######################################################################

class Demo{

public static void main(String ss[]){

System.out.println(" Welcome to Java! ");

}

}

######################################################################

// Instance Variable and Instance Method

class Student{

int id;

void display(){

System.out.println(" Result : "+id);

}

}

class Demo1{

public static void main(String aa[]){

//Student.id=900;

Student s1 = new Student();

s1.id=100;

s1.display();

System.out.println("----------------------");

Student s2 = new Student();

s2.id=200;

s2.display();

}

}

######################################################################

class Employee{

static int id;

static void test(){

System.out.println(" X : "+id);

}

}

class Demo2{

public static void main(String aa[]){

Employee.id=2000;

Employee.test();

}

}

######################################################################

class Test{

int x;

void add(){

System.out.println(" Class Test "+x);

}

}

class Demo3{

public static void main(String aa[]){

Test t1 = new Test();

t1.x=100;

t1.add();

t1.add();

t1.add();

new Test().x=500; //Anonymous Object

new Test().add();

new Test().add();

new Test().add();

}

}

######################################################################

public class TypeCasting {

public static void main(String[] args) {

int a = 100;

char b = (char)a;

System.out.println("Int into Char -----> "+b+"\n");

double d1 = 100.04;

long l1 = (long)d1; //explicit type casting required

int i1 = (int)l1; //explicit type casting required

System.out.println("Double value "+d1);

System.out.println("Long value "+l1);

System.out.println("Int value "+i1);

System.out.println("");

int i2 = 100;

long l2 = (long)i2; //no explicit type casting required

float f2 = (float)l2; //no explicit type casting required

System.out.println("Int value "+i2);

System.out.println("Long value "+l2);

System.out.println("Float value "+f2);

}

}

######################################################################

class A{

int x;

void add(){

System.out.println(" X : "+x);

}

}

class B extends A{

int y;

void sum(){

System.out.println(" X : "+x+" Y : "+y);

}

}

class Demo4{

public static void main(String aa[]){

B s1 = new B();

s1.x=100;

s1.add();

s1.y=200;

s1.sum();

System.out.println("----------------------");

A s2 = new B();

s2.x=300;

s2.add();

//s2.sum();

System.out.println("----------------------");

B s3 = (B)s2;

s3.add();

s3.y=400;

s3.sum();

}

}

######################################################################

class A{

int x;

void add(){

System.out.println(" X : "+x);

}

}

class B extends A{

int y;

void sum(){

System.out.println(" X : "+x+" Y : "+y);

}

}

class Demo5{

public static void main(String aa[]){

B s1 = new B();

s1.x=100;

s1.add();

s1.y=200;

s1.sum();

System.out.println("----------------------");

A s2 = s1;

s2.add();

//s2.sum();

System.out.println("----------------------");

B s3 = (B)s2;

s3.add();

s3.sum();

}

}

######################################################################

// Inheritance

class Test1{

int x;

void show1(){

System.out.println(" X : "+x);

}

}

class Test2{

int y;

void show2(){

System.out.println(" Y : "+y);

}

}

class Test3{

int z;

void show3(){

System.out.println(" Z : "+z);

}

}

class Demo6{

public static void main(String aa[]){

Test1 ob1=new Test1();

ob1.x=100;

ob1.show1();

System.out.println("------------------------------------");

Test2 ob2=new Test2();

ob2.y=200;

ob2.show2();

System.out.println("------------------------------------");

Test3 ob3=new Test3();

ob3.z=300;

ob3.show3();

Object r1 = new Test1();

Object r2 = ob1;

Object r3 = new Test2();

Object r4 = ob2;

Object r5 = new Test3();

Object r6 = ob3;

System.out.println("------------------------------------");

Test1 ob4=(Test1)r1;

ob4.x=400;

ob4.show1();

Test1 ob5=(Test1)r2;

ob5.show1();

System.out.println("------------------------------------");

Test2 ob6=(Test2)r3;

ob6.y=500;

ob6.show2();

Test2 ob7=(Test2)r4;

ob7.show2();

System.out.println("------------------------------------");

Test3 ob8=(Test3)r5;

ob8.z=600;

ob8.show3();

Test3 ob9=(Test3)r6;

ob9.show3();

System.out.println("------------------------------------");

}

}

######################################################################

// Inheritance (Only upcasting)

class A{

int x;

void add(){

System.out.println(" X : "+x);

}

}

class B extends A{

int y;

void sum(){

System.out.println(" Y : "+y);

}

}

class Demo6{

public static void main(String aa[]){

B s1=new B();

s1.x=100;

s1.y=200;

s1.add();

s1.sum();

System.out.println("----------------------");

A s2=new B();

s2.x=300;

s2.add();

System.out.println("----------------------");

Object s3=new B();

}

}

######################################################################

// Class casting (Upcasting and Downcasting)

class A{

int x;

void add(){

System.out.println(" X : "+x);

}

}

class B extends A{

int y;

void sum(){

System.out.println(" Y : "+y);

}

}

class Demo7{

public static void main(String aa[]){

B s1=new B();

s1.x=100;

s1.y=200;

s1.add();

s1.sum();

System.out.println("----------------------");

A s2=s1;

s2.add();

System.out.println("----------------------");

Object s3=s2;

System.out.println("----------------------");

A s4=(A)s3;

s4.add();

System.out.println("----------------------");

B s5=(B)s4;

s5.add();

s5.sum();

}

}

######################################################################