Eclipse short cut

Cntl+space+enter

Sysout --System.*out*.println();

Main method ---cntr+shift+enter

following classes from 'java.lang' package are used to create String objects

1. String

2. String Builder

3. String Buffer

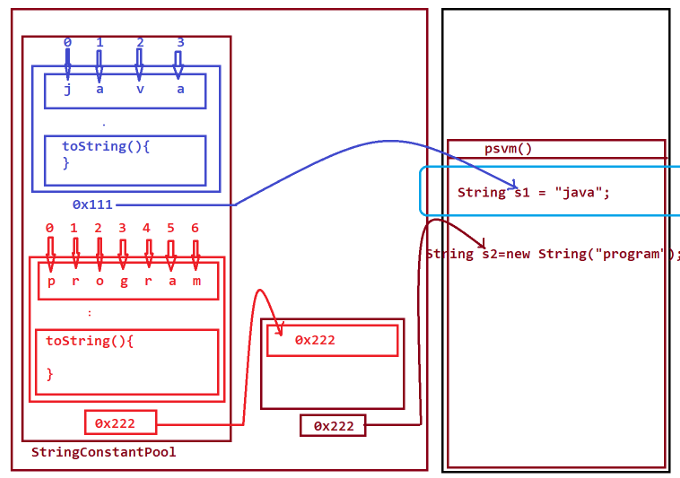
The 'String' class is from java.lang package and the objects which are generated from 'String' class are Immutable objects. (Secured)

syntax-1 : Using String literal process

String str = "java";

Syntax-2 : using new operator process

String str = new String("program");



1.STRING LITERAL PROCESS String S2=”java”;

when execution control finds String Literal process,

Then the execution control will check the StringConstantPool is any Object having the same data.

=>If same object is available then the reference of the object is used without creating new object.

Here object **str** and **S2** having same object

=>If same object is not available then the new object is created.

2. NEW OPERATOR PROCESS String str = new String("program");

In new operator process one reference is created part of Heap Area and the reference is binded with reference of Object created part of String Constant pool

This String Comparision can be done in the following ways:

1.Using equals() method

2.Using compareTo() method

3.Using 'is equal to'(==) operator

1.Using equals() method:

str.equals(s1) compare two object

str.equalsIgnoreCase(s1) due to ignore case object result get same and it’s true

=>equals() method will compare two strings and generate boolean result.

Note: =>In realtime equals() method is used in Authentication process.

2.Using compareTo() method:

=>compareTo() method also compares two Strings and generate int value.

int k = s1.compareTo(s2);

if k==0 then the Strings are equal

3.Using 'is equal to'(==) operator:

=>'is equal to'(==) operator will compare the references of an object and which will not compare the content of an Object.

Here some scenario

String s1 = "java";//AutoBoxing process

String s2 = "java";//AutoBoxing process String s5 = "Java";

String s3 = new String("java");//Boxing process

String s4 = new String("java");//Boxing process

S1 & s2 are same object or s1 & s5 also diff object

S3 & s4 both are different object

In string class charAt work as left to right

String Buffer class

StringBuffer sb=new StringBuffer (); // object creation

In this syntax the StringBuffer object is created with the default capacity 16.(characters)

=>we use append() method to add the data to the StringBuffer object.

=>The capacity of StringBuffer object increases dynamically by doubling the capacity and adding 2. 16-->(16+16+2)-->34-->(34+34+2)-->70...

=>we use reverse() and insert() methods on StringBuffer object,because StringBuffer object is mutable object.

Str=welcome;

sb.charAt(6)); in StringBuffer charAt work as right to left

StringBuffer class is Synchronized class.

define Synchronized class?

=>The class which is declared with synchronized methods is known as Synchronized class.

faq: define Synchronized methods? =>

The methods which are declared with synchronized keyuword are known as Synchronized methods.

faq: wt is the advantage of Synchronized methods?

=>The Synchronized methods will be under the lock and the methods are available to one user at a time. (Secured methods)

>StringBuilder class is same like StringBuffer,but StringBuilder class is NonSynchronized class,

which means the methods which are declared within the StringBuilder are NonSynchronized methods.

=>StrinBuffer is used in MultiThreading applications and StringBuilder is used in Non-MultiThreading applications.

Arrays

Arrays is a collection of similar type of elements.

The sequenced collection of elements of same datatype is known as Array.

Arrays in Java are categorized into two types:

1.Single Dimensional Arrays

2.Multi Dimensional Arrays

Class\_name arr\_var[] = new Class\_name[size];

Integer i[] = new Integer[3]; String str[] = new String[3];

int[] numbers = new int[5];

int[] marks = {85, 90, 78, 92};

String[] fruits = new String[]{"Apple", "Banana", "Mango"};

The Array which is declared with java.lang.Object class is known as Object Array.

Note: =>This Object Array can hold Dis-Similer objects or Objects of different classes.

DisAdvantage of Arrays:

=>Array size once defined cannot be modified at runtime ,because of this reason Arrays are not preferable to hold Dynamic data or runtime data.

=>The DisAdvantage of Array can be overcomed using Collection.

=>Collection is an interface from java.util package and which is root of Java Collection Framework. =>This Collection interface is extended into the following SubInterfaces:

1.Set

2.List

3.Queue

1.Set: =>

Set organizes elements without index values and which cannot hold duplicate elements.

public abstract int size(); public abstract boolean isEmpty();

public abstract boolean contains(java.lang.Object);

public abstract boolean add(E);

public abstract boolean remove(java.lang.Object);

(a)HashSet =organizes elements without any order. & null value

(b)LinkedHashSet= organizes elements in insertion order

(c)TreeSet =organizes elements in Ascending order.

Iterator is an interface from java.util package and which is used to retrive elements from Collection object.

hashNext(); This method is used to check the availability of element.

next(); This method will retrieve the element.

remove(); This method is used to remove the element.

Spliterator is an interface from java.util package introduced by Java8 version and which is used to retrive elements from Array objects,Collection objects and Stream objects.

forEachRemaining

2.List:

=>List organizes elements with index values and can hold duplicate elements.

=>Iterator is used to retrieve elements from Collection objects in forward direction. =>ListIterator is used to retrieve elements from only List objects in both directions(forward and backward).

=>In realtime Enumeration is used part of Servlet programming.

=>In realtime Vector is used part of Connection pooling concept. (Connection pooling means organizing multiple DataBase Connections)

=>Map is an interface from java.util package and which organizes elements in the form of Key-value pairs K - Key V – Values

(a)HashMap

(b)LinkedHashMap

(c)TreeMap

(d)Hashtable

=>Hashtable is synchronized class and remaining three classes are NonSynchronized classes. =>HashMap and Hashtable organizes elements without any order.

=>LinkedHashMap organizes elements in insertion order.

=>TreeSet organizes elements in ascending order based on key data.

Inheritance:

=>The process of establishing communication b/w two classes using 'extends' keyword is known as Inheritance.

class SubClass2 {

//members

}

class SubClass1 extends SubClass2

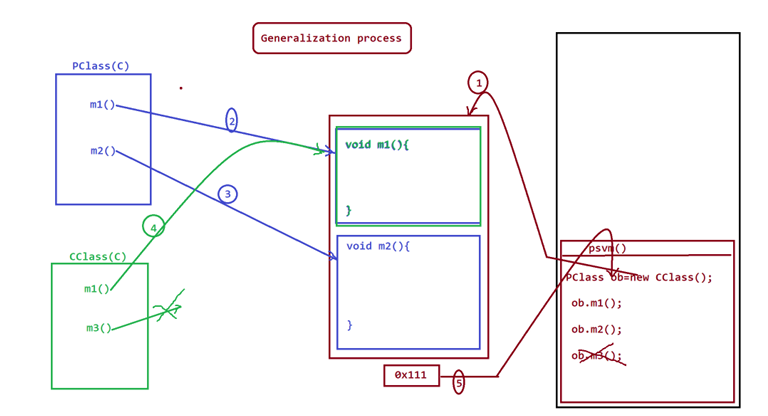
{ //members

}

In Normal Inheritance process we always create object for CClass or SubClass.

In Normal Inheritance process one reference is created and the reference is binded with all the NonStatic members of PClass and all the NonStatic members of CClass.

=>In Inheritance process the PClass is loaded first and then the CClass is loaded



In Zero param constructor no need to add super by default system add super()

Parameterized Constructor from the PClass

=>when we have parameterized constructor in the PClass then we must add super() to the Constructor of CClass to call PClass class Constructor.

when we read String data after reading numeric data,then the reading of String is skipped because "\n" is available in the buffer.

=>This can be overcomed using parseXXXX() methods,which are used to convert String data to numeric data.

byte b = Byte.parseByte(s.nextLine());

short s1 =Short.parseShort(s.nextLine());

int i = Integer.parseInt(s.nextLine());

long l = Long.parseLong(s.nextLine());

step1 : Save the program in Project Folder as Salary.java

step2 : Compile the program as javac -d . Salary.java

(-d .) indicates create package and load the package with class file

create Java Project Click in

File->new->Project->Java->select JavaProject and click next-> name the project and clicl 'finish'.

Encapsulation is the process of wrapping data (variables) and code (methods) together into a single unit (class), and restricting direct access to some of the object's components.

Abstraction is the process of hiding internal implementation details and showing only essential features of an object.

The methods which are declared with method\_body are known as Concrete methods

The methods which are declared without method\_body are known as abstract methods.

INTERFACE RULE

1. we use 'interface' keyword to declare interfaces

2.The members which are declared within the interface are automatically 'public'.

Note: =>The members which are declared within the class are automatically 'default'.

3.Interface can be declared with both Primitive datatype varaibles and NonPrimitive datatype variables

4.The variables which are declared part of Interface are automatically 'static and final' variables.

Note: =>'static' variables are binded to interface and access with interface name.

=>'final' variables must be initialized with values,once initialized cannot be modified.

Note:

1.used **implements** keyword when class need to implement

2.Interface can use the members of another interface using 'extends' keyword.

Abstract Classes in Java:

=>The class which is declared with abstract keyword is known as abstract class. =>These abstract classes can be declared with both abstract methods and concrete methods.

=>The abstract methods part of abstract classes must be declared with abstract keyword.

=>Abstract classes in Java are abstract components and which cannot be instantiated.

=>These abstract classes are extended to normal classes to have method\_bodies to abstract method signatures

Class can be declared with only concrete methods,but abstract class canbe declared with both abstract methods and concrete methods.

Classes can be instantiated,but abstract classes cannot be instantiated

Recording links

Hibernate @ 4.00 PM | Mr. Nataraj

<https://youtu.be/l4e70Yt4beQ>

<https://youtu.be/QHONIDQnEMY>

<https://youtu.be/uMEsZkwZYWw>

<https://youtu.be/eDv5_Z6tE_I>

<https://youtu.be/YxXHsPyeXkg>

<https://youtu.be/1GuL5RcyWHg>

<https://youtu.be/9tgr4wEl-8s>

**Spring JDBC Videos**

<https://youtu.be/4ccrm2cw7yw>

<https://youtu.be/ZLctgHyl5ug>

<https://youtu.be/4J1Tw518YSA>

<https://youtu.be/ko2gvNZxhEc>

<https://youtu.be/n6E40xBGyZM>

<https://youtu.be/HtXBV4-7Mls>

<https://youtu.be/qwouM75mJWQ>

<https://youtu.be/mWnIcgnu9cE>

<https://youtu.be/A1R3Wmm5KVQ>

<https://youtu.be/xV4wfz85ZuQ>

<https://youtu.be/AKObOoiXtGc>

<https://youtu.be/9aExLuLkVKo>

<https://youtu.be/iyHrP0mn8co>

<https://youtu.be/UY-OOJl8XuY>