TP - JAVA AVANCE

source de code :

TP-EXCEPTION:

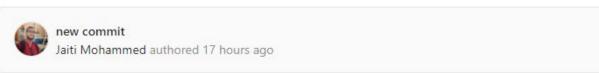
https://gitlab.com/JaitiMohammed/java_advanced/-/tree/master/TP%20Exception

TP-JDBC:

https://github.com/JaitiMohammed/Gestion_cabinet_medical

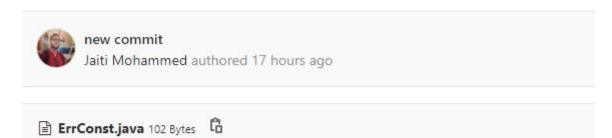
TP-EXCEPTION:

Exercice 1:

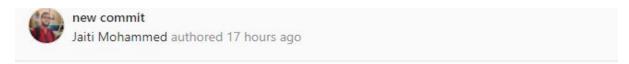


```
EntNaT.java 237 Bytes
 1
 public class EntNaT {
 4
             private int value;
             public EntNaT(int value) throws ErrConst{
  6
                    if(value < 0) {
                           throw new ErrConst("la valeur ne conviendra pas");
 8
 9
 10
                    this.value=value;
 12
            }
             public int getN() {
 14
 15
                    return value;
 16
             }
 17
 18 }
```

Jaiti Mohammed : mohammed.jaiti@edu.uca.ma

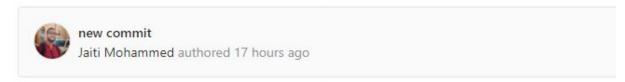


Jaiti Mohammed : mohammed.jaiti@edu.uca.ma



```
Test.java 397 Bytes
  1
      public class Test {
              public Test() {
  4
  5
                     // TODO Auto-generated constructor stub
  6
              }
  7
              public static void main(String[] args) {
  8
                     try {
                             EntNaT n1 = new EntNaT(20);
 10
                             System.out.println("n1 = " + n1.getN());
 11
 12
                             EntNaT n2 = new EntNaT(-30);
                             System.out.println("n2 = " + n2.getN());
 13
 14
                     }catch(ErrConst e) {
 16
 17
                             System.out.println("Erreur construction");
 18
                             System.exit(-1);
 19
                      }
 20
 21
 22
 23
             }
 24
 25
     }
```

Exercice 2:



```
EntNaT2.java 217 Bytes
 1
    public class EntNaT2 {
 3
 4
             private int value;
           public EntNaT2(int value) throws ErrConst2{
  6
  7
                    if(value < 0) {
  8
                            throw new ErrConst2(value);
 9
 10
                    this.value=value;
 11
 12
           }
 13
 14
           public int getN() {
                   return value;
 15
 16
             }
 17
 18 }
```

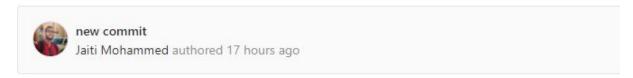


new commit

Jaiti Mohammed authored 17 hours ago

ErrConst2.java 173 Bytes

```
1
2
    public class ErrConst2 extends Exception {
3
            private int valeur;
            public ErrConst2(int value) {
4
                   this.valeur=value;
5
6
7
            }
8
            public int getValeur() {
9
10
                   return valeur;
11
            }
12 }
```



```
TestStart.java 353 Bytes 🔓
      public class TestStart {
  2
              public static void main(String[] args) {
  3
  5
                     try {
                             EntNaT2 n1 = new EntNaT2(20);
  7
                             EntNaT2 n2 = new EntNaT2(-20);
                             System.out.println("n1 = " + n1.getN());
 9
                             System.out.println("n2 = " + n2.getN());
                      } catch (ErrConst2 e) {
 10
                             System.out.println("Erreur de constructeur " + e.getValeur());
 11
 12
                             System.exit(-1);
 13
                      }
 14
 15
             }
 16 }
```

TP-JDBC

Description:

L'architecture de ce projet comprend : dao / application / entities se basant sur le design pattern MVC

dao contient:

- Interface qui englobe tout les méthodes nécessaires pour l'application (IMetier.java)
- Une classe qui met en pratique les méthodes déclarées dans l'interface IMetier (IMetierImpl.java)
- Une classe avec le design pattern Singleton qui permet de charger la connexion vers la base de donnée

application contient:

ce package comprend des classes de JavaFX:

- window : est la classe de start
- add : comprend des méthodes utilisés dans le stage avec différentes composants pratiques

entities:

ce package comprend les classes lié avec la base de donnée

- ry / creneaux
- clients / medecins / personne