What will be the output of following program?  
public class Test {  
  
public static void main(String[] args) {  
  
try{  
int o[] = new int[2];  
o[3]=23;  
}catch(Exception e){  
System.out.println(e.getMessage());  
e.printStackTrace();  
}  
  
  
}  
  
}  
  
==============================================================================  
  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
  
try{  
int o[] = new int[2];  
o[3]=23;  
o[1]=33;  
}catch(Exception e){  
System.out.println(e.getMessage());  
e.printStackTrace();  
}  
  
System.out.println(o[1]);  
}  
  
}  
  
===============================================================================  
What will be the output of following program?  
  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
int o[] = new int[2];  
  
try{  
o[3]=23;  
o[1]=33;  
}catch(Exception e){  
System.out.println(e.getMessage());  
e.printStackTrace();  
}  
  
System.out.println("2nd pos --"+o[1]);  
}  
  
}  
================================================================================  
What will be the output of following program?  
  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
SomeClass obj=null;  
try{  
obj.someMethod();  
System.out.println("success");  
}catch(Exception e){  
System.out.println(e.getMessage());  
e.printStackTrace();  
}  
  
}  
}  
==================================================================================  
What will be the output of following program?  
  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
divide(4,2);  
divide(4,0);  
  
}  
public static int divide(int a,int b) throws Exception{  
int result = a/b;  
return result;  
}  
}  
  
===================================================================================  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
int a=divide(4,2);  
System.out.println(a);  
int b=divide(4,0);  
  
System.out.println(b);  
  
}  
public static int divide(int a,int b) throws Exception{  
int result = a/b;  
return result;  
}  
}  
  
=====================================================================================  
  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
int a=divide(4,2);  
System.out.println(a);  
int b=divide(4,0);  
  
System.out.println(b);  
  
}  
public static int divide(int a,int b) {  
int result = a/b;  
return result;  
}  
}  
========================================================================================  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
int a=divide(4,2);  
System.out.println(a);  
int b=divide(4,0);  
  
System.out.println(b);  
  
}  
public static int divide(int a,int b) {  
int result=0;  
try{  
result = a/b;  
}catch(Exception e){  
e.printStackTrace();  
}  
return result;  
}  
}  
=========================================================================================  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
try{  
int a=divide(4,2);  
System.out.println(a);  
int b=divide(4,0);  
System.out.println(b);  
}catch(Exception e){  
System.out.println("error");  
}  
  
}  
public static int divide(int a,int b) {  
int result=a/b;  
  
return result;  
}  
}  
=========================================================================================  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
try{  
int a=divide(4,2);  
System.out.println(a);  
int b=divide(4,0);  
System.out.println(b);  
}catch(Exception e){  
System.out.println("error 1");  
}  
  
}  
public static int divide(int a,int b) {  
int result=0;  
try{  
result=a/b;  
}catch(Exception e){  
System.out.println("error 2");  
}  
  
return result;  
}  
}  
==========================================================================================  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
  
System.out.println("A");  
Thread.sleep(5000L);  
System.out.println("B");  
  
  
}  
  
}  
==========================================================================  
What will be the output of following program?  
  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
  
throw new Exception("Some exception");  
  
  
}  
  
}  
=========================================================================  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
xyz();  
  
  
}  
  
public static void xyz() throws Exception{  
throw new Exception("Some exception");  
  
}  
  
}  
  
===========================================================================  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\*/  
public static void main(String[] args) {  
try {  
xyz();  
} catch (Exception e) {  
// TODO Auto-generated catch block  
System.out.println("error 1");  
e.printStackTrace();  
}  
  
  
}  
  
public static void xyz() throws Exception{  
throw new Exception("Some exception");  
  
}  
  
}  
==========================================================================  
What will be the output of following program?  
  
public class Test {  
  
/\*\*  
\* @param args  
\* @throws Exception  
\*/  
public static void main(String[] args) throws Exception {  
  
xyz();  
  
  
}  
  
public static void xyz() throws Exception{  
throw new Exception("Some exception");  
  
}  
  
}