22-08-2022

String:

* String is an object which represents sequence of characters.
* String is a pre defined class in java and it is non primitive data type.
* String is immutable(i.e. we can not change the value of string).

We can create String in 2 ways:

* By using new keyword.

String s=new String(“csr”); //by using new keyword each time we create a string a new object is created and value is stored in object.

* By direct initialization.

String s=”csr”; // here the value is stored in s and if we re initialize old value is removed and new value is stored inside the object and that object is String constant pool area.

ADVANCE JAVA

22-08-2022

abstraction:

Example1:

interface Bank

{

void withdraw();

void deposit();

}

class Check

{

private class SBI implements Bank

{

public void withdraw()

{

System.out.println(“money is withdrawn”);

}

public void deposit()

{

System.out.println(“money is deposited”);

}

public Bank getter()

{

Bank obj=new SBI();

return obj;

}

}

class Test

{

Public static void main(String[] args)

{  
Check c=new Check();

Bank b=c.getter();

b.withdraw();

}

}

Example2:

interface MyIterator

{

void add();

void remove();

}

class MyList

{

private class Yogi implements MyIterator

{

public void add()

{

System.out.println(“obj is add”);

}

public void remove()

{

System.out.println(“obj is removed”);

}

}

public MyIterator iterator()

{

MyIterator my =new Yogi();

return my;

}

}

class cotroller

{

public static void main(String args[])

{

MyList my-=new MyList();

MyIterator itr=my.Iterator();

itr.add();

itr.remove();

}

}

ARRAY SYNTAX:

datatype [] varName=new datatype[size];

datatype varName[]={values};

datatype [] varName=new datatype[size]{values};

* ARRAY is a collection of homogeneous elements and array is an object.

Example:

int [] a=new int[5];

a[0]=1;

a[2]=2;

a[3]=3;

System.out.println(a.length);

Drawbacks of array:

1. We cannot increase the size of an array once it is declared.
2. We cannot store heterogeneous values.
3. No inbuilt methods to call using reference of array object.

Wrapper class:

boolean-----🡪Boolean

char----------🡪Character

byte----------🡪 Byte

short--------🡪 Short



int------------🡪Integer

long----------🡪 Long

float---------🡪Float

double------🡪 Double

* Converting primitive type value into non primitive type value is known as boxing.
* Converting non primitive type value into primitive type value is known as unboxing.

Wrapper classes are utilised to perform generalisation concept.



int i=3;

Integer b=i;



Object o=obj;

Generalisation: converting every data type values into general data type i.e, object.

Object o1=3;

Object o2=6.3;

Object o3=”csr”;

Object o4=new Car();

Example:

Object [] obj =new Object[6];

Obj[0]=2.5;

Obj[1]=”csr”;

Obj[2]=new A();

22-08-2022

Core Java(J2SE JAVA STANDARD EDITION)----🡪JDBC

Advance Java(J2EE JAVA ENTERPRISE EDITION)---🡪 Servlet & JSP

FRAMEWORKS:

1. Spring
2. Hibernate
3. Spring MVC
4. Spring boot

* With the help of core java we can create only STAND ALONE APPLICATIONS like calculator , excel sheets etc.
* In order to create any web applications we have to learn advance java technologies like servlets , JSP etc.

Web application: The applications which are stored in web server and accessed with the help of internet is known as web application.

* Web application ca be created with the help of front end technologies as well as both front end and back end technologies.

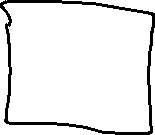
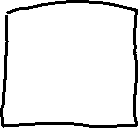
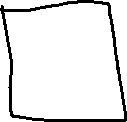
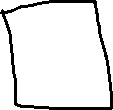
Front end web based application:

* To create these we applications it requires only front end technologies and its frameworks like HTML,CSS,JAVA SCRIPT with angular js or react js.
* These web applications does not have any connections with data bases and also servlets and jsp’s are not required hence it is called static web application.

Example: Wikipedia ,shashanksk.blogger ,word press blogs etc.

Front end and back end web based application: (DYNAMIC WEB APPLICATION)

Servlet : reads the



Data, compares the



Data ,Process the data

BROWSER WEB SERVER APP SERVER DATA BASE

In the above figure we can understand how JDBC , servlet, jsp are utilised to create Dynamic web application.

* JDBC is used to connect the application server which contains java application to the database application.
* JDBC is acting as a mediator between java application and data base application.
* In the above scenario the request passed by the user to the web server and it is forwarded to application server .
* And the servlet which is present in application server it will take help of database and process the data which is related to the user and generates the response page.
* In simple words the servlet takes the request object and process the data and generates the response page.
* So it is called as processing logic or business logic.

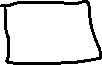
JSP:

* jsp’s are utilised to display the data. It acts as a mediator between application server and the browser application.
* So jsp is called as presentation logic or view component.

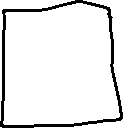
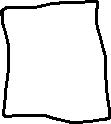
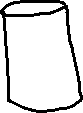
23-08-2022

JDBC:

Result set



JDBC connection



Steps to connect java application to the database:

1. Load and register the driver software. (arrange the translator 1.)
2. Establish the connection. (road way 2.)
3. Create statement object. (arrange the vehicle 3.)
4. Execute the query or hold the data in result set(delivering letter or getting response4.)
5. Close the connection (return back 5.)

Storage areas:

In order to store the data we can use text files , excel sheets , data bases and big data technologies for data ware houses etc.

Text document:

We can store any data in the text files and can be accessed easily with the help of file handling mechanism.

Drawbacks of text files:

* It allows duplicate data.
* It will not store data in any format.
* It does not provide any calculation of data.
* It is not secured.
* It does not support query language.

Excel sheets:

It can store the data in tabular format and also helps us to calculate the data.

Drawbacks of excel sheets:

* It allows duplicate data.
* It does not support query language.
* It is not secured.

Data bases:

* Data bases like rdbms is used to store the data in tabular format and provides primary key feature in order to avoid duplicacy.
* It supports query language and also provides security for the data.

Drawbacks :

* Some data bases like rdbms can not store semi structured data like xml files and unstructured data like image files and video files.
* It can not store very large amount of information.

To over come these drawbacks we have to use big data technologies like Hadoop .

Q) How to connect the java application to the database application?

A) try{

Class.forName(“com.mysql.cj.jdbc.Driver”);

Connection connection = DriverManager.getConnection(url,username,password);

}

catch(ClassNotFoundException e){}

* The only class is present in JDBC connection is DriverManager class.
* DriverManager.getConnection(url,username,password) it throws checked exception i.e., SQLException .

Q) WAP TO CREATE A DATABASE IN mysql APPLICATION

A) Class CreateDataBase

{

public static void main(String[] args)

{

try

{

Class.forName(“com.mysql.cj.jdbc.Driver”);

Connection connection = DriverManager.getConnection(url,username,password);

Statement st = connection.createStatement();

st.executeUpdate(“create database movie”);

connection.close();

System.out.println(“database is created”);

}

catch(ClassNotFoundException e){}

}

}

24-08-2022

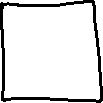
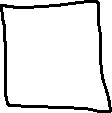
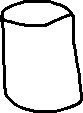
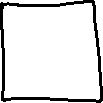
Types of drivers:

There are many driver soft wares but all the driver soft wares are classified into 4 types based on their functionalities .

1. JDBC-ODBC Driver



Java JDBC ODBC ODBC DB



application DRIVER DRIVER



* JDBC-ODBC Driver takes support of ODBC Driver.
* JDBC-ODBC Driver is also known as snail driver because of its low performance because this type 1 driver first converts JDBC calls into ODBC calls and ODBC calls into database calls.
* It has 2 levels of conversions .
* It comes along with the Jdk no need of installing any software.
* Its very easy to se and maintain.

Drawbacks:

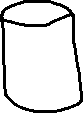
Type 1 drivers work only on windows machine because ODBC is written in mostly C or C++ so it is platform dependent.

ODBC- open database connectivity.

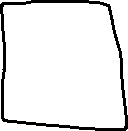
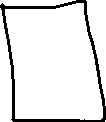
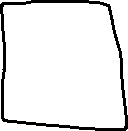
1. Native partly Java Drive:



JAVA NATIVE NATIVE



APPLICATION PARTLY SPECIFIC



JAVA DRIVER LIBRARIES



* Type 2 driver internally uses native specific libraries to communicate with the database.
* We need to install these native specific libraries (jar files).
* Type 2 drivers converts JDBC calls into vendor specific(database related libraries) calls.
* Type 2 drivers maintains only one level of conversion.
* Type 2 driver performance is much better compared to type 1 driver.
* Type 2 driver is a combination of java and C languages. So these type 2 driver is platform dependent driver.

Drawbacks:

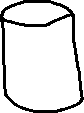
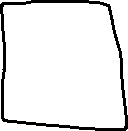
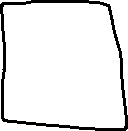
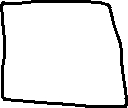
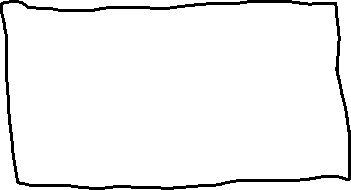
* Type 2 driver is database dependent driver because type 2 driver depends on native specific libraries.
* We need to install the jar files (database libraries) in client machine.

1. Network protocol driver:

JAVA NETWORK MIDDLEWARE



APPLICATION PROTOCOL SERVER



DRIVER

* Type 3 driver does not depend on any ODBC driver or any native specific libraries.
* It internally uses middle ware server. This middle ware server internally may utilise type 1,2 or 4drivers.
* This driver software is not communicating directly with the database so we can easily switch to any kind of database.
* Hence it is both platform independent and database independent driver.
* Network protocol driver is completely returning java language.

Drawbacks:

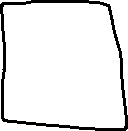
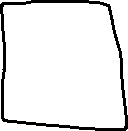
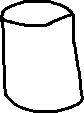
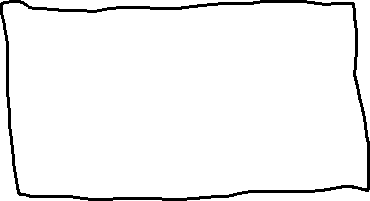
* There might be performance issues because of this middle ware server.
* Type 3 driver is more costlier compared to all other drivers.

1. Thin driver (or) pure driver:

JAVA PURE JAVA



APPLICATION DRIVER



* Type 4 driver does not require ODBC driver or native specific libraries or middle ware server.
* These type 4 driver is directly communicating with the database.
* Java calls are directly converted into data base calls.
* So this is called as tin driver.
* Type 4 driver is completely written in java.

Drawbacks:

* Type 4 driver directly communicates with the database so it is difficult to switch between different databases.
* These type 4 driver is database dependent driver.

Q) When do w utilise type 3 driver?

A) In case of large scale applications and enterprise applications requires internet dataase server , so we require network protocol driver(type 3 driver).

Q) When do we utilise type 4 driver?

A) In case of Stand alone applications , small scale applications which does not require different databases so we use Thin driver(type 4 driver).

Note:

It is never recommended to use type 1 and type 2 drivers.

To add jar files to java project

1. Select java project.
2. Right click.
3. Select build path-configure build path.
4. It will be reflected order ad export.
5. Select libraries.
6. Select class path
7. Click on external jars.
8. Add the jar file.

12-09-2022

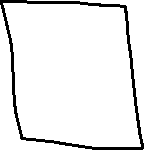
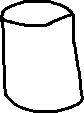
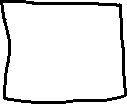
Servlet:

* Servlet is an API which is utilised to create dynamic web applications.
* Servlet is a component which will be used as a component of application servers.
* As servlets are used for creating web applications so it is called as web component.
* Servlet plays important role in case of dynamic web applications because it creates separate threads for every request by the user and gives immediate response.
* Servlet takes the data given by the user and compares with data present in data base.
* Then these servlet generates the response page dynamically according to the data provided by the user, so servlet is known as processing logic .

request



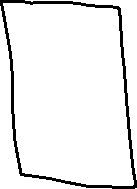
servlet



reads



,compares



and process

the data

Application server

* Every web application will be accessed through web with internet connection.
* But all these web applications which are using servlets internally utilises application servers such as tomcat server , glassfish ,web sphere etc.
* For learning purpose we’re using tomcat server and it is also known as web container, because it loads all web components like servlets ,jsp etc.
* Servlet is a java file where we have to connect the view page (jsp or html) to this servlet class .

**1.WHAT IS A METHOD?**

Ans. Method is a part or module of program which contain a block and inside the block it can contain logic. The logic can be reused later at different instance just by call the method by its name.

Methods are used to perform operations.

**2.WHAT ARE THE DIFFERENT TYPES OF METHOD?**

Ans. 1.On the basis of who is declaring the method, there are two types:

i)Predefined method: Predefined methods are the methods which are written by java programmers and are located I predefined classes which are inside predefined packages. Predefined methods are called to perform a specific task.

ii)User defined method: sometimes the predefined methods don’t match the requirement of programmer. Hence to overcome this problem the user programmer can define his own logic and let the other programmer reuse his logic inside a method. So that we can call the method where ever we want the logic to be executed. The methods which will be declared by user programmer are called user defined methods.

2. on the basis of how we can call he methods we have two types:

1) static methods:

1. The methods which are declared using the keyword static are called static methods.
2. static methods are used when an operation has to be common for all the objects.
3. static methods are loaded in static pool area.
4. We can call the static methods directly within the same class.
5. We can access static methods of one class from another class using class name which contains the SM that we want to call.
6. We can call SM from NSM directly, because there is only one copy of static method available inside static pool area.
7. static methods get loaded into the static pool area only once during the execution of program.
8. Variables declared inside SM are only of type of local variables.

Variables declared inside static methods can not be accessed from any other static method.

(whether its within same class or from other class.)

2)non static methods:

1. The methods which are declared without using the keyword static are called non static methods.
2. Non static methods are used when an operation is not common for all the objects.
3. Non static methods are loaded inside heap area inside heap memory inside object.
4. We can call the non static methods directly from another non static method within the same class.
5. We can access non static methods of one class from another class using objects and class name which contains the NSM that we want to call.
6. We can call NSM from NSM directly, because they are loaded inside the objects in heap memory.
7. Non static methods get loaded into the heap memory inside objects .
8. Variables declared inside NSM are only of type of local variables.

Variables declared inside non static methods can not be accessed from any other non static method.

**3. Can we have multiple number of same methods declared in a class?**

Ans. Yes we can have multiple number of same method declaration in a class.

But if we want to do so we must take care of the following:

* Number of parameters must be different.
* If number of parameters are same then we must consider the type of parameters are different or the order of parameters are different.

If we will try to have multiple methods with same name in a class then it’s a process of method overloading (Compile time polymorphism).

**4. What happens when we declare a method as final?**

Ans. The method can not be reimplemented in sub class. i.e. we can’t perform method overriding.

**5.What is method over riding?**

Ans. Sub class having same non static method name with same signature and different implementation.

Method overriding can be achieved by using inheritance.

Steps :

1. declare a NSM with implementation in a class(parent)
2. inherit the property of parent class to a new class(child class)
3. re-implement the method logic I the child class.

**6.Why do we need to reimplement the parent class method in child class or**

**Why do we need method over riding?**

Ans. Because sometimes parent class method’s logic may not satisfy child class required logic/operation. Hence we need to re-implement the logic of method which is shared to the child class. we will re-implement the logic in the sub class for sub class object inside the same non static method which is declared inside the parent class.

**7.What are factors which are taken care during method over riding? Or can we have different return type in method over riding?**

**Or can we have different parameter in method over riding?**

**Ans.** NO. because for method over riding the method signature in child class must be same as that of the method signature of parent class.

FAQ

1.Is it mandatory to make the class abstract if there is an abstract method?

A. Yes

2.Can we make a class abstract , if does not contain any abstract method?

A. yes

3.Can we create object of abstract class?

A. no

4.Can we create reference variable for abstract class?

A. yes

5.What is the benefit of taking abstract methods?

A. method signature standardization.

6.Can we create constructors with abstract class?

A. yes, but without creating object constructor is not called.

7.If sub class does not over ride abstract method, then what should we do to the sub class?

A. Make the sub class abstract.

07-10-2022

1. what is a constructor?
2. Types of constructors?
3. Default, non-parameterised and parameterised.
4. Difference between no arg constructor and default constructor?
5. What is constructor overloading?
6. What is constructor chaining?
7. Can we call constructors recursively?
8. In a class one super () statement is mandatory because every class should call the super most class the object, if we have recursive calling for constructors, we cannot have super () statement.
9. Difference between this and super?
10. Difference between this () and super ()?
11. Difference between this, this () and super, super ()?
12. What is encapsulation?
13. How we are providing security for encapsulated class?
14. How we are getting code re usability from encapsulated class?

15/07/2022 JAVA PROGRAMMING

LOOPS:

1.ENTRY CONTROLLED LOOP: FIRST CHECK THE CONDITION AND THEN EXECUTE THE LOOP

EX. for loop,while

2.EXIT LEVEL CONTROLLED LOOP: FIRST EXECUE THE LOOP AT LEAST ONCE AND THEN CHECK THE CONDITION

EX. do-while loop

METHOD RECURSION:

WAP TO DISPLAY A MESSAGE 5 TIMES WITOUT USING LOOP

class DisplayMessage

{

static int i=1;

public static void displaymessage()

{

if(i<=5)

{

System.out.println("display message");

i++;

displaymessage();

}

public static void main(String[] args)

{

displaymessage();

}

}

}

Constructors overloading vs Method overloading

constructor overloading is somewhat similar to method overloading.

If we want to have different ways of initializing an object using different number of

parameters, then we must do constructor overloading , because we do method overloading for the

situation when we want different definitions of a method (different ways of performing a task)based on different parameters.

/\*

Method Hiding and Method overriding:

class A

{

static void m1(){}

void m2(){}

void m3(){}

}

class B extends A{

static void m1(){}

void m2(){}

void m4(){}

}

When a child class is encountered by compiler, it will consider all

the methods to be either belonging to sub class are overriden,hidden

or same class method.

Compiler's Job role:

Compiler checks the equality of method signature in both the class

i.e., in child class as well as parent class. If both the method

signature are same then compiler says that the method present it says

that the method is not of sub class. Its of super class.

Then it checks is it static or non-static method.

If static then its method hiding.

If non static then its method overriding.

e.g: Once a cheater may always be a cheater. :-)

Inheritance extends the functionality of super class by hiding the

method or overriding a method.

Q. what is MH?

Answer: Process of redefing a super class static method

in sub class to provide new implementation to super class method

as per req. of sub class is called method hiding.

Q. What is MOVR?

Answer: The process of redefining super class non static method

in sub class for providing sub class's requirement is called

method overriding.

NOTE: MH is for SM.

MOVR is for NSM.

Q. When do go for MH?

Answer: When super class SM is not satisfying the sub class req.

Q. When do we go for MOVR?

Answer: When super clas NSM is not satisfying the sub class req.

JAVA API LEVEL e.g:

class Object

{

public String toString()

{

//logic for displaying classname@hashcode

}

}

class Demo

{

int x;

//sub class not satisfying super class NSM

public String toString()

{

return ""+x;

}

}

e.g: Parent searched a girl for marriage but child wants to marry someone

else

-------------------------------------------------------------

COMPILER & JVM ACTIVITIES :

Compiler Role: Linking the methods(binding method call to method definations)

JVM Role: Executing method which is bind by Compiler

----------------------------------------

How do compiler binds method?

Answer: Binding the invoked method with its definition. For which

it will consider :

By argument-> which parameter method has to be executed

By reference-> finding which class will method bind and execute

NOTE: Compiler always works for RV belonging to class type.

------------------------------------------------

What is JVM doing in MH and MOVR?

Answer: Executing the invoked method i.e, bind by compiler.

-------------------------------------

WAP to create a class named as Parent and define one SM and two

NSM with implementations. Create another class named as Chinna

extending Parent.Define same SM with different implementation.

one NSM with another implementation and sub class own NSM with

some implementation.

Take a TestClass1 and create object of Chinna and access all the

methods.

Take TestClass2 and create object of Parent and access all the methods.

Take TestClass3 and create object of Chinna belonging to Parent.

(upcasting)

Access all the methods.

Take TestClass4 and create object of Parent belonging to Chinna(DC)

Access all the methods.

\*\*\*\* NOTE:COMPILER STORY

1. Compiler searches for method definitions in the referenced

variable class when invoked method from sub class referenced variable

it seraches for method in sub class, if not available then goes and

searches in super class if not available then move to further sub class.

2. If you invoke a method using super class referenced variable then

compiler seraches method only in super class if not available then

goes to search in further super class.

3. If super class Rv taken and stored sub class type object then

compiler searches for method in super class.Compiler doesnot have any

idea of object.Objects come to existence during run time only.

SM flow will be different from NSM flow.

WHEN COMPILER GETS SUPER CLASS RV WITH SUB CLASS OBJECT,IT GOES

TO SUPER CLASS AND CHECK WHETHER ITS STATIC OR NOT. IF STATIC THEN

REMOVES RV AND PROVIDES CLASSNAME DURING COMPILATION.

------------------------------------------IMPORTANT TOPIC\_\_\_\_\_\_\_\_

Q. Does JVM has any information regarding from which class it should execute

the static method?

Yes.JVM goes to super class without object reference if method is static.

And will take the RV(object) alongwith it,if the method is non static.

It dont execute the method from super class, it first checks whether

the same method is overridden in sub class then invokes method from

sub class.Hence SM are searched and then executed.

BOth SM are searched in Super Class but executes from RV of class,

bcz it doesnot require object for execution.

But NSM are searched in super class but executes from current object class.

Q.From which class SM is searched?

Answer: from RV class.

Q. From which class NSM is searched?

Answer: from RV class.

NOTE: COMPILER & JVM searches for SM and NSM in RV class.

If method is available in RV class then execution depends on two things:

i) based on method type i.e whether SM or NSM

ii) based on object.

If method is static type, then execute from RV class.

If method is NS type, then execute from 'current object'class

Hence, in the e.g: if CO is Parent type then execute the method from

Parent.

If CO is Chinna type then execute from Chinna.

Q. From which class SM is executed?

Answer: from the RV class.

if available in RV class,executed or else error.

Q. From which class NSM is excuted?

Answer: from current object class.

if available in RV class then search in sub class then execute from

there if current object is of sub class.

If method is not found in RV class then search for Parent Type class upto

java.lang.Object class as its the supermost class for every class.If

not found there the only CTE.

\*\*\*\*SM and NSM with inheritance, polymorphism and abstraction done\*\*\*\*.

Important Questions:

Q.1 what is Method Hiding?

Answer: Process of redefining the super class static method in sub class

with different implementations as per the requirement of sub class,

is called Method Hiding.

Q.2 What is Method Overriding?

Answer: The process of redefining the super class NSM in sub class with

different implementation as per the requirement of subclass,is called

Method Overriding.

Q.3 What is Method Overloading?

Answer: The process of defining or declaring multiple methods with same

name but with different parameter type,list or order of parameter is

called Method Overloading.

Q.4 Can we perform Method Hiding or Method Overriding in the same class?

Answer: NO.

Q.5 Can we perform Method Hiding or Method Overriding in the sub class?

Answer: Yes, we can hide or override a method in subclass 'ONLY'.

Q.6 Can we overload a method in sub class?

Answer: Yes its possible in sub class as well as in sub class.

Q.7 How do we identify whether a method in sub class is hidden,overriden

or overloaded in sub class?

Answer: By Considering the following points:

i) If the method present in super class is static type,

and is redefined in sub class with the same method signature,

then its Method Hiding.

ii) If the method present in super class is non static type,

and is redefined in the sub class with the same method signature,

then its Method Overriding.

iii) If there are multiple methods with same name

in same class or in sub class but with different parameters

list,type or order, then its Method Overloading.

Q8. Identify the concepts in the below codes:

\*/

18/7/2022

1.WAP TO ENTER A NUMBER AND DISPLAY THE DIGITS IN DIFFERENT LINES

2.WP TO ENTER A NUMBER AND DISPLAY THE SUM OF DIGITS.

3.WAP TO ENTER A NUMBER AND DISPLAY PRODUCT OF DIGITS.

4.WAP TO ENTER A NUMBER AND DISPLAY THE NUMBERS IN ASCENDING ORDER.

5.WAP TO ENTER A NUMBER AND DISPLAY REVERSE NUMBER ALONG WITH DIFFERENCE BETWEEN THE REVERSE NUMBER

22/07/2022

1.WAP TO COUNT EVERY DIGIT OF A GIVEN NUMBER.

2.WAP TO PRINT THE DIGITAL SUM OF GIVEN NUMBER.(DIGITAL SUM=SUM OF DIGITS).

3.WAP TO PRINT THE DIGITAL PRODUCT OF GIVEN NUMBER.

4.WAP TO PRINT SPY OR NO.

5.WAP TO PRINT REVERSE OF NUMBER.

6.WAP TO PRINT DIFFERENCE BETWEEN REVERSE ON NUMBER AND GIVEN NUMBER.

7.WAP TO PRINT SUM OF DIGITS IN EVEN PLACES AND PRODUCT OF NUMBERS IN ODD PLACES.

8.WAP TO PRINT FACTORIAL OF A NUMBER.

9.FIBONACCI SERIES.

10.WAP TO CHECK PERFECT NUMBER OR NOT.

11.WAP TO CHECK

i)ARMSTRONG

ii)STRONG

iii) HAPPY NUMBER

iv) HARSHAD NUMBER

v) NEON NUMBER

vi) AUOMORPHIC NUMBER

vii) SUNNY NUMBER

viii) DISARIUM NUMBER

ix) PRONIC NUMBER

x) TRIMORPHIC NUMBER

xi) EVIL ODIOUS NUMBER

xii) DUCK NUMBER

22-11-2022

String and String handling:

String is a combination of characters represented in double quotes.

In this case we take string as instance of String class.

String as a class:

* String is a pre-defined class.
* String is public and final.
* Generally String can be taken as a data type which is used to create variable that can hold string literal.
* String is available in the package `java.lang`.
* we don’t have to import java.lang in our program because it’s a default package, hence whenever a programmer uses String in the program or the methods of String in the program he doesn’t have to import java.lang .
* String is immutable. Immutable means unchangeable.

Q: What is the difference between the concept of final and immutable?

A: final means fixed. If any property is made final then we’re not allowed to make any changes to the property. If we try to make any changes then we will get Error.

Immutable: if anything is immutable then it becomes unchangeable. If we try to make any changes then a new copy of memory is generated and the original data remains as it is.

Ex:

String is immutable.

class Immutable\_String

{

public static void main(String[] args)

{

String str=new String("csr");

System.out.println("before changes");

System.out.println(str);

System.out.println("after changes");

System.out.println(str.concat("c"));

/\* str.concat("c") is a de referred string object and finalize method from object class will come and remove from memory \*/

}

}

In java.lang we have 3 different classes which are required to handle group of characters.

They are

1. String
2. StringBuffer
3. StringBuilder

* All these classes implement CharSequence (interface).
* Only the String class implement from Comparable interface where as the other 2 don’t implement from Comparable interface.

Object

StringBuffer©

StringBuffer©

String©



Comparable(i)



Serializable(i)

CharSequence(i)



All the classes that is String, StringBuilder, StringBuffer implements from Serializable interface.

In java any data with in double quotes will be considered as instance of String.

Q: How do we create instance of String?

A: There are 2 ways to create instance of String

1. Using new keyword.
2. Without using new keyword.

Q: What is the difference between creating String object using new keyword and without using new keyword?

A: To understand the difference



1040

1030

1020

1010

james

james

james

Java Stack area

S1

S2

S3

S4

String pool area

1111

Heap area

1030

1040

Method area

we need to consider the following program

==: content level comparison for PTD

Address level for NPTD

equals(): content level comparison for NPTD

Ex1:

/\*

int x;

System.out.println(x);// CTE: Local variable x is not initialised

Note: or PTv we must initialize before using them

String str;

System.out.println(str);//CTE: Local variable str is not initialised

\*/

String str=new String();

System.out.println(str);//empty space

Ex2:

class A

{

static String s;

public static void main(String[] args){

System.out.println(s);// null

}

}

Ex:3

Class A{

public static void main(String[] args){

String s=new String(null);

System.out.println(s);

}

}

Note: we can not pass null as argument directly to the String constructor because if we do so then we will get CTE which is an ambiguous error because null is matched with all parameters of String class constructor.

CTE: error: reference to String is ambiguous

String s=new String(null);

^

Both constructor String (StringBuffer) and String (StringBuilder) in String match

Ex4:

class P

{

public static void main(String[] args)

{

String s=null;

System.out.println(s);

}

}

Note: we can display the null if its stored in a variable. We can’t display null directly through System.out.println().

Ambiguous situation:

* Displaying null

Sopln(null);

* Passing null to String constructor

String str=new String(null);

Ex5:

Class A{

public static void main(String[] args){

String str=new String((StringBuffer)null);

}

}

RTE: Exception in thread “main” java.lang.NullPointerException:  
cannot invoke “java.lang.StringBuffer.toString()”

Because buffer is null.

* In String we have 8 important constructors.
* The constructors are as follows

1. String (): It is used to create an empty String object.

Ex: String str=new String ();

System.out.println(str);// no output

1. String (String value): It is used to create String object with the given String literal or group of characters.

Ex:

String str1=”csr”;

String str2= new String(str1);// getting the String copy of str1

String str2= new String(“James”);// passing String literal directly

Note: whenever we have String copy, we get 2 different String object with the same data.

If we use String assignment then the current object reference is copied to the given variable. Here no new object is created.

Calculating the number of objects created in String program: