```java
public class TestThrow{
    public static void checkNum(int num){
        if (num<1){
            throw new ArithmeticException("\nNumber is negative,cannot calculate square");

        }
        else {
            System.out.println("Squre of"+num+"is"+(num*num));
        }
    }
    public static void main(String[]args){
        TestThrow obj = new TestThrow();
        obj.checkNum(-3);
        System.out.println("Rest of the code ");
    }
}
```

input

```
Exception in thread "main" java.lang.ArithmeticException:
Number is negative,cannot calculate square
        at TestThrow.checkNum(TestThrow.java:4)
        at TestThrow.main(TestThrow.java:13)


...Program finished with exit code 1
Press ENTER to exit console.
```

```java
public class TestThrow {
    public static void checkNum(String s){
        if(s==null){
        throw new ArithmeticException("\n Number is negative, cannot calculaye square");

    }
    else {
        //System.out.println("square of"+s+"is"+(num*num));
    }
}
public static void main(String[] args){
    TestThrow obj = new TestThrow();
    obj.checkNum(null);
    System.out.println("rest of the code");
}
}
```

input

```
Exception in thread "main" java.lang.ArithmeticException:
 Number is negative, cannot calculaye square
        at TestThrow.checkNum(TestThrow.java:4)
        at TestThrow.main(TestThrow.java:13)


...Program finished with exit code 1
Press ENTER to exit console.
```

```java
public class Main{
    static void checkAge(int age) throws ArithmeticException{
        if(age<18){
            throw new ArithmeticException("Access denied - You must be at least 18 years old");
        }
        else{
            System.out.println("Access granted - You are old enough");
        }
    }
    public static void main(String[] args){
        checkAge(20);
        //set age to 15

    }
}
```

input

```
Access granted - You are old enough


...Program finished with exit code 0
Press ENTER to exit console.
```

TestThrowAndThro...

```java
public class TestThrowAndThrows{
    static void method() throws ArithmeticException{
        System.out.println("inside the method()");
        throw new ArithmeticException("throwing ArithmeticException");
    }
    public static void main(String[] args){
        try{
            method();
        }
        catch(ArithmeticException e){
            System.out.println("caught in main() method");
        }
    }
}
```

input

```
inside the method()
caught in main() method


...Program finished with exit code 0
Press ENTER to exit console.
```

TestThrows.java

```java
public class TestThrows{
    public static int divideNum(int m,int n)throws ArithmeticException{
        int div = m/n;
        return div;
    }
    public static void main(String[] args){
        TestThrows obj = new TestThrows();
        try{
            System.out.println(obj.divideNum(45,0));
        }
        catch(ArithmeticException e){
            System.out.println("\nNumber cannot be divided by 0");
        }
        System.out.println("Rest of the code..");
    }
}
```

input

```
Number cannot be divided by 0
Rest of the code..


...Program finished with exit code 0
Press ENTER to exit console.
```