

Rapport Du TP1:

Les Exceptions en Java

Encadré Par :

Mr. Carami

Réalisé Par :

Jahid Khadija

Le : 09/02/2020

L'Ecole Supérieur De Technologie d'Essaouira

Exercice 1 :

- Classe EntNat :

```
1 package Exercice1;
2
3 import Exceptions.ErrConst;
4
5 public class EntNat {
6     private int number;
7     public EntNat(int num) throws ErrConst
8     {
9         if(num < 0) throw new ErrConst(num);
10        this.number=num;
11    }
12    public int getNumber() {
13        return number;
14    }
15    public void setNumber(int num) throws ErrConst {
16        if(num < 0) throw new ErrConst(num);
17        this.number=num;
18    }
19 }
20
21
```

- Classe ErrConst:

```
1 package Exceptions;
2
3 public class ErrConst extends Exception {
4     /**
5      *
6      */
7     private static final long serialVersionUID = 1L;
8     private int n;
9
10
11    public ErrConst(int x)
12    {
13        this.n=x;
14    }
15
16    public int getN() {
17        return n;
18    }
19
20
21 }
22
```

- Classe Test :

```
1 package Exercice1;
2
3 import Exceptions.ErrConst;
4
5 public class Test {
6
7     public static void main(String[] args) {
8         try {
9             EntNat n=new EntNat(-25);
10
11         } catch (ErrConst e) {
12             System.out.println(" Attention !! Le nombre est négatif :"+e.getN());
13         }
14
15
16     }
17
18 }
19
```

Exercice 2 :

- Classe EntNat :

```
1 package Exercice2;
2 import Exceptions.ErrConst;
3 import Exceptions.ErrDiff;
4 import Exceptions.ErrProd;
5 import Exceptions.ErrSom;
6 public class EntNat {
7     private int number;
8     public EntNat(int num)throws ErrConst
9     { if(num <0)throw new ErrConst(num); this.number=num;}
10    public int getNumber() {return number;}
11    public void setNumber(int num)throws ErrConst {
12        if(num <0)throw new ErrConst(num);
13        this.number=num;
14    }
15    public static EntNat somme(EntNat n1,EntNat n2)throws ErrConst, ErrSom
16    {
17        int n3 =n1.getNumber()+n2.getNumber();
18        if(n3>Integer.MAX_VALUE)throw new ErrSom(n1,n2);
19        return new EntNat(n3);
20    }
21    public static EntNat produit(EntNat n1,EntNat n2)throws ErrConst, ErrProd
22    {
23        int n3 =n1.getNumber()*n2.getNumber();
24        if(n3>Integer.MAX_VALUE)throw new ErrProd(n1,n2);
25        return new EntNat(n3);
26    }
27    public static EntNat difference(EntNat n1,EntNat n2)throws ErrConst, ErrDiff
28    {
29        int n3 =n1.getNumber()+n2.getNumber();
30        if(n3>Integer.MAX_VALUE)throw new ErrDiff(n1,n2);
31        return new EntNat(n3);
32    }
33 }
```

- Classe Test :

```
1 package Exercice2;
2
3 import Exceptions.ErrConst;
4 import Exceptions.ErrDiff;
5 import Exceptions.ErrProd;
6 import Exceptions.ErrSom;
7 public class Test {
8     public static void main(String[] args) throws ErrConst,ErrSom,ErrProd,ErrDiff {
9         try
10         {
11             new EntNat(-2);
12             EntNat n1=new EntNat(52);
13             EntNat n2=new EntNat(12);
14             EntNat.difference(n2, n1);
15         }
16         catch(ErrConst e){System.out.println("Erreur");}
17         try
18         {
19             EntNat n1=new EntNat(10000);
20             EntNat n2=new EntNat(12);
21             EntNat.produit(n2,n1);
22             EntNat.somme(n2,n1);
23             EntNat.difference(n2,n1);
24         }
25         catch(ErrSom e){System.out.println("Erreur de somme");}
26         catch(ErrProd e){System.out.println("Erreur de produit");}
27         catch(ErrDiff e) {System.out.println("Erreur de difference");}
28     }
29 }
```

- Classe ErrDiff:

```
1 package Exceptions;
2
3 import Exercice2.EntNat;
4
5 public class ErrDiff extends Exception {
6
7     public EntNat n1,n2;
8     private static final long serialVersionUID = 1L;
9     public ErrDiff(EntNat n1,EntNat n2)
10    {
11        this.n1=n1;
12        this.n2=n2;
13    }
14
15 }
16
```

- Classe ErrSom:

```
1 package Exceptions;
2
3 import Exercice2.EntNat;
4
5 public class ErrSom extends Exception {
6     public EntNat n1,n2;
7     private static final long serialVersionUID = 1L;
8     public ErrSom(EntNat n1,EntNat n2)
9     {
10         this.n1=n1;
11         this.n2=n2;
12     }
13
14
15 }
16
```

- Classe ErrProd :

```
1 package Exceptions;
2
3 import Exercice2.EntNat;
4
5 public class ErrProd extends Exception {
6     public EntNat n1,n2;
7     private static final long serialVersionUID = 1L;
8     public ErrProd(EntNat n1,EntNat n2)
9     {
10         this.n1=n1;
11         this.n2=n2;
12     }
13
14 }
15
```