public class Prog17 {

public static void main(String[] args) {

test(101);

test(new Integer(101));

}

public static void test(Object iObject) {

System.out.println("Object=" + iObject);

}

public static void test(Integer iObject) {

System.out.println("Integer=" + iObject);

}

public static void test(int iValue) {

System.out.println("int=" + iValue);

}

}

Int 101

Integer 101

public class Prog18 {

public static void main(String[] args) {

test(101);

// test(new Integer(101));

}

public static void test(Float lObject) {

System.out.println("Float =" + lObject);

}

public static void test(long lValue) {

System.out.println("long=" + lValue);

}

}

Long 101

public class Prog19 {

public static void main(String[] args) {

test(101L);

test(new Long(101L));

}

public static void test(long lObject) {

System.out.println("LongValue=" + lObject);

}

public static void test(Long lobj) {

System.out.println("long Object " +lobj);

}

public static void test(Object obj) {

System.out.println("Object=" + obj);

}

}

Long Value

Long Object

import java.util.Map;

import java.util.HashMap;

public class Prog20 {

public static void main(String[] args) {

Map m = new HashMap();

m.put(1, "Sanju");

m.put(2, "Niki");

m.put(3, "Nishtha");

m.put(4, "Kolage");

m.put(5, "Salve");

m.put(6, "Babbar");

// m.put(null, "Hi");

String res = (String) m.getOrDefault(6, "Not Found");

System.out.println(res);

}

}

Babbar

import java.util.ArrayList;

import java.util.List;

public class Prog21 {

public static void main(String[] args) {

List<Integer> list = new ArrayList<>();

list.add(101);

list.add(102);// auto boxing will work

int aValue = list.get(1); // auto unboxing will work, too

System.out.println(aValue);

}

}

102

public class Prog22 {

static {

try {throw new ArithmeticException("my exception");

}

catch(Exception e){

System.out.println("2");

}

}

public static void main(String[] args){

System.out.println("1");

}

}

2

1

public class Prog22\_1 {

static {

try {throw new Exception("my exception");

}

catch(Exception e){

System.out.println("2");

}

}

public static void main(String[] args){

System.out.println("1");

}

}

2

1

public class Quiz4 {

public static void main(String[] args) {

test(101);

}

public static void test(Object ob) {

System.out.println("Object " +ob);

}

public static void test(Integer ob) {

System.out.println("Integer " +ob);

}

}

Integer 101

public class Quiz5 {

public static void main(String[] args) {

test(101.9F);

}

public static void test(Object ob) {

System.out.println("Object " +ob);

}

public static void test(double ob) {

System.out.println("double " +ob);

}

}

Double 101.9

class First {

public void show(int x) {

System.out.println("Hello " +x);

}

public int show(int x, int y) {

return x+y;

}

}

class Second extends First {

// public int show(int x) {

// return x;

// }

public int show(int x,int y, int z) {

return x+y+z;

}

}

public class Quiz8 {

public static void main(String[] args) {

new Second().show(12,77,567);

}

}

Compile Time error, same method with different data types not possible

public class Quiz10 {

public static void main(String[] args) {

int x;

System.out.println(x);

}

}

Compile Time error, as use of unassigned variable

public class Quiz11 {

static int x;

public void incr() {

++x;

System.out.println(x);

}

public static void main(String[] args) {

Quiz11 obj1 = new Quiz11();

Quiz11 obj2 = new Quiz11();

Quiz11 obj3 = new Quiz11();

obj1.incr();

obj2.incr();

obj3.incr();

}

}

1 2 3

public class Quiz12 {

public static void main(String[] args) {

System.out.println("5" +3+8);

System.out.println("5 + 3" +8);

System.out.println("5" + (3+8));

}

}

538

5+38

511

public class Prog17 {

public static void main(String[] args) {

test(101);

test(new Integer(101));

}

public static void test(Object iObject) {

System.out.println("Object =" + iObject);

}

public static void test(int iValue) {

System.out.println("int=" + iValue);

}

}

public class Quiz18 {

int x;

public static void main(String[] args) {

System.out.println(new Quiz18().x);

}

}

0 as class variable integer default value 0

class First {

public void show(int x) {

System.out.println("Hello " +x);

}

}

class Second extends First {

public int show(int x) {

return x+5;

}

}

public class Quiz21 {

public static void main(String[] args) {

new Second().show(12);

}

}

Compile time error, as same method with different return type cannot be redefined in derived class

class Employ {

}

class Customer {

}

public class Quiz30 {

public void check(Object ob) {

if (ob instanceof Employ) {

System.out.println("Its Employ Class Object");

} else if (ob instanceof Customer) {

System.out.println("Its Customer Class Object");

} else {

System.out.println("its invalid Class object...");

}

}

public static void main(String[] args) {

new Quiz30().check(new Employ());

}

}

Its an Employ class object.

public class Quiz99 {

public static void main(String[] args) {

System.out.println("HI");

}

public static void main() {

System.out.println("Bye");

}

}

we must call static method without creating an object to class.

public class P1 {

public void show(int x) {

switch(x) {

case 1 :

System.out.println("Hi 1");

case 2 :

System.out.println("Bye 2");

case 3 :

System.out.println("Hello 3");

case 4 :

System.out.println("Test 4");

case 5 :

System.out.println("Demo 5");

default :

System.out.println("Default Message");

}

}

public static void main(String[] args) {

P1 obj = new P1();

obj.show(3);

}

}

Hello 3 Test 4 Demo 5 DefaultMessage

public class P2 {

public void show(Object ob) {

if (ob == "ABC") {

System.out.println("Correct A AND D");

} else {

System.out.println("Error B AND C");

}

}

public static void main(String[] args) {

String s="ABC";

P2 obj=new P2();

obj.show(s);

}

}

Correct A and D

Sting is a reference variable and object also reference.

When comparing two refences it show true.

public class P3 {

public static void main(String[] args) {

byte b=4;

int x=(Integer)b;

System.out.println(x);

int y =12;

byte p = (Byte)y;

System.out.println(p);

}

}

Compile time Error ;

public class P6 {

public void check(int x) {

if (x=2) {

System.out.println("Hi");

} else {

System.out.println("Bye");

}

}

public static void main(String[] args) {

new P6().check(2);

}

}

Compile Time error

public class P7 {

static int count=0;

public P7() {

count++;

System.out.println(count);

}

public static void main(String[] args) {

new P7();

new P7();

new P7();

}

}

1 2 3

enum Test {

A, B, X, Y, P, C,N;

private Test() {

System.out.println("Hi");

}

}

public class P13 {

public static void main(String[] args) {

Test t =Test.Y;

}

}

It prints message Hi for 7 types.

public class P14 {

public static void main(String[] args) {

String arr[] =new String[]{"Divya",

"Fantasy"};

String a="Divya";

System.out.println(a.equals(arr[0]));

}

}  
true

public class P15 {

public static void main(String[] args) {

try {

int x=5/0;

} catch(ArithmeticException e) {

System.out.println("Arithmetic Exception");

}

catch(Exception e) {

System.out.println("Global Exception");

}

}

}

Arithmetic Exceprtion

public class P16 {

public static void main(String[] args) {

try {

System.out.println("B");

throw new Exception();

} catch(ArithmeticException e) {

System.out.println("A");

} catch(Exception e) {

System.out.println("C");

} finally {

System.out.println("Z");

}

}

}

B C Z

public class P17 {

public static void main(String[] args) {

try {

System.out.println("B");

throw new NumberFormatException();

} catch(ArithmeticException e) {

System.out.println("A");

} catch(Exception e) {

System.out.println("C");

} finally {

System.out.println("Z");

}

}

}

B C Z

public class P23 {

public static void main(String[] args) {

int[] a=new int[5];

try {

a[6]=5/0;

} catch(ArrayIndexOutOfBoundsException e) {

System.out.println("Array Size Small...");

} catch(ArithmeticException e) {

System.out.println("Division By Zero Impossible");

} catch(Exception e) {

System.out.println("General Exception");

}

}

}

Division by zero impossible

public class P26 {

public static void main(String[] args) {

int[] a=new int[]{12,5,3,23,45};

for(int i : a) {

System.out.println(a);

}

}

}

Address print 5 times

class C1 {

void test() {

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

}

}

class C2 extends C1 {

void test() {

System.out.println("Bye");

}

}

public class P28 {

public static void main(String[] args) {

// C1 obj=(C1)new C2();

// C1 obj = (C2)new C2();

// C2 obj = (C1)new C2(); error

C2 obj = (C2)new C2();

obj.test();

}

}

Bye

public class P31 {

public static void main(String[] args) {

try {

int a=5;

int b=4;

int c=a/b;

System.out.println("Hi");

} catch(ArithmeticException e) {

System.out.println("Divex");

} finally {

System.out.println("Program End...");

}

System.out.println("Program from FTP114");

}

} HI Program End Program from FTP114

public class P33 {

public static void main(String[] args) {

String s = "Hello";

s.concat("World");

System.out.println(s);

String p = "I";

p=p + " Like ";

p=p + " Java";

System.out.println(p);

}

}

Hello

ILikeJava

public class P34 extends String {

public static void main(String[] args) {

System.out.println("Strings are Immutable...");

}

}

Compile Time error as String class is Final

public class P35 {

static int x=12;

public static int show() {

return x--;

}

public static void main(String[] args) {

show();

System.out.println(x);

}

}

11

public class P36 {

public static void main(String[] args) {

if (null==null) {

System.out.println("Hi");

} else {

System.out.println("Bye");

}

}

}

Hi

public class P39 {

static boolean climate;

public static void main(String[] args) {

System.out.println(climate);

}

}

false

class Demo {

static {

System.out.println("Hi");

}

}

public class P40 {

public static void main(String[] args) {

System.out.println("Bye");

}

static {

System.out.println("Hello");

}

}

Hello Bye

import java.util.HashSet;

public class P44 {

public static void main(String[] args) {

HashSet hs = new HashSet();

hs.add(new Integer(12));

hs.add(new Integer(12));

hs.add(new Integer(12));

hs.add(new Integer(12));

for(Object ob : hs) {

System.out.println(ob);

}

}

} 12

import java.util.Date;

import java.text.SimpleDateFormat;

import java.text.ParseException;

public class P45 {

public static void main(String[] args) throws ParseException {

Date obj = new Date();

System.out.println(obj);

String str ="2019/02/23";

SimpleDateFormat sdf = new SimpleDateFormat("yyyy/MM/dd");

Date d = sdf.parse(str);

System.out.println(d.getMonth());

System.out.println(d.getDay());

System.out.println(d.getDate());

}

}

public class Quiz22 {

public static void main(String[] args) {

int[] a=new int[]{1,2,3,4,5};

try {

for(int i=0;i<=7;i++) {

System.out.println(a[i]);

}

} catch(ArrayIndexOutOfBoundsException e) {

System.out.println("0");

}

}

}

1 2 3 4 5 0

What is the top level class in java (Object)

What is the default package in java (java.lang)

Which method of object class will have string representation (toString())