JAVA TRAINING CENTER

Java Training Center

(No.1 in Training & placement)

java.lang WorkBook

Master the content…

W.B-3

Author

Som Prakash Rai

JAVA TRAINING CENTER

Topic

Jtc 1: Example using Object class Method.

Jtc 2: Example using equals Method.

Jtc 3: Example using cloning.

Jtc 4: Example for GC

Jtc 5: Example For Immutable Class

Jtc 6 – Jtc 19: Example using String Class

Jtc 20: Example using StringBuffer class .

Jtc 21: Example using StringBuffer class .

Jtc 22: Example using System class .

Jtc 23: Example using Runtime class .

Jtc 24: Example using Singleton class. (C)

Jtc 25: Example using Wrapper Class

java.lang APIs

Jtc 1: Example using Object class Method

```
package com.JTC.obj.test;
/*
* @Author    : Som Prakash Rai
* @Join      : Java Training Center
* @visit     : www.jtcindia.org
*@Call       :+91-9990399111
* */
public class Jtc1 {
public static void main(String[] args) {
System.out.println("**** hashCode *****");
System.out.println("=== Student ===");
Student st1 = new Student(99, "Som", 399111);
Student st2 = new Student(99, "Som", 399111);
Student st3 = new Student(88, “Prakash”, 399111);
Student st4 = st1;
System.out.println(st1.hashCode());
System.out.println(st2.hashCode());
System.out.println(st3.hashCode());
System.out.println(st4.hashCode());
System.out.println(st1 == st2);
System.out.println(st1 == st3);
System.out.println(st1 == st4);
System.out.println("=== Employee ===");
Employee emp1 = new Employee(99, "Som", 399111);
Employee emp2 = new Employee(99, "Som", 399111);
Employee emp3 = new Employee(88, “Prakash”, 7458965);
Employee emp4 = emp1;
System.out.println(emp1.hashCode());
System.out.println(emp2.hashCode());
System.out.println(emp3.hashCode());
System.out.println(emp4.hashCode());
System.out.println(emp1 == emp2);
System.out.println(emp1 == emp3);
System.out.println(emp1 == emp4);
System.out.println("\n**** getClass *****");
ClassService.showClassInfo("JTC");
ClassService.showClassInfo(st1);
ClassService.showClassInfo(emp1);
System.out.println("\n**** toString *****");
System.out.println(st1);
```

```
System.out.println(st2);
System.out.println(st1.toString());
System.out.println(st2.toString());
String str1 = "OK";
str1 = str1 + st1;
System.out.println(str1);
System.out.println(emp1);
System.out.println(emp3);
Employee emp5 = null;
System.out.println(emp5);
String str11 = "OK";
str11 = str11 + emp5;
System.out.println(str11);
//System.out.println(emp5.toString());
System.out.println("----------");
System.out.println(st1);
System.out.println(st1.toString());
String cName=st1.getClass().getName();
int hs=st1.hashCode();
String hx=Integer.toHexString(hs);
String msg=cName+"@"+hx;
System.out.println(msg);
}
}
class ClassService {
static void showClassInfo(Object obj) {
Class cl = obj.getClass();
System.out.println("\nName\t:" + cl.getName());
System.out.println("Name\t:" + cl.getSimpleName());
System.out.println("Super Class\t:" + cl.getSuperclass());
System.out.println("Package \t:" + cl.getPackage());
Class intrs[] = cl.getInterfaces();
for (int i = 0; i < intrs.length; i++) {
Class in = intrs[i];
System.out.println(in);
}
}
}
interface Inter1 {}
interface Inter2 {}
class Person {}
class Student extends Person implements Inter1, Inter2 {
```

```
int sid;
String name;
long phone;
public Student(int sid, String name, long phone) {
this.sid = sid;
this.name = name;
this.phone = phone;
}
}
class User {}
class Employee extends User {
int eid;
String name;
long phone;
public Employee(int eid, String name, long phone) {
this.eid = eid;
this.name = name;
this.phone = phone;
}
public String toString() {
return "Info Eid :" + eid + ", Name :" + name + ", Phone :" + phone;
}
public int hashCode() {
return (int) (phone << name.length());
}
}
```

Jtc 2: Example using equals Method

```
package edu.SP.obj.test;
/*
* @Author    : Som Prakash Rai
* @Join      : Java Training Center
* @visit     : www.jtcindia.org
*@Call       :+91-9990399111
* */
public class Jtc2 {
public static void main(String[] args) {
String st1 = "JTC";
String st2 = "JTC";
String st3 = new String("JTC");
System.out.println(st1 + "\t" + st2 + "\t" + st3);
```

```
System.out.println(st1 == st2);
System.out.println(st1 == st3);
System.out.println("--------");
System.out.println(st1.equals(st2));
System.out.println(st1.equals(st3));
System.out.println("-- Student ---");
Student stu1 = new Student(99, "SomPrakash");
Student stu2 = new Student(99, "SomPrakash");
System.out.println("==\t:" + (stu1 == stu2));
System.out.println("eq()\t:" + stu1.equals(stu2));
System.out.println("-- Employee ---");
Employee emp1 = new Employee(99, "SomPrakash");
Employee emp2 = new Employee(99, "SomPrakash");
Employee emp3 = new Employee(98, "SomPrakash");
System.out.println("==\t:" + (emp1 == emp2));
System.out.println("eq()\t:" + emp1.equals(emp2));
System.out.println("eq()\t:" + emp1.equals(emp3));
}
}
class Student {
int sid;
String name;
public Student(int sid, String name) {
this.sid = sid;
this.name = name;
}
}
class Employee {
int eid;
String name;
public Employee(int eid, String name) {
this.eid = eid;
this.name = name;
}
public boolean equals(Object obj) {
Employee emp = (Employee) obj;
return this.eid == emp.eid && this.name.equals(emp.name);
}
}
```

Jtc 3: Example using cloning

```
/*
* @Author : Som Prakash Rai
* @Join  : Java Training Center
* @Visit : www.jtcindia.org
*@Call   :+91-9990399111
* */
public class Jtc3 {
public static void main(String[] args) throws CloneNotSupportedException {
Employee emp1 = new Employee(88, "Manish");
emp1.showClone();
System.out.println("\n----------");
LoginInfo log = new LoginInfo(101, "somsree", "JTCindia");
Address ad = new Address("c-29", "Noida", 201301);
Student st = new Student(999, "SomPrakash", 6526668, ad, log);
System.out.println(st);
Student stud = (Student) st.clone();
System.out.println("\n*** After Cloning the Object ***");
System.out.println(st == stud);
System.out.println(st.studAdd == stud.studAdd);
System.out.println(st.login == stud.login);
System.out.println(st);
System.out.println(stud);
System.out.println("\n=== MODIFYING THE DATA ====");
stud.sid = 90909;
stud.name = "Manish";
stud.phone = 745896;
stud.studAdd.street = "DELHI";
stud.login.uname = "JTCuser";
System.out.println(st);
System.out.println(stud);
}
}
class Student  implements Cloneable
{
int sid;
String name;
long phone;
Address studAdd;
LoginInfo login;
public Student(int sid, String name, long phone, Address studAdd,LoginInfo login) {
```

```
this.sid = sid;
this.name = name;
this.phone = phone;
this.studAdd = studAdd;
this.login = login;
}
public String toString() {
return "\nStud Info\t:" + sid + "\t" + name + "\t" + phone+ "\nAdd Info\t:" + studAdd +
"\nLogin Info\t:" + login;
}
public Object clone() throws CloneNotSupportedException {
Object obj=null;
if(this instanceof Cloneable){
Address ad=new Address(this.studAdd.aid,this.studAdd.street,this.studAdd.pin);
LoginInfo info=new LoginInfo(this.login.loginId, this.login.uname, this.login.pwd);
obj=new Student(this.sid,this.name,this.phone,ad,info);
}else{
throw new CloneNotSupportedException(this.getClass().getName());
}
return obj;
}
}
class Employee implements Cloneable {
int eid;
String name;
public Employee(int eid, String name) {
this.eid = eid;
this.name = name;
System.out.println("-- Employee(int,String) --");
}
public String toString() {
return eid + "\t" + name;
}
void showClone() throws CloneNotSupportedException {
Employee ep = (Employee) clone();
System.out.println("this\t\t:" + this);
System.out.println("Cloned Obj\t:" + ep);
System.out.println(this == ep);
}
}

class Address {
```

```
String aid;
String street;
int pin;
public Address(String aid, String street, int pin) {
super();
this.aid = aid;
this.street = street;
this.pin = pin;
}
public String toString() {
return aid + "\t" + street + "\t" + pin;
}
}
class LoginInfo {
int loginId;
String uname;
String pwd;
public LoginInfo(int loginId, String uname, String pwd) {
this.loginId = loginId;
this.uname = uname;
this.pwd = pwd;
}
public String toString() {
return loginId + "\t" + uname + "\t" + pwd;
}
}
```

Jtc 4: Example for GC

```
package com.JTC.gc.test;
import java.sql.Connection;
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc4 {
public static void main(String[] args) {
System.out.println("-- IN Main --");
new Student(99);
Student st = new Student(88);
st.finalize();
System.out.println("Value of sid\t:"+st.sid);
```

```
st = null;
Student st1 = new Student(77);
st1 = new Student(66);
Student st2 = new Student(55);
Student st3 = new Student(44);
st2 = st3;
System.gc();
show();
for (int i = 0; i < 5; i++) {
System.out.println("I value\t:" + i);
}
Student st99 = new Student(33);
System.gc();
System.out.println("-- Main Completed --");
}
static void show() {
Student stud1 = new Student(101);
Student stud2 = new Student(202);
// Statements
}
}
final class Student {
int sid;
private Connection con = null;
Student(int sid) {
this.sid = sid;
// Code to Connect With DB
System.out.println("-- Connected with Resource --\t:" + sid);
}
void addInfo() {
// Using the con to intract with DB
}
void deleteInfo() {
// Using the con to intract with DB
}
public void finalize() {
// Code to close the DB Connection
System.out.println("** Resource is RELEASED **\t:" + sid);
}
}
```

Jtc 5: Example For Immutable Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc5 {
public static void main(String[] args) {
Address add = new Address(101, " c-29 Sector-2  noida ", "Noida", 201301);
Student stud = new Student(99, "SomPrakash", 6526668, add);
System.out.println(stud);
System.out.println("-- Modifying the add ref in Main --");
// stud.studAdd.aid = 1234;
add.aid = 14523;
add.street = "Sector 2  noida";
System.out.println(stud);
System.out.println(stud.getSid());
Address ref = stud.getStudAddress();
ref.aid = 11111;
System.out.println(stud);
}
}
final class Student {
private final int sid;
private final String name;
private final long phone;
private final Address studAdd;
public Student(int sid, String name, long phone, Address studAdd) {
this.sid = sid;
this.name = name;
this.phone = phone;
this.studAdd = new Address(studAdd.aid, studAdd.street, studAdd.city,studAdd.pin);
}
public String toString() {
return "Stud Info\t:" + sid + "\t" + name + "\t" + phone+ "\nAdd Info\t:" + studAdd;
}
public int getSid() {
return this.sid;
}

public Address getStudAddress(){
```

```
return (Address)studAdd.clone();
}
}
class Address{
int aid;
String street;
String city;
int pin;
public Address(int aid, String street, String city, int pin) {
this.aid = aid;
this.street = street;
this.city = city;
this.pin = pin;
}
public Object clone() {
return new Address(this.aid, this.street, this.city, this.pin);
}
public String toString() {
return aid + "\t" + street + "\t" + city + "\t" + pin;
}
}
```

Jtc 6: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc6 {
public static void main(String[] args) {
String str1 = "JTC";
String str2 = "JTC";
String str3 = "JTC";
System.out.println(str1 + "\t" + str2 + "\t" + str3);
System.out.println(str1 == str2);
System.out.println(str1 == str3);
System.out.println("----------");
String ref1 = new String("JTC");
String ref2 = new String("JTC");
System.out.println(ref1 + "\t" + ref2);
System.out.println(ref1 == ref2);
```

```
System.out.println(str1 == ref1);
System.out.println(str1 == ref2);
String ref3 = new String("SP");
String ref4 = new String("SP");
System.out.println(ref3 + "\t" + ref4);
System.out.println(ref3 == ref4);
}
}
```

Jtc 7: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc7 {
public static void main(String[] args) {
String st1 = new String("JTC");
String st2 = st1.intern();
System.out.println(st1 == st2);
String st3 = "JTC";
System.out.println(st3 == st2);
String st4 = "OK".intern();
String st5 = "OK";
System.out.println(st4 + "\t" + st5);
System.out.println(st4 == st5);
}
}
```

Jtc 8: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc8 {
public static void main(String[] args) {
String st1 = "JTC";
String st2 = "INDIA";
String st3 = st1 + st2;
String st4 = st1 + "INDIA";
String st5 = "JTC" + st2;
```

```
System.out.println(st3+"\t"+st4+"\t"+st5);
System.out.println(st3==st4);
System.out.println(st3==st5);
System.out.println(st4==st5);
String st6 = "JTC" + "INDIA";
final String st7 = "JTC";
final String st8 = "INDIA";
String st9 = st7 + st8;
System.out.println(st6+"\t"+st9);
System.out.println(st6==st9);
String st10="JTCINDIA";
System.out.println(st6==st10);
}
 }
```

Jtc 9: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc9 {
public static void main(String[] args) {
String st1 = "JTC";
String st2 = "JTC";
String st3 = new String("JTC");
String st4 = "JTC";
System.out.println(st1 + "\t" + st2 + "\t" + st3 + "\t" + st4);
System.out.println(st1 == st2);
System.out.println(st1 == st3);
System.out.println(st1.equals(st3));
System.out.println(st1.equals(st4));
System.out.println(st1.equalsIgnoreCase(st4));
System.out.println("----------------");
int ab = 94;
int bc = 76;
int res = ab - bc;
System.out.println(res);
System.out.println("ABC".compareTo("ABC"));
System.out.println("AEC".compareTo("ABC"));
System.out.println("ABH".compareTo("ADC"));
```

```
System.out.println("ABC".compareTo("ABCDEFGH"));
System.out.println("ABC".compareTo("ABMDEFGH"));
System.out.println("ABC".compareTo("Abc"));
System.out.println("ABC".compareToIgnoreCase("Abc"));
System.out.println("ABC".compareToIgnoreCase("Abg"));
}
}
```

Jtc 10: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc10 {
public static void main(String[] args) {
int ab = 10;
// String st1=ab;
// String st2=(String)ab;
String st3 = String.valueOf(ab);
System.out.println(st3);
boolean b1 = true;
String st4 = String.valueOf(b1);
System.out.println(st4);
long val = 1234L;
String st5 = String.valueOf(val);
System.out.println(st5);
Employee emp1 = null;
Employee emp2 = new Employee();
String st6 = String.valueOf(emp1);
String st7 = String.valueOf(emp2);
Person p = new Person();
String st8 = String.valueOf(p);
System.out.println(st6);
System.out.println(st7);
System.out.println(st8);
}
}
class Employee {}
class Person {
public String toString() {
return "Person Obj";
```

```
}
}
```

Jtc 11: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc11 {
public static void main(String[] args) {
String str = "Hi This is JTC. Full name Java Training Center. We are providing Best
training for Java  with Best SylJtcus. Jtc is no.1 in Training & placement,Thanks";
System.out.println(str);
System.out.println(str.startsWith("Hi"));
System.out.println(str.startsWith("Welcome"));
System.out.println(str.startsWith("Full"));
System.out.println(str.startsWith("Full", 16));
System.out.println(str.endsWith("placement"));
System.out.println(str.endsWith("Thanks"));
System.out.println("---------- ");
String st1 = str.replace('J', 'Y');
System.out.println(st1);
String st2 = str.replaceAll("Java", "XYZ");
System.out.println(st2);
String st3 = str.replaceFirst("Java", "XYZ");
System.out.println(st3);
System.out.println(str);
System.out.println("-----------");
String st4 = str.toLowerCase();
String st5 = str.toUpperCase();
System.out.println(st4);
System.out.println(st5);
}
}
```

Jtc 12: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc12 {
```

```
public static void main(String[] args) {
String str = " Hi This is JTC. Full name Java Training Center. We are providing Best training for Java  with Best Syllabus. Jtc is no.1 in Training & placement,Thanks ";
System.out.println(str.indexOf('X'));
System.out.println(str.indexOf('J'));
System.out.println(str.indexOf('J', 11));
System.out.println(str.indexOf('J', 12));
System.out.println(str.indexOf('J', 27));
System.out.println();
System.out.println(str.indexOf("XYZ"));
System.out.println(str.indexOf("Java"));
System.out.println(str.indexOf("Java", 26));
System.out.println(str.indexOf("Java", 27));
System.out.println("------");
System.out.println(str.lastIndexOf('J'));
System.out.println(str.lastIndexOf('J', 120));
System.out.println(str.lastIndexOf("Java"));
System.out.println(str.lastIndexOf("Java", 120));
System.out.println(str.lastIndexOf("Java", 15));
}
}
```

Jtc 13: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc13 {
public static void main(String[] args) {
String st = "  Hello  JTC Students   ";
System.out.println(st + ":\t:" + st.length());
String st1 = st.trim();
System.out.println(st1 + ":\t:" + st1.length());
System.out.println();
String str1 = "Hello JTC Students,";
String str2 = "Welcome to All";
String str = str1.concat(str2);
System.out.println(str);
System.out.println();
String st11 = str.substring(7);
System.out.println(st11);
```

```
String st12 = str.substring(7, 17); // 17 is lastIndex+1 (16+1)
System.out.println(st12);
// str.substring(7, 2);//Exception
System.out.println(st1.charAt(0));
System.out.println(st1.charAt(14));
System.out.println(st1.charAt(189));// Exception
}
}
```

Jtc 14: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc14 {
public static void main(String[] args) {
String st = "JTCINDIA";
char arr[] = st.toCharArray();
for (int i = 0; i < arr.length; i++) {
char c = arr[i];
System.out.println(i + "\t" + c);
}
System.out.println("-----");
byte bArr[] = st.getBytes();
for (int i = 0; i < bArr.length; i++) {
byte b = bArr[i];
System.out.println(i + "\t" + b + "\t" + (char) b);
}
System.out.println("-------");
char chArr[] = new char[10];
st.getChars(2, 6, chArr, 0);
for (int i = 0; i < chArr.length; i++) {
char c = chArr[i];
System.out.println(i + "\t" + c);
}
System.out.println("\n-------");
char chArr1[] = new char[10];
st.getChars(2, 6, chArr1, 4);
for (int i = 0; i < chArr1.length; i++) {
char c = chArr1[i];
System.out.println(i + "\t" + c);
```

```
}
}
}
```

Jtc 15: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc15 {
public static void main(String[] args) {
char chArr[] = { 'J', 'T', 'C', 'N', 'O', 'I', 'D', 'A' };
String st1 = new String(chArr);
System.out.println(st1);
String st2 = new String(chArr, 2, 3);// 3 is number of character
System.out.println(st2);
System.out.println();
byte bArr[] = { 65, 89, 97, 78, 120 };
String st3 = new String(bArr);
System.out.println(st3);
String st4 = new String(bArr, 1, 3);// 3 is number of character
System.out.println(st4);
}
}
```

Jtc 16: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc16 {
public static void main(String[] args) {
String str = "Hi this is Java Training Center.No 1 in training and placement.Providing
training on java and java related concepts";
System.out.println(str);
String res[] = str.split("java");
for (int i = 0; i < res.length; i++) {
String st = res[i];
System.out.println(i + "\t" + st);
}
```

```
System.out.println("----------");
res = str.split("java", 2);
for (int i = 0; i < res.length; i++) {
String st = res[i];
System.out.println(i + "\t" + st);
}
}
}
```

Jtc 17: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc17 {
public static void main(String[] args) {
String st = "BBCmni45";
String pattern = "^[A-H][A-Za-z]*"; // RegularExpression pattern
System.out.println(st.matches(pattern));
String st1 = "ABCmni987";
String pattern1 = "^[A-H][A-Za-z0-9]*"; // RegularExpression pattern
System.out.println(st1.matches(pattern1));
System.out.println("-----------");
String st11 = "JTC is No 1 in Noida";
String st12 = "welcome to JTCNoida";
String st13 = "welcome to JTCNoida";
System.out.println(st11.regionMatches(12, st12, 14, 5));
System.out.println(st11.regionMatches(12, st12, 14, 2));
System.out.println(st11.regionMatches(12, st13, 14, 2));
System.out.println(st11.regionMatches(false, 12, st13, 14, 2));
System.out.println(st11.regionMatches(true, 12, st13, 14, 2));
}
}
```

Jtc 18: Example using String Class (From Java 5)

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc18 {
```

```
public static void main(String[] args) {
int ab = 10;
int bc = 23;
String str = String.format("Sum of %d and %d is %d", ab, bc, (ab + bc));
System.out.println(str);
int val = 97;
String st = String.format("%d %h %o %c", val, val, val, val);
System.out.println(st);
System.out.printf("%d %h %o %c", val, val, val, val);
}
}
```

Jtc 19: Example using String Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc19 {
public static void main(String[] args) {
String st1 = "JTC";
String st2 = "JTC";
System.out.println(st1 == st2);
System.out.println("Result is :" + st1 == st2);
System.out.println("Result is :" + (st1 == st2));
}
}
```

Jtc 20: Example using StringBuffer Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc20 {
public static void main(String[] args) {
StringBuffer sb = new StringBuffer();
System.out.println(sb.length());
System.out.println(sb.capacity());
sb.append(true);
sb.append(1234);
sb.append(985.251);
```

```
System.out.println();
System.out.println(sb.length());
System.out.println(sb.capacity());
sb.append('A');
sb.append(new Student());
System.out.println();
System.out.println(sb.length());
System.out.println(sb.capacity());
System.out.println("------------");
StringBuffer sb1 = new StringBuffer(30);
System.out.println(sb1.length());
System.out.println(sb1.capacity());
System.out.println("------------");
StringBuffer sb2 = new StringBuffer("HELLO");
System.out.println(sb2.length());
System.out.println(sb2.capacity());
System.out.println("----");
StringBuffer sb3 = new StringBuffer();
System.out.println(sb3.length());
System.out.println(sb3.capacity());
sb3.append("qwertyuioplkjhgfdsazxcvbnm1234567mnbghjkiuytredfr");
System.out.println(sb3.length());
System.out.println(sb3.capacity());
}
}
class Student {}
```

Jtc 21: Example using StringBuffer Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc21 {
public static void main(String[] args) {
StringBuffer sb = new StringBuffer("JTC");
char arr[] = { 'A', 'B', 'C', 'D' };
sb.append(true);
sb.append(123);
sb.append(arr);
System.out.println(sb);
```

```
sb.reverse();
System.out.println(sb);
sb.deleteCharAt(4);
System.out.println(sb);
System.out.println(sb.length());
sb.setLength(20);
System.out.println(sb.length());
System.out.println(sb);
sb.append("SomPrakash");
System.out.println(sb);
System.out.println(sb.length());
System.out.println(sb.capacity());
sb.setLength(7);
System.out.println(sb);
System.out.println(sb.length());
System.out.println(sb.capacity());
sb.insert(4, "JAVATRAININGCENTER");
System.out.println(sb);
System.out.println("------------");
String str = "ABC";
StringBuffer sbo = new StringBuffer("ABC");
System.out.println(str.equals(sbo));
System.out.println(str.contentEquals(sbo));
System.out.println();
StringBuffer obj=new StringBuffer("OKJTC");
System.out.println(obj.length());
System.out.println(obj.capacity());
obj.trimToSize();
System.out.println(obj.length());
System.out.println(obj.capacity());
}
}
```

Jtc 22: Example using System Class

```
import java.io.*;
import java.util.*;
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
```

```
* */
public class Jtc22 {
public static void main(String[] args) throws Exception {
System sys = null;
// sys = new System();
System.out.println("OUT MSG 1");
System.out.println("OUT MSG 2");
System.err.println("ERR MSG 1");
System.err.println("ERR MSG 2");
// System.out=null;
long ms = System.currentTimeMillis();
System.out.println(new Date(ms));
System.out.println(System.getenv("path")); // From Java 5
System.out.println("**********");
Map values = System.getenv();
Iterator it = values.entrySet().iterator();
while (it.hasNext()) {
Map.Entry ent = (Map.Entry) it.next();
System.out.println(ent.getKey() + "\t\t" + ent.getValue());
}
System.out.println("\n****** Properties Name ******\n");
System.setProperty("java.ins.name", "Java Training Center");
System.out.println(System.getProperty("os.name"));
System.setProperty("os.name", "UNIX");
System.out.println(System.getProperty("os.name"));
Properties p = System.getProperties();
Enumeration enu = p.propertyNames();
while (enu.hasMoreElements()) {
String nm = (String) enu.nextElement();
System.out.println(nm + "\t\t" + p.getProperty(nm));
}
System.out.println("**********");
System.setProperty("runtim.test", "true");
System.out.println(Boolean.getBoolean("true"));
System.out.println(Boolean.getBoolean("runtim.test"));
System.out.println("**********");
System.out.println("----------");
System.out.println("Message Before Changing");
System.err.println(System.out);
System.err.println(System.err);
System.err.println(System.out == System.err);
System.setOut(new PrintStream(new FileOutputStream("res.txt")));
```

```
System.out.println("Message After Changing 1");
System.out.println("Message After Changing 2");
System.err.println("ERR MSG After changing Out");
System.exit(0);
System.err.println("Error");
for (int i = 0; i < 10; i++) {
System.err.println("Value is \t:" + i);
}
}
}
```

Jtc 23: Example using Runtime Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public class Jtc23 {
public static void main(String[] args) throws Exception {
Runtime rt = null;
// rt=new Runtime();
// rt.avaiJtcleProcessors();
rt = Runtime.getRuntime();
System.out.println(rt);
System.out.println(Runtime.getRuntime());
System.out.println(Runtime.getRuntime());
System.out.println(Runtime.getRuntime());
System.out.println(rt.totalMemory());
System.out.println(rt.freeMemory());
System.out.println(rt.maxMemory());
System.out.println(rt.avaiJtcleProcessors());
// No_Of_Processors        env  value
Process p1 = rt.exec("notepad");
Process p2 = rt.exec("calc");
Process p3 = rt.exec("mspaint");
System.out.println(System.getenv("path"));
Process p4 = rt.exec("C:\\Program Files\\Windows Media Player\\wmplayer");
// rt.exec("res.txt");
System.out.println("Press Enter To Close Calculator");
System.in.read();
p1.destroy();
p2.destroy();
```

```
p3.destroy();
p4.destroy();
}
}
/*
java -Xms256m -Xmx1024m RuntimeTest
*/
```

Jtc 25: Example using Wrapper Class

```
/*
* @Author : Som Prakash Rai
* @Join: Java Training Center
* @Visit : www.jtcindia.org
* */
public final class Jtc25 {
public static void main(String[] args) {
/* Primitive to Wrapper */
Integer in1 = new Integer(987);
Boolean b1 = new Boolean(true);
Character ch1 = new Character('A');
Byte by11 = new Byte((byte) 12);
Integer in2 = Integer.valueOf(123);
Character ch2 = Character.valueOf('A');
Boolean b2 = Boolean.valueOf(false);

/* Wrapper to Primitive */
byte byVal = in1.byteValue();
short shVal = in1.shortValue();
int iVal = in1.intValue();
long lnVal = in1.longValue();
float fVal = in1.floatValue();
double dVal = in1.doubleValue();

char chVal = ch1.charValue();
boolean bVal = b1.booleanValue();

/* Primitive To String */
int val = 12345;
String st1 = val + "";
String st2 = "" + val;
String st3 = String.valueOf(val);
```

```
String st4 = Integer.toString(val);

/* String to Primitive */
int ab = Integer.parseInt("765");
byte by1 = Byte.parseByte("123");
// int bc=Integer.parseInt("123.90");
// int cd=Integer.parseInt("A");
// byte by2=Byte.parseByte("128");
boolean b3 = Boolean.parseBoolean("OK");// From Java 5

/* String to Wrapper */
Integer in3 = new Integer("1234");
// Integer in4 = new Integer("JTC");
Boolean b4 = new Boolean("Som");
Boolean b5 = new Boolean("TrUE");

/* Wrapper to String */
String str1 = in1.toString();
String str2 = ch1.toString();
String str3 = b1.toString();

/* OTHER METHODS */
String binStr = Integer.toBinaryString(6567533);
System.out.println(binStr);
String hexStr = Integer.toHexString(6567533);
System.out.println(hexStr);
String octStr = Integer.toOctalString(6567533);
System.out.println(octStr);
Integer in11 = new Integer(123);
Integer in12 = new Integer(123);
System.out.println(in11.compareTo(in12));
System.out.println(in11.equals(in12));
System.out.println(Character.isAlphabetic('A'));
System.out.println(Character.isAlphabetic('9'));
System.out.println(Character.iSPigit('4'));
System.out.println(Character.iSPigit('j'));
System.out.println(Character.isMirrored('M'));
System.out.println(Character.isMirrored('('));
System.out.println(Character.isMirrored('{'));
}
}
```

java.lang.Object

```
public class java.lang.Object{
public java.lang.Object();
public final native java.lang.Class getClass();
public native int hashCode();
public boolean equals(java.lang.Object);
protected native java.lang.Object clone()  throws java.lang.CloneNotSupportedException;
public java.lang.String toString();
public final native void notify();
public final native void notifyAll();
public final native void wait(long)       throws java.lang.InterruptedException;
public final void wait(long, int)       throws java.lang.InterruptedException;
public final void wait()       throws java.lang.InterruptedException;
protected void finalize()       throws java.lang.Throwable;
}
```

java.lang.String

```
public final class java.lang.String extends java.lang.Object implements java.io.
Serializable,java.lang.Comparable,java.lang.CharSequence{
public java.lang.String();
public java.lang.String(java.lang.String);
public java.lang.String(char[]);
public java.lang.String(byte[], int, int, int);
public java.lang.String(java.lang.StringBuffer);
public java.lang.String(java.lang.StringBuilder);
public int length();
public boolean isEmpty();
public char charAt(int);
void getChars(char[], int);
public void getChars(int, int, char[], int);
public void getBytes(int, int, byte[], int);
public boolean equals(java.lang.Object);
public boolean contentEquals(java.lang.StringBuffer);
public boolean contentEquals(java.lang.CharSequence);
public boolean equalsIgnoreCase(java.lang.String);
```

```
public int compareTo(java.lang.String);
public int compareToIgnoreCase(java.lang.String);
public boolean regionMatches(int, java.lang.String, int, int);
public boolean regionMatches(boolean, int, java.lang.String, int, int);
public boolean startsWith(java.lang.String, int);
public boolean startsWith(java.lang.String);
public boolean endsWith(java.lang.String);
public int hashCode();
public int indexOf(int);
public int indexOf(int, int);
public int lastIndexOf(int);
public int lastIndexOf(int, int);
public int indexOf(java.lang.String);
public int indexOf(java.lang.String, int);
public int lastIndexOf(java.lang.String);
public int lastIndexOf(java.lang.String, int);
public java.lang.String substring(int);
public java.lang.String substring(int, int);
public java.lang.CharSequence subSequence(int, int);
public java.lang.String concat(java.lang.String);
public java.lang.String replace(char, char);
public boolean matches(java.lang.String);
public java.lang.String replaceFirst(java.lang.String, java.lang.String);
public java.lang.String replaceAll(java.lang.String, java.lang.String);
public java.lang.String[] split(java.lang.String, int);
public java.lang.String[] split(java.lang.String);
public java.lang.String toLowerCase(java.util.Locale);
public java.lang.String toLowerCase();
public java.lang.String toUpperCase(java.util.Locale);
public java.lang.String toUpperCase();
public java.lang.String trim();
public java.lang.String toString();
public char[] toCharArray();
public static java.lang.String format(java.lang.String, java.lang.Object[]);
public static java.lang.String valueOf(java.lang.Object);
public static java.lang.String valueOf(double);  //Overloaded Method
public native java.lang.String intern();
public int compareTo(java.lang.Object);
}
```

java.lang.StringBuffer

```
public final class java.lang.StringBuffer extends java.lang.AbstractStringBuilder
implements java.io.Serializable,java.lang.CharSequence{
public java.lang.StringBuffer();
public java.lang.StringBuffer(int);
public java.lang.StringBuffer(java.lang.String);
public java.lang.StringBuffer(java.lang.CharSequence);
public synchronized int length();
public synchronized int capacity();
public synchronized void ensureCapacity(int);
public synchronized void trimToSize();
public synchronized void setLength(int);
public synchronized char charAt(int);
public synchronized void getChars(int, int, char[], int);
public synchronized void setCharAt(int, char);
public synchronized java.lang.StringBuffer append(java.lang.Object);
//  Overloaded Method

public synchronized java.lang.StringBuffer delete(int, int);
public synchronized java.lang.StringBuffer deleteCharAt(int);
public synchronized java.lang.StringBuffer replace(int, int, java.lang.String);
public synchronized java.lang.String substring(int);
public synchronized java.lang.CharSequence subSequence(int, int);
public synchronized java.lang.String substring(int, int);
public java.lang.StringBuffer insert(int, boolean);
// Overloaded Method

public int indexOf(java.lang.String);
public synchronized int indexOf(java.lang.String, int);
public int lastIndexOf(java.lang.String);
public synchronized int lastIndexOf(java.lang.String, int);
public synchronized java.lang.StringBuffer reverse();
public synchronized java.lang.String toString();
}
```

java.lang.Number

```
public abstract class java.lang.Number extends java.lang.Object implements
java.io.Serializable{
public java.lang.Number();
```

```
public abstract int intValue();
public abstract long longValue();
public abstract float floatValue();
public abstract double doubleValue();
public byte byteValue();
public short shortValue();
}
```

java.lang.Boolean

```
public final class java.lang.Boolean extends java.lang.Object implements
java.io.Serializable,java.lang.Comparable{
public static final java.lang.Boolean TRUE;
public static final java.lang.Boolean FALSE;
public static final java.lang.Class TYPE;                    //From Java 5
public java.lang.Boolean(boolean);
public java.lang.Boolean(java.lang.String);
public static boolean parseBoolean(java.lang.String);  //From Java 5
public boolean booleanValue();
public static java.lang.Boolean valueOf(boolean);
public static java.lang.Boolean valueOf(java.lang.String);
public static java.lang.String toString(boolean);
public java.lang.String toString();
public int hashCode();
public boolean equals(java.lang.Object);
public static boolean getBoolean(java.lang.String);
public int compareTo(java.lang.Boolean);
public int compareTo(java.lang.Object);
}
```

java.lang.Character

```
public final class java.lang.Character extends java.lang.Object implements java.
io.Serializable,java.lang.Comparable{
public static final int SIZE;
public java.lang.Character(char);
public static java.lang.Character valueOf(char);
public char charValue();
public int hashCode();
public boolean equals(java.lang.Object);
public java.lang.String toString();
public static java.lang.String toString(char);
```

```
public static boolean isLowerCase(char);
public static boolean isUpperCase(char);
public static boolean isTitleCase(char);
public static boolean iSPigit(char);
public static boolean isLetter(char);
public static boolean isLetterOrDigit(char);
public static boolean isJavaLetter(char);
public static boolean isJavaLetterOrDigit(char);
public static boolean isJavaIdentifierStart(char);
public static boolean isJavaIdentifierPart(char);
public static char toLowerCase(char);
public static char toUpperCase(char);
public static int getNumericValue(char);
public static boolean isSpaceChar(char);
public static boolean isWhitespace(char);
public static boolean isISOControl(char);
public static boolean isMirrored(char);
public int compareTo(java.lang.Character);
public int compareTo(java.lang.Object);
}
```

java.lang.Integer

```
public final class java.lang.Integer extends java.lang.Number implements java.la
ng.Comparable{
public static final int MIN_VALUE;
public static final int MAX_VALUE;
public static final java.lang.Class TYPE;
public static final int SIZE;
public static java.lang.String toHexString(int);
public static java.lang.String toOctalString(int);
public static java.lang.String toBinaryString(int);
public static java.lang.String toString(int);
public static int parseInt(java.lang.String, int)
public static int parseInt(java.lang.String)
public static java.lang.Integer valueOf(java.lang.String)
public static java.lang.Integer valueOf(int);
public java.lang.Integer(int);
public java.lang.Integer(java.lang.String)
public byte byteValue();
public short shortValue();
```

```
public int intValue();
public long longValue();
public float floatValue();
public double doubleValue();
public java.lang.String toString();
public int hashCode();
public boolean equals(java.lang.Object);
public static java.lang.Integer getInteger(java.lang.String);
public int compareTo(java.lang.Integer);
public int compareTo(java.lang.Object);
}
```

java.lang.Long

```
public final class java.lang.Long extends java.lang.Number implements java.lang.
Comparable{
public static final long MIN_VALUE;
public static final long MAX_VALUE;
public static final java.lang.Class TYPE;
public static final int SIZE;
public static java.lang.String toString(long, int);
public static java.lang.String toHexString(long);
public static java.lang.String toOctalString(long);
public static java.lang.String toBinaryString(long);
public static java.lang.String toString(long);
public static long parseLong(java.lang.String, int)
public static long parseLong(java.lang.String)
public static java.lang.Long valueOf(java.lang.String, int)
public static java.lang.Long valueOf(java.lang.String)
public static java.lang.Long valueOf(long);
public static java.lang.Long decode(java.lang.String)
public java.lang.Long(long);
public java.lang.Long(java.lang.String
public byte byteValue();
public short shortValue();
public int intValue();
public long longValue();
public float floatValue();
public double doubleValue();
public java.lang.String toString();
public int hashCode();
```

```
public boolean equals(java.lang.Object);
public static java.lang.Long getLong(java.lang.String);
public int compareTo(java.lang.Long);
}
```

java.lang.Byte

```
public final class java.lang.Byte extends java.lang.Number implements java.lang.
Comparable{
public static final byte MIN_VALUE;
public static final byte MAX_VALUE;
public static final java.lang.Class TYPE;
public static final int SIZE;
public static java.lang.String toString(byte);
public static java.lang.Byte valueOf(byte);
public static byte parseByte(java.lang.String)
public static byte parseByte(java.lang.String, int)
public static java.lang.Byte valueOf(java.lang.String, int)
public static java.lang.Byte valueOf(java.lang.String
public java.lang.Byte(byte);
public java.lang.Byte(java.lang.String)
public byte byteValue();
public short shortValue();
public int intValue();
public long longValue();
public float floatValue();
public double doubleValue();
public java.lang.String toString();
public int hashCode();
public boolean equals(java.lang.Object);
public int compareTo(java.lang.Byte);
public int compareTo(java.lang.Object);
}
```

java.lang.Short

```
public final class java.lang.Short extends java.lang.Number implements
java.lang.Comparable{
public static final short MIN_VALUE;
public static final short MAX_VALUE;
```

```
public static final java.lang.Class TYPE;
public static final int SIZE;
public static java.lang.String toString(short);
public static short parseShort(java.lang.String)
public static short parseShort(java.lang.String, int)
public static java.lang.Short valueOf(java.lang.String, int)
public static java.lang.Short valueOf(java.lang.String)
public static java.lang.Short valueOf(short);
public static java.lang.Short decode(java.lang.String)
public java.lang.Short(short);
public java.lang.Short(java.lang.String)
public byte byteValue();
public short shortValue();
public int intValue();
public long longValue();
public float floatValue();
public double doubleValue();
public java.lang.String toString();
public int hashCode();
public boolean equals(java.lang.Object);
public int compareTo(java.lang.Short);
public int compareTo(java.lang.Object);
}
```

java.lang.Float

```
public final class java.lang.Float extends java.lang.Number implements
java.lang.Comparable{
public static final float POSITIVE_INFINITY;
public static final float NEGATIVE_INFINITY;
public static final float NaN;
public static final float MAX_VALUE;
public static final float MIN_VALUE;
public static final int SIZE;
public static final java.lang.Class TYPE;
public static java.lang.String toString(float);
public static java.lang.String toHexString(float);
public static java.lang.Float valueOf(java.lang.String)
public static java.lang.Float valueOf(float);
public static float parseFloat(java.lang.String)
public static boolean isNaN(float);
public static boolean isInfinite(float);
```

```
public java.lang.Float(float);
public java.lang.Float(double);
public java.lang.Float(java.lang.String)
public boolean isNaN();
public boolean isInfinite();
public java.lang.String toString();
public byte byteValue();
public short shortValue();
public int intValue();
public long longValue();
public float floatValue();
public double doubleValue();
public int hashCode();
public boolean equals(java.lang.Object);
public int compareTo(java.lang.Float);
public int compareTo(java.lang.Object);
}
```

java.lang.Double

```
public final class java.lang.Double extends java.lang.Number implements
java.lang.Comparable{
public static final double POSITIVE_INFINITY;
public static final double NEGATIVE_INFINITY;
public static final double NaN;
public static final double MAX_VALUE;
public static final double MIN_VALUE;
public static final int SIZE;
public static final java.lang.Class TYPE;
public static java.lang.String toString(double);
public static java.lang.String toHexString(double);
public static java.lang.Double valueOf(java.lang.String)
public static java.lang.Double valueOf(double);
public static double parseDouble(java.lang.String)
public static boolean isNaN(double);
public static boolean isInfinite(double);
public java.lang.Double(double);
public java.lang.Double(java.lang.String)
public boolean isNaN();
public boolean isInfinite();
public java.lang.String toString();
public byte byteValue();
```

```
public short shortValue();
public int intValue();
public long longValue();
public float floatValue();
public double doubleValue();
public int hashCode();
public boolean equals(java.lang.Object);
public int compareTo(java.lang.Double);
public int compareTo(java.lang.Object);
}
```

java.lang.System

```
public final class java.lang.System extends java.lang.Object{
public static final java.io.InputStream in;
public static final java.io.PrintStream out;
public static final java.io.PrintStream err;
public static void setIn(java.io.InputStream);
public static void setOut(java.io.PrintStream);
public static void setErr(java.io.PrintStream);
public static java.io.Console console();
public static void setSecurityManager(java.lang.SecurityManager);
public static java.lang.SecurityManager getSecurityManager();
public static native long currentTimeMillis();
public static native long nanoTime();
public static native void arraycopy(java.lang.Object, int, java.lang.Object,
int, int);
public static java.util.Properties getProperties();
public static void setProperties(java.util.Properties);
public static java.lang.String getProperty(java.lang.String);
public static java.lang.String getProperty(java.lang.String, java.lang.String);
public static java.lang.String setProperty(java.lang.String, java.lang.String);
public static java.lang.String clearProperty(java.lang.String);
public static java.lang.String getenv(java.lang.String);
public static java.util.Map getenv();
public static void exit(int);
public static void gc();
public static void runFinalization();
public static void load(java.lang.String);
public static void loadLibrary(java.lang.String);
}
```

java.lang.Runtime

```
public class java.lang.Runtime extends java.lang.Object{
public static java.lang.Runtime getRuntime();
public void exit(int);
public void addShutdownHook(java.lang.Thread);
public boolean removeShutdownHook(java.lang.Thread);
public void halt(int);
public java.lang.Process exec(java.lang.String)
public java.lang.Process exec(java.lang.String, java.lang.String
public native int avaiJtcleProcessors();
public native long freeMemory();
public native long totalMemory();
public native long maxMemory();
public native void gc();
public void runFinalization();
public void load(java.lang.String);
public void loadLibrary(java.lang.String);
}
```

java.lang.Math

```
public final class java.lang.Math extends java.lang.Object{
public static final double E;
public static final double PI;
public static double sin(double);
public static double cos(double);
public static double tan(double);
public static double asin(double);
public static double acos(double);
public static double atan(double);
public static double toRadians(double);
public static double toDegrees(double);
public static double exp(double);
public static double log(double);
public static double sqrt(double);
public static double cbrt(double);
public static double ceil(double);
public static double floor(double);
public static double pow(double, double);
public static long round(double);
public static double random();
public static int abs(int);
```

```
public static long abs(long);
public static double abs(double);
public static int max(int, int);
public static long max(long, long);
public static double max(double, double);
public static int min(int, int);
public static long min(long, long);
public static double min(double, double);
public static double sinh(double);
public static double cosh(double);
public static double tanh(double);
}
```