Data structure which follow First In Last Out.

Or Last In First Out

Method invocation internally follow stack concept.

07/13/23

ArrayList : ArrayList isa type list class which provided set of methods which help to add, remove, search data from ArrayList very easily.

ArrayList Vs Generic or Normal Array of int, float, string etc

1. ArrayList by default allow to store any types of values but normal array allow to store only same type of values.
2. Normal array fixed in memory size. ArrayList is dynamic in memory size.
3. Normal array doesn’t provide any pre defined method to add, remove, search we need to write the code. But ArrayList provided set of methods we can add, remove, search element very easily.

ArrayList index position start with zero.

LinkedList : LinkedList is a type of data structure which internally use node concept to store the value.

ArrayList use index position to store the value.

If we store any value in linked list it create node.

Value reference

Pre ref value next ref

3 type of linked list

Single linked list

10 ref 20 30

Double linked list

Pre v next p v n p v n

Circular linked list in this linked list first node as well as last node connected each others.

By in Java LinkedList consider as double linked list

In ArrayList if we do more insertion and deletion if will effect the performance.

In LinkedList we can add and remove elements very easily with property performance because we need add node in between or remove the node in between.

int abc[]={10,20,30,40,50};

abc[0]

retrieve the element from ArrayList is faster than LinkedList because ArrayList use index position concept.

Vector : It is a type of legacy class (old class). In Vector by default all methods are synchronized.

So vector class is thread safe class when we use multi threading concept.

But performance wise slower than LinkedList and ArrayList. But safe in work.

Queue : Queue First In First out by nature.

PriorityQueue

LinkedList

Map: Map allow to store the value in key-value pairs.

Key is unique and value can be duplicate.

HashMap : display the element unorder

LinkedHashMap : maintain the order

TreeMap : Ascending order as key. So key must be same data type.

Hashtable : by default all methods are synchronized. Thread safe but slow in performance.

Retrieve the elements from collection one by one

1. For each or enhanced loop.
2. Using Iterator : Iterator is a interface which provided set of method which help to retrieve

Value one by one.

07/14/2023

Collection Framework Utilities classes

Arrays and Collections

These two classes provided set of methods which help to do some operation on primitive array ie Arrays and List API ie Collections like sort, search etc.

int num[]={2,5,1,8,6,9};

Arrays.sort(num); // sort by ascending order

1 option to use logic to do sorting

2 option convert normal array to list and Collections provided sort() then reverse() List : ArrayList or LinkedList or Vector.

Normal Array ie primitive array of type int, float, string , double

ArrayList : type of list class

Arrays : type of utility class.

If we want to search the data present or not in array then

Arrays class provided pre defined method binarySearch.

**Design Patterns, Secure Coding and coding standards**

**Day 1 :**

**07/17/2023**

Design pattern: Design pattern is known as best practise or solution of repeating problems.

Design pattern it is a concept.

While developing any application we need to follow separation of concern.

Package : collection of classes and interfaces

Class as well as interfaces

Collection of methods

Method : set of instruction or code.

We need to give proper name for the package and package name must be lower case.

Your project name must be meaningful name.

We need to create more than one package with proper name and all package name in lower case.

In Each package we need to write more than one class and each class follow Pascal naming rules.

If class contains one word then first letter upper case if class contains more than one word

Each word first letter upper case.

Employee

EmployeeDetails

EmployeeService

Inside a class we need to write all instance variable with proper access specifiers

private

default

protected

public

empty constructor

parameter constructor

then write business method with private or public (but method must be public).

JavaBean class (JavaBean design pattern).

JavaBean class known as 100% pure encapsulation class.

vadafone.banking.com.team.bean

according to java bean class we need to write all variable must be private

and for each variable we need to write setter and getter method.

Setter method name start with set followed by variable name

Getter method name start with get followed by variable name.

If we display any user defined class reference in printlin method. Internally It will call toString() method of Object class by default and that method return the object in string format as [packageName.classname@hashcode](mailto:packageName.classname@hashcode).

Service design pattern : This class contains business method.

EmployeeService

Add employee, delete employee, update employee, retrieve employee

ProductService

AccountService

LoginService

Project Name : ProductManagementSystem

Create three package : pms.bean, pms.service, pms.main,pms.exception

As a rules service class not to interact with keyboard.

As a rules main class only responsible to interact with keyboard to receive the value.

So inside a service class don’t create scanner class object.

Variable and method must be follow camel naming rules.

If variable name or method name contains one word then it must be in lower case.

If it contains more than one word from second word onward first letter upper case.

**Design Patterns, Secure Coding and coding standards**

**Day 2 :**

**07/18/2023**

**Design pattern : Best Practise or solution for repeating problems.**

**GOF : Design pattern. This GOF design pattern mainly divided into 3 types.**

**Creational design pattern**

**5 types of creational design pattern : these design pattern help to create the objects different ways.**

1. **Factory design pattern**
2. **Abstract factory design pattern**
3. **Singleton design pattern**
4. **Builder design pattern**
5. **Prototype design pattern**

**Structural design pattern**

1. **Adapter design pattern**
2. **Bridge design pattern**
3. **Composite design pattern**
4. **Decorator design pattern**
5. **Flyweight design pattern**
6. **Façade design pattern**
7. **Proxy design pattern**

**Behavioural design pattern.**

1. **Chain of responsibility design pattern**
2. **Command design pattern**
3. **Interpreter design pattern**
4. **Iterator design pattern (Iterator or for each is base upon iterator design pattern).**
5. **Mediator design pattern (main method is interact with keyboard according to mediator design pattern).**
6. **Visitor design pattern.**
7. **Observable design pattern. (Observable and subscriber) one to many**

**Why**

**When**

**Where**

**Anti – design pattern.**

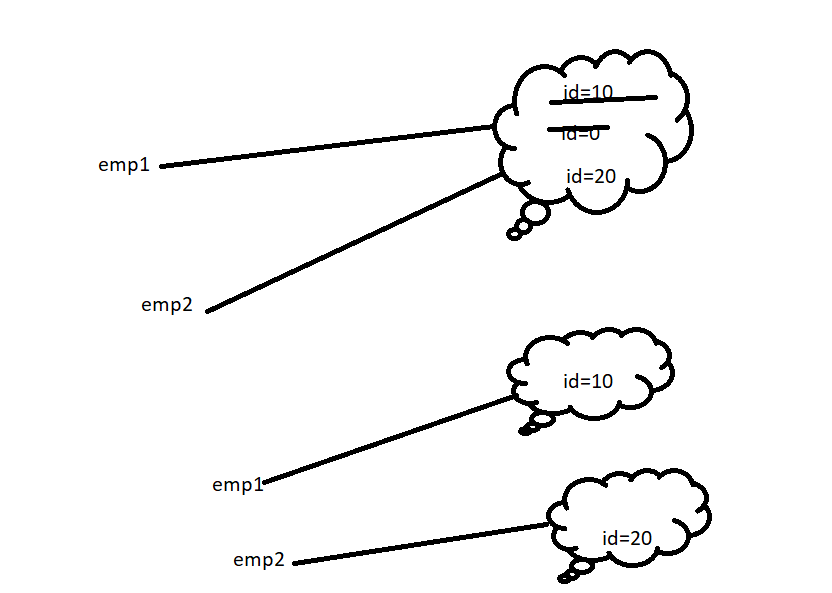
**Static factory method design pattern :**

**The method is responsible to create the object is known as static factory method design pattern.**

**When we create private constructor we can’t create the object of that class from outside.**

**Singleton design pattern : if we want to allow to create only one object of that class not more than one then we can use singleton design pattern.**

**But we can create more than one reference all reference point to same memory.**

****

**Database connection or any shared resources we use singleton design pattern.**

Factory pattern Factory design pattern create the object in such a ways that let sub – classes decide how to implements the object creation logic.

Loan rateOfInterest : 0.5% generic

class Shape {

draw(“Generic Draw”);

}

HomeLoan rateOfInterest : 0.8%

class Circle extends Shape {

draw(“Circle Draw”);

}

CarLoan rateOfInterest : 1.2%

Class Triangle extends Shape {

draw(“Triangle draw)

}

LoanFactory

class ShapeFactory {

public Shape static getShapeInstance(String type) {

if type is Circle we will return Circle object

if type is Triangle we will return Triangle object else return null.

}

}

Public class Demo {

Public static void main(String args[]) {

Shape ss = ShapeFactory.getShapeInstance(“Triangle”);

ss.draw();

Loan ll = LoanFactory.getInstance(“car”);

ll.getInterestDetails();

}

}

Prototype design pattern : using this design pattern we can create the clone of the object or duplicate object. Rather than creating new object. (ie copy constructor).

Employee

id=100;

name=”Steven”

salary = 24000;

salary = 24000 + 10%salary;

salary =salary updated base upon %

to create the clone java provided pre defined method ie clone() part of Object class and this method throw checked exception .

which class object we want to create the clone that class must be implements Cloneable interface.

This interface doesn’t contains any method. Marker : the interface doesn’t contains any method is known as marker interface.

**Day 3 :**

**07/19/2023**

Type casting :

Type casting on data types level.

Converting one data type to another data type is known as type casting.

int family

byte 1

short 2

int 4

long 8

-------------🡪implicit type casting ----🡪

byte short int long

🡨----------explicit type casting -----------

byte a=10; range -128 to 127

short b=a; implicit type casting.

short c =129; c size 2 byte memory

byte d=c; error

byte d =(type)c;

byte d = (byte)c; explicit type casting

byte age=21;

int age=21;

int : 4 byte without decimal

float : 4 byte with decimal

double 8 bye with decimal

int to float : implicit

float to int : explicit

in java by default every decimal number consider as double.

DemoTest.java

Main method

do{

1: Display all files in ascending order

2: Sub option

switch()

case 1 : display all the files in ascending orders.

case 2 : sub switch or if statement

a : create a new file and take the file name

through keyboards. If files exits don’t create else create it.

b: ask the file name to delete if present delete else

display message file not present.

c: search the option file present or not.

d.exit

case 3 : exit main option

}while();

You need static folder path :

Demo

a.txt

b.txt

c.txt

DemoTest.java

Main class

MainMenu

Method

1: display all files in ascending

2: to call submenu method

3 : exit

SubMenu

Methods

1: create the file

2: delete the file

3 search the file

4 : exit

FileService

Display all file in ascending order

Create the new file

Delete the file

Search the files

DemoTest

With main option and sub option

FileService

Display all file in ascending order

Create the new file

Delete the file

Search the files

Create the Folder with name

Phase1 Capstone Project

File Handling : IO package

Input and Output operation

In Java we do IO operation using stream concept.

Stream : flow of data or it is an abstraction between source and destination.

import java.io.\*;

Stream

byte char

1 byte 2 byte

Input Output Input Output

InputStream OutputStream Reader Writer

All four type of abstract classes.

DataInputStream DataOutputStream InputStreamReader OutputStreamWriter

FileInputStream FileOutputStream FileReader FileWriter

BufferedInputStream BufferedOutputStream BufferedReadere BufferedWriter

ObjectInputStream ObjectOutputStream

PrintStream PrintWriter

Byte wise operation

Source : Keyboard, File

Destination : Console or monitor , File

System.out.println(“Welcome to Java”);

Scanner sc = new Scanner(System.in);

out static variable of PrintStream

in static variable of InputStream

System is a pre defined class part of lang package which contains three static field

It out, in, err.

out and err are reference of PrintStream and in is reference of InputStream.

System.in : it is consider as standard input device reference ie keyboard. InputStream always refer

To standard input device ie keyword.

System.out : it is consider as standard output device reference is console. PrintStream always refer standard output device.

PrintStream ps = System.out;

ps.println(“Welcome”);

Byte wise input and output operation

Source : keyboard

Destination : Console.

Character wise operation

09/04/2023

Basic Java programming

OOPS concept

Object

Class

Constructor

Encapsulation

Polymorphism

Inheritance

Abstraction

Abstract keyword

Static keyword

Final keyword

Interface

Multi threading

Software Testing

Testing is use to find the defect or bugs or error etc.

Programs

Demo.java

Input int a,b,sum

Process sum = a+a;

Output print (sum)

DevOps

class Operation {

public int add(int x, int y) {

int sum = x+y;

return sum;

}

}