1. Write an application that accepts two numbers, divide the first number with the second number and display the result. Hint: You need to handle ArithmeticException which is thrown when there is an attempt to divide a number by a zero.

**import** java.util.Scanner;

**public** **class** ArithExcep {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

**try** {

**int** a,b;

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter the first number ");

a = sc.nextInt();

System.***out***.println("Enter the second number ");

b = sc.nextInt();

System.***out***.println(a/b);

}

**catch**(ArithmeticException e)

{

System.***out***.println("Exception Handled "+ e);

}

}

}

**Output:**

Enter the first number

12

Enter the second number

0

Exception Handled java.lang.ArithmeticException: / by zero

1. Carrying forward with the above problem, handled **ArithmeticException** by raising **UnsupportedOperationException** as a solution.

**package** assignment4;

**import** java.util.Scanner;

**public** **class** Unsupported

{

**public** **static** **void** main(String[] args)

{

Scanner sc=**new** Scanner(System.***in***);

**try**

{

System.***out***.println("Enter the first number:");

**int** a=sc.nextInt();

System.***out***.println("Enter the second number");

**int** b=sc.nextInt();

**int** c=a/b;

System.***out***.println("answer is: "+c);

}

**catch**(ArithmeticException e)

{

e.printStackTrace();

System.***out***.println("Divide by zero is an runtime error");

}

**throw** **new** UnsupportedOperationException("Invalid");

}

}

**Output:**

Enter the first number:

10

Enter the second number

0

java.lang.ArithmeticException: / by zero

Divide by zero is an runtime error

at assignment4.Unsupported.main(Unsupported.java:16)

Exception in thread "main" java.lang.UnsupportedOperationException: Invalid

at assignment4.Unsupported.main(Unsupported.java:24)

1. Perform withdraw functionality with saving account object.

i)Raise InsufficientBalanceException if you are trying to withdraw more than balance.

**package** Account;

**import** java.util.Scanner;

**class** InsufficientBalanceException **extends** RuntimeException{

}

**public** **class** SavingAccount {

Scanner sc = **new** Scanner(System.***in***);

**public** **void** withdrawal(**double** a)

{

System.***out***.println("Enter your Id ");

**long** id = sc.nextLong();

System.***out***.println("Enter your balance ");

**double** b = sc.nextDouble();

**try** {

**if**(a<=b) {

b = b - a;

System.***out***.println("Balance= " + b);

}

**else** {

**throw** **new** InsufficientBalanceException();

}

}

**catch** (InsufficientBalanceException e) {

e.printStackTrace();

}

}

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

SavingAccount s = **new** SavingAccount();

s.withdrawal(2000);

}

}

**Output:** Enter your Id

1277

Enter your balance

700

Account.InsufficientBalanceException

at Account.SavingAccount.withdrawal(SavingAccount.java:21)

at Account.SavingAccount.main(SavingAccount.java:32)

ii) Raise a illigalBankTransaction if you are trying to withdraw negative amount from account.

**package** Account;

**import** java.util.Scanner;

**class** IllegalBankTransactionException **extends** RuntimeException{

}

**public** **class** SavingsAccount {

Scanner sc = **new** Scanner(System.***in***);

**public** **void** withdrawal(**double** a)

{

System.***out***.println("Enter your Id ");

**long** id = sc.nextLong();

System.***out***.println("Enter your balance ");

**double** b = sc.nextDouble();

**try** {

**if**(a>0) {

System.***out***.println("Balance= " + b);

}

**else** {

**throw** **new** IllegalBankTransactionException();

}

}

**catch** (IllegalBankTransactionException e) {

e.printStackTrace();

}

}

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

SavingsAccount s = **new** SavingsAccount();

s.withdrawal(-10201);

}}

**Output:** Enter your Id

124563

Enter your balance

100

Account.IllegalBankTransactionException

at Account.SavingsAccount.withdrawal(SavingsAccount.java:21)

at Account.SavingsAccount.main(SavingsAccount.java:32)