

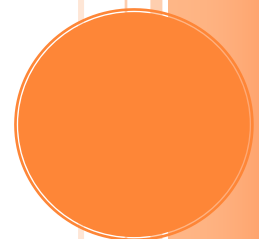
RAPPORT DE TP 1:

EXCEPTIONS

Encadré par: M. Fahd KARAMI

Réalisé par : Kenza BOUALAME

2019/2020



RAPPORT DE TP 1:

INTRODUCTION :

Dans le cadre des études en licence ingénierie des systèmes informatiques et logiciels, la séance de TP a une très bonne utilité afin d'appliquer les connaissances et comprendre plus le cours.

Ce TP permet d'appliquer les exemples de cours des exceptions en Java avancée et également les exercices de TP

EXEMPLES DE COURS :

Exemple 1 : [ArrayIndexOutOfBoundsException](#)

```

TestException.java
1 package tp;
2
3 public class TestException {
4 public static void main (String [] args) {
5 int[] t=new int [5]; // indices de 0 à 4
6 System.out.println("début");
7 int x= t[5]; // arrêt du programme
8 System.out.println("affectation réussie!");
9 } }

```

- *Exécution :*

```

Console
<terminated> TestException [Java Application] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe (2 févr. 2020 à 14:54:13)
début
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 5
    at tp.TestException.main(TestException.java:7)

```

Exemple 2 : [ArithmeticException](#)

```

TestException.java
1 package tp;
2
3 public class TestException {
4 public static void main (String [] args) {
5 int[] t={6,2,7,9,7};
6 System.out.println("début");
7 int x= 1/(t[4]-t[2]) ; // arrêt du programme
8 System.out.println("affectation réussie!");
9 } }

```

- *Exécution :*

```

Console
<terminated> TestException [Java Application] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe (2 févr. 2020 à 15:08:33)
Exception in thread "main" java.lang.ArithmeticException: / by zero
    at tp.TestException.main(TestException.java:7)
début

```

Exemple 3 : Catch

```

TestException.java
1 package tp;
2
3 public class TestException {
4     public static void main (String [] args) {
5         int[] t={6,2,7,9,7};
6         try {
7             System.out.println("début");
8             int x= 1/(t[4]-t[2]) ;
9             System.out.println("affectation reussie!");
10        }
11        catch (ArithmeticException e) {
12            System.out.println ("Operation impossible");
13        }
14        System.out.println("Fin") ;
15    } }

```

- *Exécution :*

```

Console
<terminated> TestException [Java Application] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe (4 févr. 2020 à 00:03:35)
début
Operation impossible
Fin

```

Exemple 4 : catch multiples

```
TestException.java
1 package tp;
2
3 public class TestException {
4     public static void main (String [] args) {
5         int[] t={6,2,7,9,7};
6         try {
7             System.out.println("début");
8             int x= 1/(t[4]-t[2]) ;
9             int y=t [5];
10            System.out.println("affectation reussie!");
11        } catch (ArithmeticException e) {
12            System.out.println ("Operation impossible");
13        }
14        catch (ArrayIndexOutOfBoundsException e ) {
15            System.out.println ("Operation impossible");
16        }
17        System.out.println("Fin") ;
18    } }
```

- *Exécution :*

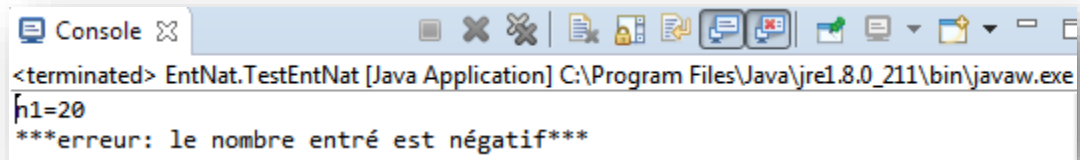
```
Console
<terminated> TestException [Java Application] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe (4 févr. 2020 à 00:03:35)
début
Operation impossible
Fin
```

TP 1 :

Exercice 1 :

```
EntNat.java ✖
1 package tp;
2
3 public class EntNat{
4     private int n;
5     public EntNat (int n) throws ErrConst
6     {
7         if (n<0) throw new ErrConst() ;
8         this.n = n;
9     }
10    public int getN() {return n;
11    }
12    class ErrConst extends Exception
13    {
14    }
15    public static class TestEntNat
16    {
17        public static void main (String[] args)
18        {
19            try
20            {
21                EntNat n1 = new EntNat (20);
22                System.out.println("n1=" + n1.getN( ));
23                EntNat n2 = new EntNat (-2);
24                System.out.println("n2=" + n2.getN( ));
25            }
26            catch (ErrConst e)
27            {
28                System.out.println("***erreur: le nombre entré est négatif***");
29                System.exit (-1) ;
30            }
31        }
32    }
33
34 }
```

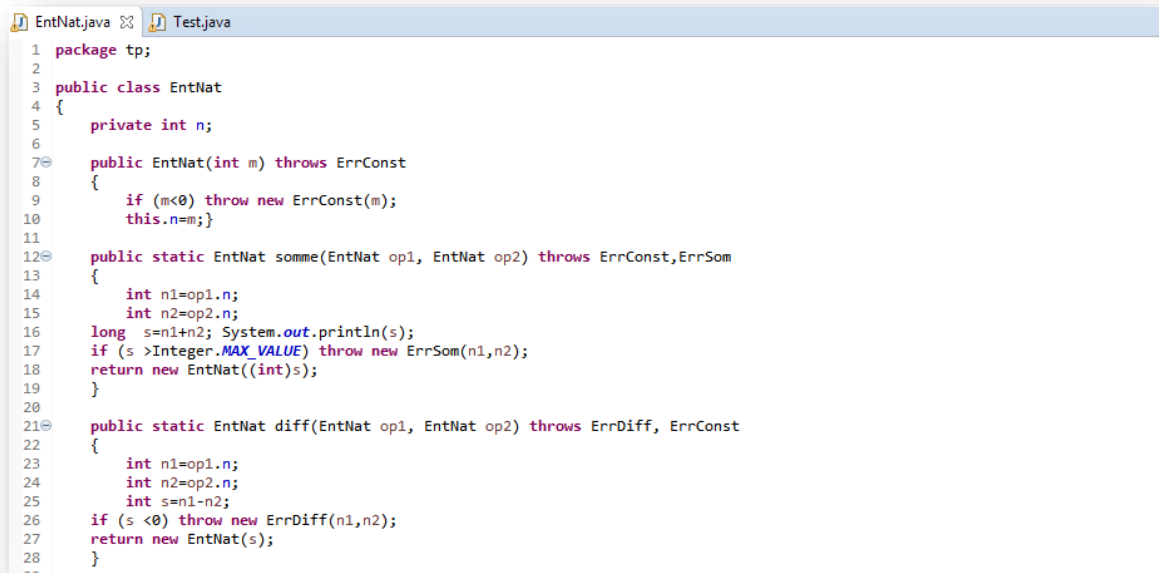
- *Exécution :*



```

Console
<terminated> EntNat.TestEntNat [Java Application] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe
h1=20
***erreur: le nombre entré est négatif***
  
```

Exercice 2 :



```

EntNat.java  Test.java
1  package tp;
2
3  public class EntNat
4  {
5      private int n;
6
7      public EntNat(int m) throws ErrConst
8      {
9          if (m<0) throw new ErrConst(m);
10         this.n=m;
11     }
12
13     public static EntNat somme(EntNat op1, EntNat op2) throws ErrConst,ErrSom
14     {
15         int n1=op1.n;
16         int n2=op2.n;
17         long s=n1+n2; System.out.println(s);
18         if (s >Integer.MAX_VALUE) throw new ErrSom(n1,n2);
19         return new EntNat((int)s);
20     }
21
22     public static EntNat diff(EntNat op1, EntNat op2) throws ErrDiff, ErrConst
23     {
24         int n1=op1.n;
25         int n2=op2.n;
26         int s=n1-n2;
27         if (s <0) throw new ErrDiff(n1,n2);
28         return new EntNat(s);
29     }
30 }
  
```

```

    public static EntNat produit(EntNat op1, EntNat op2) throws ErrProd, ErrConst
    {
        int n1=op1.n;
        int n2=op2.n;
        long s=(long)n1*(long)n2;
        if (s > Integer.MAX_VALUE) throw new ErrProd(n1,n2);
        return new EntNat((int)s);
    }

    public int getN(){
        return n;
    }
}

class ErrNat extends Exception {}

class ErrConst extends ErrNat
{
    protected int errval;
    public ErrConst(int valeur)
    {errval=valeur;}
}

```

```

class ErrOp extends ErrNat
{
    protected int n1,n2;
    public ErrOp(int a, int b)
    {
        this.n1=a;
        this.n2=b;
    }
}

class ErrSom extends ErrOp
{
    public ErrSom(int a, int b)
    {
        super(a,b);
    }
}

class ErrDiff extends ErrOp
{
    public ErrDiff(int a, int b)
    {
        super(a,b);
    }
}

class ErrProd extends ErrOp
{
    public ErrProd(int a, int b)
    {
        super(a,b);
    }
}

```



```

public class Test {

    public static void main (String[] args)
    {
        System.out.println(Integer.MAX_VALUE);
        try
        {
            EntNat n1= new EntNat(20);
            EntNat n2=new EntNat(12);
            EntNat d;
            d=EntNat.diff(n1,n2);
            d=EntNat.diff(n2,n1);
        }
        catch (ErrNat e)
        {
            System.out.println("Erreur Entier Naturel");
        }
        try
        {
            EntNat n3=new EntNat(450000);EntNat n4=new EntNat(563000);
            EntNat d= EntNat.diff(n3,n4);
        }

        catch (ErrConst e)
        {
            System.out.println("Erreur de construction avec valeur"+e.errval);
        }
    }
}

```

```

    catch (ErrDiff e)
    {
        System.out.println("Erreur de diff avec valeurs"+e.n1+" "+e.n2);}

    try {
        EntNat n3=new EntNat(999999999);EntNat n4=new EntNat(999999999);
        EntNat s= EntNat.somme(n3,n4);
    }

    catch (ErrConst e)
    {
        System.out.println("Erreur de construction avec valeur"+e.errval);}

    catch (ErrSom e)
    {System.out.println("Erreur de somme avec valeurs"+e.n1+" "+e.n2);}

    try {EntNat n3=new EntNat(450000000);EntNat n4=new EntNat(563000000);
        EntNat p=EntNat.produit(n3,n4);}

    catch (ErrConst e)
    {System.out.println("Erreur de construction avec valeur"+e.errval);}
    catch (ErrProd e)
    {System.out.println("Erreur de produit avec valeurs"+e.n1+" "+e.n2);}
    }
}

```

- *Exécution :*

Console

<terminated> Test [Java Application] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe (3 févr. 2020 à 22:22:26)

2147483647

Erreur Entier Naturel

Erreur de diff avec valeurs450000 563000

1999999998

Erreur de produit avec valeurs450000000 563000000

CONCLUSION

Ce TP était d'une très bonne utilité pour appliquer les connaissances acquises en cours des Exceptions ainsi que les exemples du cours