	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

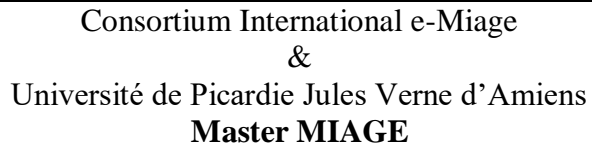
Exercice 1 : Couverture de Code

```
public class Calcul {

    /** Calcul la somme de deux nombres */
    public static int somme(int a,int b){
        return a+b;
    }

    public static int maFonction(int a, int b) {
        if (b >= 10) {
            return a/b;
        }
        return b;
    }
    /**
     * @return a / b si b != 0
     * @throw IllegalArgumentException si b == 0
     */
    public static int division(int a,int b){
        if ( b == 0 ) {
            throw new IllegalArgumentException("b ne doit pas etre 0");
        }
        return a/b;
    }
}
```

```
import static org.junit.jupiter.api.Assertions.assertEquals;
import org.junit.jupiter.api.Test;
public class CalculTest {
    @Test
    public void testConstructeur() {
        new Calcul();
    }
    @Test
    public void testSomme() {
        assertEquals(5,Calcul.somme(2,3));
    }
    @Test
    public void testDivision() {
        assertEquals(4,Calcul.division(8,2));
    }
}
```

Code du Module : **C306**

Intitulé du module : Ingénierie du logiciel (ISI_05)


Nom du tuteur : Nissrine Nehiri

```

[INFO] DOSSIER_MIAGE2021_DECEMBRE/C306/Activites/Activite02/Exercice01/src (master -> origin)
λ java -jar checkstyle-8.45.1-all-jar -c sun_checks.xml Calcul.java
D[-erreur de la v-@rification...
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:2:1: Les classes utilitaires ne doivent pas avoir de constructeur par d[-@fait ou public. [HideUtilityClassConstructor]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:3: Line has trailing spaces. [RegexpSingleline]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:3:1: Le fichier contient des caract [-@res de tabulation (ce n'est que la premi [-@tre occurrence). [FileTabCharacter]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:4: La premi [-@tre ligne de la Javadoc doit se terminer avec un point. [JavadocStyle]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:5: La balise @return doit [-@tre pr [-@sente et avoir une description. [JavadocMethod]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:5:36: Le param [-@tre a devrait [-@tre final. [FinalParameters]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:5:41: Balise Javadoc @param manque param 'a'. [JavadocMethod]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:5:42: Il manque un espace apr [-@s 's'. [WhitespaceAfter]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:5:43: Le param [-@tre b devrait [-@tre final. [FinalParameters]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:5:48: Balise Javadoc @param manque param 'b'. [JavadocMethod]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:5:50: Il manque un espace avant ' '. [WhitespaceAround]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:6:26: Il manque un espace apr [-@s 's'. [WhitespaceAround]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:6:26: Il manque un espace avant '+'. [WhitespaceAround]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:8:9: Commentaire Javadoc manquant. [MissingJavadocMethod]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:8:41: Le param [-@tre a devrait [-@tre final. [FinalParameters]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:8:50: Le param [-@tre b devrait [-@tre final. [FinalParameters]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:9:21: '0' devrait [-@tre d [-@fin comme une constante. [MagicNumber]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:10:18: Il manque un espace apr [-@s 's'. [WhitespaceAround]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:10:18: Il manque un espace avant ' '. [WhitespaceAround]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:18:39: Le param [-@tre a devrait [-@tre final. [FinalParameters]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:18:44: Balise Javadoc @param manque param 'a'. [JavadocMethod]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:18:45: Il manque un espace apr [-@s 's'. [WhitespaceAfter]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:18:51: Le param [-@tre b devrait [-@tre final. [FinalParameters]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:18:51: Balise Javadoc @param manque param 'b'. [JavadocMethod]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:18:53: Il manque un espace avant '!'. [WhitespaceAround]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:19:13: Il y a un espace de trop apr [-@s '('. [ParenPad]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:19:26: Il y a un espace de trop avant ')'. [ParenPad]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:22:18: Il manque un espace apr [-@s 's'. [WhitespaceAround]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:22:18: Il manque un espace avant ' '. [WhitespaceAround]
[ERROR] D:\DOSSIER_MIAGE2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\Calcul.java:23: Line has trailing spaces. [RegexpSingleline]
V[-@rification termin [-@re.
Checkstyle se termine par 30 erreurs.

```

```
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src (master -> origin)
λ java -jar checkstyle-8.45.1-all.jar -c sun_checks.xml CalculTest.java
D:\8bit de la v>@friction...
[ERROR] D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:1:47: L'utilisation des import. * est prohibé [® - org.junit.jupiter.api.Assertions. *. [AvoidStarImport] [ERROR]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:6:1: Commentaire Javadoc manquant. [MissingJavadocMethod]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:10:1: Commentaire Javadoc manquant. [MissingJavadocMethod]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:12:14: 'S' devrait être défini comme une constante. [MagicNumber]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:12:15: Il manque une espace après 'S'. [WhitespaceAfter]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:12:30: Il manque une espace après 'S'. [WhitespaceAfter]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:12:31: 'S' devrait être défini comme une constante. [MagicNumber]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:14:1: Commentaire Javadoc manquant. [MissingJavadocMethod]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:16:14: 'd' devrait être défini comme une constante. [MagicNumber]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:16:15: Il manque une espace après 'S'. [WhitespaceAfter]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:16:32: 'd' devrait être défini comme une constante. [MagicNumber]
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src\CalculTest.java:16:33: Il manque une espace après 'S'. [WhitespaceAfter]
V @friction terminé [®
Checkstyle se termine par 12 erreurs
```

	Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE
<u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32	Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri

1. Réécrivons le code pour qu'il passe les vérifications de checkstyle, spotbugs et PMD.

```

/**
 * Une classe permettant d'effectuer certaines
 * operations de base sur les entiers.
 * @author Sebastien Choplin
 */
public class Calcul {


    /**
     * Constructeur protected avec exception
     * pour empêcher l'instantiation de la classe.
     * @throws Exception On ne peut instancier cette classe.
     * @author Ndourdom Ezechiel
     */
    protected Calcul() throws Exception {
        throw new Exception("On ne peut instancier cette classe.");
    }

    /**
     * Calcul de la somme de deux entiers.
     * @author Sebastien Choplin
     * @param a est un entier.
     * @param b est un entier.
     * @return retourne la somme de a + b.
     */
    public static int somme(final int a, final int b) {
        return a + b;
    }

    /**
     * Fonction quotient a sur b.
     * @author Sebastien Choplin
     * @param a est un entier.
     * @param b est un entier.
     * @return retourne a/b si b est supérieur
     *         ou égale a 10 et b dans le cas
     *         contraire.
     */
    public static int maFonction(final int a, final int b) {
        final int bMin = 10;
        if (b >= bMin) {
            return a / b;
        }
        return b;
    }

    /**
     * Fonction division de deux entiers.

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

* @author Sebastien Choplin
* @param a est un entier.
* @param b est un entier.
* @return a / b si b != 0.
* @throws IllegalArgumentException si b == 0
**/
public static int division(final int a, final int b) {
    if (b == 0) {
        throw new IllegalArgumentException("b ne doit pas etre 0");
    }
    return a / b;
}
}

import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.junit.jupiter.api.Assertions.fail;

import org.junit.jupiter.api.Test;

/** Tests unitaire pour la classe Calcul. */
public class CalculTest {

    /** Constant entier VAL1. */
    private static final int VAL1 = 2;

    /** Constant entier VAL2. */
    private static final int VAL2 = 3;


    /** Constant entier VAL3. */
    private static final int VAL3 = 4;

    /** Constant entier VAL4. */
    private static final int VAL4 = 8;

    /** Constant entier somme VAL1 + VAL2. */
    private static final int SOMME = 5;

    /**
     * Test constructeur de la classe Calcul.
     */
    @Test
    public void testConstructeur() {
        try {
            new Calcul();
            fail("Reaction anormale car une exception est ignoree.");
        } catch (Exception e) {
            System.out.println("C'est normal d'etre ici car :\"" + e.getMessage() + "\"");
        }
    }
}

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

    }
}

/**
 * Test somme de deux entiers a et b.
 */
@Test
public void testSomme() {
    assertEquals(SOMME, Calcul.somme(VAL1, VAL2));
}

/**
 * Test division de deux entiers a/b avec b>0.
 */
@Test
public void TestDivision() {
    assertEquals(VAL3, Calcul.division(VAL4, VAL1));
}
}

```

```


D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src (master -> origin)
λ java -jar checkstyle-8.45.1-all.jar -c sun_checks.xml Calcul.java
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src (master -> origin)
λ java -jar checkstyle-8.45.1-all.jar -c sun_checks.xml CalculTest.java
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src (master -> origin)
λ |

```

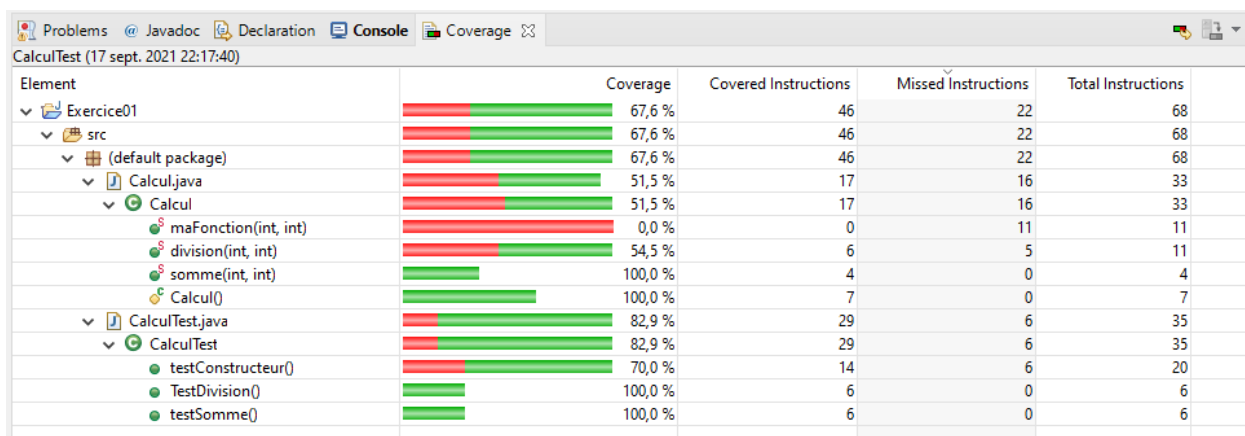
```

D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src (master -> origin)
λ java -jar checkstyle-8.45.1-all.jar -c sun_checks.xml Calcul.java
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src (master -> origin)
λ java -jar checkstyle-8.45.1-all.jar -c sun_checks.xml CalculTest.java
D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice01\src (master -> origin)
λ java -jar checkstyle-8.45.1-all.jar -c sun_checks.xml CalculTest.java

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

2. Donnons le rapport de couverture de code des tests unitaires de 'CalculTest' sur la classe 'Calcul'.




Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
Exercice01	67,6 %	46	22	68
src	67,6 %	46	22	68
(default package)	67,6 %	46	22	68
Calcul.java	51,5 %	17	16	33
Calcul	51,5 %	17	16	33
maFonction(int, int)	0,0 %	0	11	11
division(int, int)	54,5 %	6	5	11
somme(int, int)	100,0 %	4	0	4
Calcul()	100,0 %	7	0	7
CalculTest.java	82,9 %	29	6	35
CalculTest	82,9 %	29	6	35
testConstructeur()	70,0 %	14	6	20
TestDivision()	100,0 %	6	0	6
testSomme()	100,0 %	6	0	6

3. Expliquons pourquoi la couverture n'est pas complète (en couverture de ligne et en couverture de branche).

Couverture de ligne : dans la classe « CalculTest » il n'existe aucune méthode de test pour la méthode « *maFonction* » de la classe « Calcul » pour cela, aucune des lignes de cette méthode n'est parcourue lors de l'exécution des tests unitaires. Par ailleurs, lors de l'exécution du test de la méthode « *division* » de la classe « Calcul », aucune « exception n'est levée car le paramètre « b » n'est pas égal à « 0 » donc la ligne « *throw new IllegalArgumentException("b ne doit pas etre 0");* » n'est pas atteinte pendant l'exécution des tests unitaires ;

Couverture de branche : Ici, la méthode « *maFonction* » non testée lors de l'exécution des tests unitaires fait en sorte qu'aucune des branches contenues dans cette méthode n'est parcourue. Par ailleurs, pour le cas de la méthode « *division* », le test unitaire écrit pour cette méthode ne teste qu'un cas (b différent de 0) ce qui fait qu'une branche de cette méthode (vu qu'elle en possède deux) n'est pas parcourue.

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiel Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

4. Proposons une amélioration des tests pour obtenir une couverture complète (donner le résultat de couverture obtenu).

```
import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.junit.jupiter.api.Assertions.fail;
import org.junit.jupiter.api.Test;

/** Tests unitaire pour la classe Calcul. */
public class CalculTest {

    /** Constant entier VAL1. */
    private static final int VAL1 = 2;

    /** Constant entier VAL2. */
    private static final int VAL2 = 3;

    /** Constant entier VAL3. */
    private static final int VAL3 = 4;

    /** Constant entier VAL4. */
    private static final int VAL4 = 8;

    /** Constant entier VAL5. */
    private static final int VAL5 = 16;


    /** Constant entier VAL6. */
    private static final int VAL6 = 32;

    /** Constant entier VAL7. */
    private static final int VAL7 = 160;

    /** Constant entier VAL8. */
    private static final int VAL8 = 10;

    /** Constant entier somme VAL1 + VAL2. */
    private static final int SOMME = 5;

    /**
     * Test constructeur de la classe Calcul.
     */
    @Test
    public void testConstructeur() {
        try {
            new Calcul();
            fail("Reaction anormale car une exception est ignoree.");
        } catch (Exception e) {
            System.out.println("C'est normal d'etre ici car :\"" + e.getMessage() + "\"");
        }
    }
}
```


	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

/**
 * Test somme de deux entiers a et b.
 */
@Test
public void testSomme() {
    assertEquals(SOMME, Calcul.somme(VAL1, VAL2));
}

/**
 * Test division de deux entiers a/b avec b>0.
 */
@Test
public void testDivision() {
    assertEquals(VAL3, Calcul.division(VAL4, VAL1));
}

/**
 * Test division par zero.
 */
@Test
public void testDivisionFail() {
    try {
        Calcul.division(VAL1, 0);
        fail("Reaction anormale car une exception est ignoree.");
    } catch (IllegalArgumentException e) {
        System.out.println("C'est normal d'etre ici car : \""
+ e.getMessage() + "\"");
    }
}

/** Test MaFonction avec b < 10. */
@Test
public void testMaFonctionMin() {
    assertEquals(VAL3, Calcul.maFonction(VAL1, VAL3));
}

/** Test MaFonction avec b >= 10. */
@Test
public void testMaFonctionMax() {
    assertEquals(VAL1, Calcul.maFonction(VAL6, VAL5));
}

/** Test MaFonction avec b = 10. */
@Test
public void testMaFonction() {
    assertEquals(VAL5, Calcul.maFonction(VAL7, VAL8));
}
}

```





Consortium International e-Miage
&
Université de Picardie Jules Verne d'Amiens
Master MIAGE

Nom : NDOURDOM
Prénom : Ezéchiél
Devoir n°2, Décembre 2021
Nombre de pages : 32

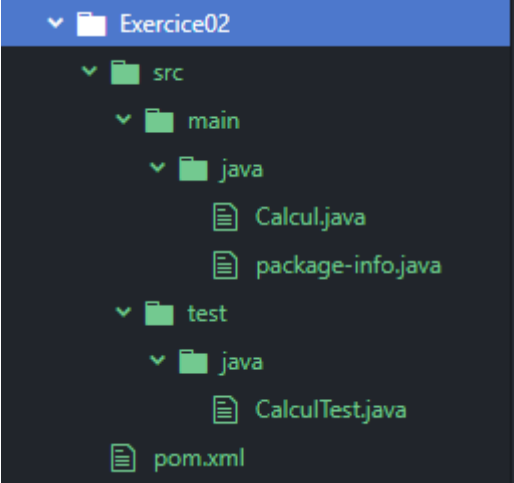
Code du Module : C306
Intitulé du module : Ingénierie du logiciel (ISI_05)
 Nom du tuteur : Nissrine Nehiri

Problems @ Javadoc Declaration Console Coverage							
CalculTest (17 sept. 2021 23:23:13)							
Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions			
Exercise01	<div><div></div></div> 87,0 %	94	14	108			
src	<div><div></div></div> 87,0 %	94	14	108			
(default package)	<div><div></div></div> 87,0 %	94	14	108			
CalculTest.java	<div><div></div></div> 81,3 %	61	14	75			
CalculTest	<div><div></div></div> 81,3 %	61	14	75			
testDivisionFail()	<div><div></div></div> 63,6 %	14	8	22			
testConstructeur()	<div><div></div></div> 70,0 %	14	6	20			
testDivision()	<div><div></div></div> 100,0 %	6	0	6			
testMaFonction()	<div><div></div></div> 100,0 %	6	0	6			
testMaFonctionMax()	<div><div></div></div> 100,0 %	6	0	6			
testMaFonctionMin()	<div><div></div></div> 100,0 %	6	0	6			
testSomme()	<div><div></div></div> 100,0 %	6	0	6			
Calcul.java	<div><div></div></div> 100,0 %	33	0	33			
Calcul	<div><div></div></div> 100,0 %	33	0	33			

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

Exercice 2 : Compilation assistée

1. Organisons l'arborescence du code et des tests telle que prévue par l'outil 'maven'



```

'maven'.
./pom.xml
./src
./src/main
./src/main/java
./src/main/java/Calcul.java
./src/test
./src/test/java
./src/test/java/CalculTest.java

```

2. Construisons un fichier "pom.xml" pour 'maven' permettant de :


- compiler le code
- compiler les tests (en utilisant JUnit5)
- exécuter les tests unitaires
- générer les rapports de checkstyle, spotbugs et PMD
- générer le rapport de couverture du code sur les tests unitaires (en utilisant JaCoCo)
- générer la documentation javadoc
- générer les différents rapports au format HTML
-

```

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <name>Activite02Exercice02</name>
  <description>Exercice02</description>
  <version>1.1</version>
  <groupId>default</groupId>
  <artifactId>activite02-exercice02</artifactId>

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

<dependencies><!-- dependance du projet -->
  <dependency><!-- bibliotheque junit -->
    <groupId>org.junit.jupiter</groupId>
    <artifactId>junit-jupiter-engine</artifactId>
    <version>5.4.2</version>
    <scope>test</scope>
  </dependency>
<dependency>
  <groupId>junit</groupId><!-- NOT org.junit here -->
  <artifactId>junit-dep</artifactId>
  <version>4.8.2</version>
  <scope>test</scope>
</dependency>

</dependencies>

<build>
  <plugins><!-- lister les plugins et leur version permet d'eviter que maven prenne celui qui trouvera
par default -->


    <plugin><!-- pour compiler -->
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.8.1</version>
    </plugin>

    <plugin>
      <artifactId>maven-project-info-reports-plugin</artifactId>
      <version>3.0.0</version>
    </plugin>

    <plugin><!-- pour l'execution des tests -->
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-surefire-plugin</artifactId>
      <version>3.0.0-M3</version>
      <configuration>
<testFailureIgnore>true</testFailureIgnore>
      </configuration>
    </plugin>

    <plugin><!-- pour la generation de rapports -->

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-site-plugin</artifactId>
<version>3.7.1</version>
<configuration>
  <locales>fr,en</locales><!-- en francais par default -->
</configuration>
</plugin>

```

```

<plugin><!-- pour la couverture de code avec JaCoco lors de l'exécution des tests JUnit -->
  <groupId>org.jacoco</groupId>
  <artifactId>jacoco-maven-plugin</artifactId>
  <version>0.8.4</version>
  <executions>
    <execution>
      <goals>
        <goal>prepare-agent</goal>
      </goals>
    </execution>
    <execution>
      <id>report</id>
      <phase>prepare-package</phase>
      <goals>
        <goal>report</goal>
      </goals>
    </execution>
  </executions>
</plugin>

```

```

<plugin><!-- pour l'analyse avec checkstyle -->
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-checkstyle-plugin</artifactId>
  <version>3.0.0</version>
</plugin>

```

```


</plugins>
</build>

```

```

<reporting><!-- on indique ici les rapports qu'on veut obtenir lors de la phase de generation de
rapports -->
<plugins>

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiel Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>


```
<plugin>
  <groupId>org.jacoco</groupId>
  <artifactId>jacoco-maven-plugin</artifactId>
  <reportSets>
    <reportSet>
      <reports>
        <report>report</report>
      </reports>
    </reportSet>
  </reportSets>
</plugin>
```

```
<plugin><!-- pour avoir le rapport checkstyle -->
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-checkstyle-plugin</artifactId>
  <version>3.0.0</version>
  <reportSets>
    <reportSet>
      <reports>
        <report>checkstyle</report>
      </reports>
    </reportSet>
  </reportSets>
</plugin>
```

```
<plugin><!-- pour avoir le lien au code source dans les rapports -->
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-jxr-plugin</artifactId>
  <version>2.3</version>
</plugin>
```

```
<plugin><!-- pour la verification du code avec spotbugs -->
  <groupId>com.github.spotbugs</groupId>
  <artifactId>spotbugs-maven-plugin</artifactId>
  <version>4.0.4</version>
</plugin>
```

```
<plugin><!-- pour la verification du code avec PMD -->
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-pmd-plugin</artifactId>
```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p>Nom : NDOURDOM Prénom : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

    <version>3.12.0</version>
</plugin>

<plugin><!-- pour avoir le rapport surfire (execution des tests) -->
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-surefire-report-plugin</artifactId>
  <version>3.0.0-M3</version>
</plugin>

</plugins>
</reporting>

<properties>
  <!-- encodage du code source -->
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
</properties>

</project>

```

3. Donnons la commande permettant d'exécuter toutes les lignes de la question 2 en une fois.

mvn clean compile test site:site

```

D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02 (master -> origin)
A mvn clean compile test site:site
[INFO] Scanning for projects...
[INFO]
[INFO] -----< default:activite02-exercice02 >-----
[INFO] Building Activite02Exercice02 1.1
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ activite02-exercice02 ---
[INFO]
[INFO] --- jacoco-maven-plugin:0.8.4:prepare-agent (default) @ activite02-exercice02 ---
[INFO] argline set to -javaagent:C:\Users\HP\m2\repository\org\jacoco\org.jacoco.agent\0.8.4\org.jacoco.agent-0.8.4-runtime.jar=destfile=D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02\target\jacoco.exec
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ activite02-exercice02 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02\src\main\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:compile (default-compile) @ activite02-exercice02 ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 2 source files to D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02\target\classes
[INFO]
[INFO] --- jacoco-maven-plugin:0.8.4:prepare-agent (default) @ activite02-exercice02 ---
[INFO] argline set to -javaagent:C:\Users\HP\m2\repository\org\jacoco\org.jacoco.agent\0.8.4\org.jacoco.agent-0.8.4-runtime.jar=destfile=D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02\target\jacoco.exec
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ activite02-exercice02 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02\src\main\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:compile (default-compile) @ activite02-exercice02 ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 2 source files to D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02\target\classes
[INFO]
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ activite02-exercice02 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02\src\test\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:testCompile (default-testCompile) @ activite02-exercice02 ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Activite02\Exercice02\target\test-classes
[INFO]
[INFO] --- maven-surefire-plugin:3.0.0-M3:test (default-test) @ activite02-exercice02 ---
[INFO]
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running CalculTest

```



Consortium International e-Miage
&
Université de Picardie Jules Verne d'Amiens
Master MIAGE

Nom : NDOURDOM
Prénom : Ezéchiél
Devoir n°2, Décembre 2021
Nombre de pages : 32

Code du Module : C306
Intitulé du module : Ingénierie du logiciel (ISI_05)
Nom du tuteur : Nissrine Nehiri

```
[INFO] Scanning for projects...
[INFO] -----< default:active02-exercice02 >-----
[INFO] Building Active02Exercice02 1.1
[INFO] -----[ jar ]-----
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ active02-exercice02 ---
[INFO] --- jacoco-maven-plugin:0.8.4:prepare-agent (default) @ active02-exercice02 ---
[INFO] argLine set to -javaagent:C:\Users\NHP\m2\repository\org\jacoco\org.jacoco.agent\0.8.4\org.jacoco.agent-0.8.4-runtime.jar=destfile=D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\
Active02\Exercice02\target\jacoco.exec
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ active02-exercice02 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Active02\Exercice02\src\main\resources
[INFO] --- maven-compiler-plugin:3.8.1:compile (default-compile) @ active02-exercice02 ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 2 source files to D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Active02\Exercice02\target\classes
[INFO] --- jacoco-maven-plugin:0.8.4:prepare-agent (default) @ active02-exercice02 ---
[INFO] argLine set to -javaagent:C:\Users\NHP\m2\repository\org\jacoco\org.jacoco.agent\0.8.4\org.jacoco.agent-0.8.4-runtime.jar=destfile=D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\
Active02\Exercice02\target\jacoco.exec
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ active02-exercice02 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Active02\Exercice02\src\main\resources
[INFO] --- maven-compiler-plugin:3.8.1:compile (default-compile) @ active02-exercice02 ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 2 source files to D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Active02\Exercice02\target\classes
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ active02-exercice02 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Active02\Exercice02\src\test\resources
[INFO] --- maven-compiler-plugin:3.8.1:testCompile (default-testCompile) @ active02-exercice02 ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to D:\DOSSIER_MIAGE\2021_DECEMBRE\C306\Activites\Active02\Exercice02\target\test-classes
[INFO] --- maven-surefire-plugin:3.0.0-M3:test (default-test) @ active02-exercice02 ---
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running CalculTest
C'est normal d'etre ici car : "On ne peut instancier cette classe."
C'est normal d'etre ici car : "b ne doit pas etre 0"
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.294 s - in CalculTest
```

4. Donnons les rapports HTML générés

Active02Exercice02

Documentation sur le projet

- Info Project
- Report Project
- Checkstyle
- Source Code
- Source Code des Tests
- Builds
- Surefire Report

Surefire Report

Summary

[Summary] [Package List] [Test Cases]

Tests	Errors	Failures	Skipped	Success Rate	Time
7	0	0	0	100%	0.294

Note: failures are anticipated and checked for with assertions while errors are unanticipated.

Package List

[Summary] [Package List] [Test Cases]

Package	Tests	Errors	Failures	Skipped	Success Rate	Time
CalculTest	7	0	0	0	100%	0.294

Note: package statistics are not computed recursively, they only sum up all of its test suites numbers.

Class	Tests	Errors	Failures	Skipped	Success Rate	Time
CalculTest	7	0	0	0	100%	0.294


Test Cases

[Summary] [Package List] [Test Cases]

CalculTest

Test Case	Time
testSomme	0.004
testConstructeur	0.001
testDivisionFail	0.001
testHaFonctionMax	0.001
testHaFonctionMin	0
testHaFonction	0
testDivision	0

Copyright © 2021. All rights reserved.

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p>Nom : NDOURDOM Prénom : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

Activite02Exercice02

Dernière publication: 2021-09-18 | Version: 1.1

Activite02Exercice02

Documentation sur le projet

- Info Projet
- Rapports Projet
 - JaCoCo
 - Checkstyle
 - Sources Xref
 - Sources Xref des Tests
 - SpotBugs
 - Surefire Report

Built by 

Rapports générés par Maven

Ce document fournit une vue d'ensemble des divers rapports générés automatiquement par Maven. Chaque rapport est décrit brièvement ci-dessous.

Vue d'ensemble

Document	Description
JaCoCo	JaCoCo Coverage Report.
Checkstyle	Rapport sur les conventions de style de codage
Sources Xref	Code source Java généré au format HTML.
Sources Xref des Tests	Code source Java pour les tests généré au format HTML.
SpotBugs	Génère un rapport du code source avec la librairie SpotBugs.
Surefire Report	Report on the test results of the project.

Copyright © 2021.All rights reserved.


Activite02Exercice02

Dernière publication: 2021-09-18 | Version: 1.1

Activite02Exercice02

Documentation sur le projet

- Info Projet
- Rapports Projet
 - JaCoCo
 - Checkstyle
 - Sources Xref
 - Sources Xref des Tests
 - SpotBugs
 - Surefire Report

Built by 

Résultats Checkstyle

Le document suivant contient les résultats de Checkstyle 6.18 avec les règles sun_checks.xml.

Résumé

Fichiers	Infos	Avertissements	Erreurs
2	0	0	0

Fichiers

Fichier	I	A	E
---------	---	---	---

Règles

Catégorie	Règle	Violations	Sévérité
-----------	-------	------------	----------

Détails

Copyright © 2021.All rights reserved.

Activite02Exercice02

Dernière publication: 2021-09-18 | Version: 1.1

Activite02Exercice02

Documentation sur le projet

- Info Projet
- Rapports Projet
 - JaCoCo
 - Checkstyle
 - Sources Xref
 - Sources Xref des Tests
 - SpotBugs
 - Surefire Report

Built by 

Rapport SpotBugs de détection des bugs

Le document suivant contient les résultats du SpotBugs.

La version de SpotBugs est 4.0.4

La limite est medium

L'effort est default

Résumé

Classes	Bugs	Erreurs	Classes Manquantes
1	0	0	0

Fichiers

Classe	Bugs
--------	------

Copyright © 2021.All rights reserved.



Nom : NDOURDOM
Prénom : Ezéchiél
Devoir n°2, Décembre 2021
Nombre de pages : 32

Code du Module : C306
Intitulé du module : Ingénierie du logiciel (ISI_05)
Nom du tuteur : Nissrine Nehiri

Activite02Exercice02 > default > Calcul

Sessions

Calcul

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed Cxty	Missed Lines	Missed Methods
maFonction(int int)	100 %	100 %	0	2	0	4	0
division(int int)	100 %	100 %	0	2	0	3	0
Calcul()	100 %	n/a	0	1	0	2	0
somme(int int)	100 %	n/a	0	1	0	1	0
Total	0 of 33	100 %	0 of 4	100 %	0	6	0


Created with JaCoCo 0.8.4.201905082037

5. Déposons les données utiles (pom.xml, src) dans le dossier “activite2” du dépôt (SVN ou GIT) utilisé précédemment.

Vous trouverez [ici](#) les données utiles (pom.xml, src) dans le dossier *Activite02*.

Exercice 3 : Sudoku

```
public interface Grille {
    /** Caractere de case vide */
    char EMPTY = '@';
    /**
     * Caractere possible a mettre dans la grille
     * pour une grille 4x4 : 1..4
     * pour une grille 9x9 : 1..9
     * pour une grille 16x16: 0..9-a..f
     * pour une grille 25x25: 0..9-a..o
     */
    char[] possible = new char[] {
        '1', '2', '3', '4', '5', '6', '7', '8', '9', '0',
        'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
        'k', 'l', 'm', 'n', 'o'
    };
    /** @return largeur/hauteur de la grille */
    int getDimension();
    /**
     * Affecte une valeur dans la grille
     * @param x position x dans la grille
     * @param y position y dans la grille
     * @param value valeur a mettre dans la case
     * @throws HorsBornesException si x ou y sont hors bornes (0-8)
     * @throws ValeurImpossibleException si la valeur est interdite aux vues des autres valeurs de la grille
     * @throws CaractereInterditException si value n'est pas un caractere autorise ('1',..., '9')
     */
    void setValue(int x, int y, char value) throws HorsBornesException, ValeurImpossibleException,
        CaractereInterditException;
    /**
     * Recupere une valeur de la grille
     * @param x position x dans la grille
     * @param y position y dans la grille
     * @return valeur dans la case x,y
     * @throws HorsBornesException si x ou y sont hors bornes (0-8)
     */
    char getValue(int x, int y) throws HorsBornesException;
    /**
     * Test si une grille est terminee
     * @return true si la grille est complete
     */
    boolean complete();
    /**
     * Test si une valeur est possible dans la grille par rapport a ce qu'elle
     * contient deja
     * @param x position x dans la grille
     * @param y position y dans la grille
     * @param value valeur a mettre dans la case
     * @throws HorsBornesException si x ou y sont hors bornes (0-8)
     * @throws CaractereInterditException si value n'est pas un caractere autorise ('1', ..., '9', ...)
     */
    boolean possible(int x, int y, char value) throws HorsBornesException, CaractereInterditException;
}
```

	Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE
<u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiel Devoir n°2, Décembre 2021 Nombre de pages : 32	Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri

1. Ecrivons les classes :

- *HorsBornesException* (exception levée lorsque les paramètres de positions sortent de la grille),

```
public class HorsBornesException extends Exception {

    public HorsBornesException(String message) {

        super(message); // message est le message a retourner a l'utilisateur

    }

}
```

- *ValeurImpossibleException*,

```
public class ValeurImpossibleException extends Exception {

    public ValeurImpossibleException(String message) {

        super(message); // message est le message a retourner à l'utilisateur

    }

}
```

- *CaractereInterditException*

```
public class CaractereInterditException extends Exception {

    public CaractereInterditException(String message) {

        super(message); // message est le message a retourner à l'utilisateur

    }

}
```

2. Ecrivons les tests

```
import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.junit.jupiter.api.Assertions.fail;

import org.junit.jupiter.api.Test;

/**
 * Classe GrilleImplTest representant de
 * tester les methodes de la classe GrilleImpl.
```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p>Nom : NDOURDOM Prénom : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

* @author Ndourdom Ezechiél
*
*/
public class GrilleImplTest {
    static final int TAILLE_4 = 4;
    static final int TAILLE_9 = 9;
    static final int TAILLE_16 = 16;
    static final int TAILLE_25 = 25;
    static final char EMPTY_CHAR = '@';
    static final char[] POSSIBLE = new char[] { '1', '2', '3', '4', '5', '6',
        '7', '8', '9', '0', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
        'k', 'l', 'm', 'n', 'o' };


    char [][] grilleDeQuatre = new char[][] {
        { '2', '3', '4', '1', },
        { '4', '1', '2', '3', },
        { '1', '4', '3', '2', },
        { '3', '2', '1', '4', }};

    char [][] grilleDeNeuf = new char[][] {
        { '9', '1', '3', '6', '2', '7', '5', '8', '4' },
        { '6', '4', '2', '5', '8', '9', '7', '3', '1' },
        { '8', '7', '5', '3', '4', '1', '6', '9', '2' },
        { '5', '8', '9', '1', '7', '4', '3', '2', '6' },
        { '2', '6', '1', '8', '9', '3', '4', '5', '7' },
        { '7', '3', '4', '2', '6', '5', '8', '1', '9' },
        { '1', '2', '8', '7', '3', '6', '9', '4', '5' },
        { '3', '9', '6', '4', '5', '2', '1', '7', '8' },
        { '4', '5', '7', '9', '1', '8', '2', '6', '3' }};

    char [][] grilleDeSeize = new char[][] {
        { '8', 'f', '6', '5', '1', 'a', '7', 'd', '4', '9', 'c', '3', '2', 'b', '0', 'e' },
        { '1', '4', '2', '3', 'b', 'c', 'e', '9', '0', '6', '8', 'd', '5', 'f', '7', 'a' },
        { 'c', '7', 'a', '9', '0', '4', '2', '3', 'b', 'e', 'f', '5', '6', 'd', '8', '1' },
        { 'b', '0', 'd', 'e', 'f', '5', '8', '6', '7', 'a', '2', '1', '4', '3', 'c', '9' },
        { '2', '6', '4', '7', 'd', '8', '9', 'b', 'c', '0', 'e', 'f', '3', 'a', '1', '5' },
        { '3', '9', '5', '1', '6', '0', 'a', 'f', '2', '8', 'b', '4', 'c', '7', 'e', 'd' },
        { 'f', 'd', '8', '0', 'c', '1', '3', 'e', '5', '7', 'a', '9', 'b', '6', '4', '2' },
        { 'a', 'b', 'e', 'c', '2', '7', '5', '4', 'd', '3', '1', '6', '9', '0', 'f', '8' },
        { 'd', '5', '9', '4', 'e', '3', 'f', '8', '1', 'c', '7', 'a', '0', '2', '6', 'b' },
        { '7', '2', 'b', 'a', '9', '6', 'd', 'c', 'e', '4', '0', '8', 'f', '1', '5', '3' },
        { '0', 'c', '3', 'f', '4', 'b', '1', 'a', '6', '5', 'd', '2', 'e', '8', '9', '7' },
        { '6', 'e', '1', '8', '5', '2', '0', '7', '9', 'f', '3', 'b', 'd', 'c', 'a', '4' },
        { 'e', '1', '7', '6', '3', 'f', '4', '2', 'a', 'd', '9', 'c', '8', '5', 'b', '0' },
        { '9', '8', '0', '2', 'a', 'd', '6', '5', 'f', 'b', '4', '7', '1', 'e', '3', 'c' },
        { '5', 'a', 'c', 'd', '8', '9', 'b', '0', '3', '1', '6', 'e', '7', '4', '2', 'f' },
        { '4', '3', 'f', 'b', '7', 'e', 'c', '1', '8', '2', '5', '0', 'a', '9', 'd', '6' }};

```

@Test

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiel Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

    public void testConstructeurQuatre() {
        GrilleImpl test = new GrilleImpl(TAILLE_4);
        assertEquals(TAILLE_4, test.getDimension());
        assertEquals(EMPTY_CHAR, test.getGrille()[0][0]);
    }

@Test
    public void testConstructeurNeuf() {
        GrilleImpl test = new GrilleImpl(TAILLE_9);
        assertEquals(TAILLE_9, test.getDimension());
        assertEquals(EMPTY_CHAR, test.getGrille()[0][0]);
    }

@Test
    public void testConstructeurSeize() {
        GrilleImpl test = new GrilleImpl(TAILLE_16);
        assertEquals(TAILLE_16, test.getDimension());
        assertEquals(EMPTY_CHAR, test.getGrille()[0][0]);
    }


@Test
    public void testConstructeurVingtEtCinq() {
        GrilleImpl test = new GrilleImpl(TAILLE_25);
        assertEquals(TAILLE_25, test.getDimension());
        assertEquals(EMPTY_CHAR, test.getGrille()[0][0]);
    }

@Test
    public void testGrilleQuatreSetValue() throws CaractereInterditException,
HorsBornesException, ValeurImpossibleException {
        GrilleImpl test = new GrilleImpl(TAILLE_4);
        test.setValue(0, 0, '1');
    }

@Test
    public void testGrilleNeufSetValue() throws CaractereInterditException,
HorsBornesException, ValeurImpossibleException {
        GrilleImpl test = new GrilleImpl(TAILLE_9);
        test.setValue(0, 0, '1');
    }

@Test
    public void testGrilleSeizeSetValue() throws CaractereInterditException,
HorsBornesException, ValeurImpossibleException {
        GrilleImpl test = new GrilleImpl(TAILLE_16);
        test.setValue(0, 0, '0');
    }

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

@Test
public void testGrilleVingtEtCinqSetValue() throws CaractereInterditException,
HorsBornesException, ValeurImpossibleException {
    GrilleImpl test = new GrilleImpl(TAILLE_25);
    test.setValue(0, 0, '1');
}


@Test
public void testGrilleSetValueXHorsBorne() throws CaractereInterditException,
HorsBornesException {
    GrilleImpl test = new GrilleImpl(TAILLE_4);
    try {
        test.setValue(TAILLE_4, 0, '1');
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (CaractereInterditException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\"");
    }
}

@Test
public void testGrilleSetValueYHorsBorne() throws HorsBornesException {
    GrilleImpl test = new GrilleImpl(TAILLE_9);
    try {
        test.setValue(0, TAILLE_9, '1');
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (CaractereInterditException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\"");
    }
}

@Test
public void testGrilleSetValueXHorsBorne2() throws HorsBornesException,
CaractereInterditException {
    GrilleImpl test = new GrilleImpl(TAILLE_16);
    try {
        test.setValue(-1, 0, '1');
    } catch (CaractereInterditException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\"");
    }
}

@Test
public void testGrilleSetValueYHorsBorne2() throws HorsBornesException,
ValeurImpossibleException, CaractereInterditException {

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```


GrilleImpl test = new GrilleImpl(TAILLE_16);
try {
    test.setValue(0, -1, '1');
    fail("Reaction anormale, une exception a ete ignoree.");
} catch (CaractereInterditException e) {
    System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
}
}

@Test
public void testGrilleNeufSetValueCaractereNonAutorise() throws
CaractereInterditException {
    GrilleImpl test = new GrilleImpl(TAILLE_9);
    try {
        test.setValue(0, 0, '0');
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (CaractereInterditException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
    }
}

@Test
public void testGrilleSeizeSetValueCaractereNonAutorise() throws
CaractereInterditException {
    GrilleImpl test = new GrilleImpl(TAILLE_16);
    try {
        test.setValue(0, 0, 'w');
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (CaractereInterditException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
    }
}

@Test
public void testGrilleSetValeurImpossible() throws ValeurImpossibleException,
CaractereInterditException {
    GrilleImpl test = new GrilleImpl(TAILLE_9);
    try {
        test.setValue(0, 0, 'y');
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (Exception e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
    }
}

```


	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```


@Test
public void testGrilleSetValueValeurDejaAjouteePresentSurLigne() throws
CaractereInterditException {
    GrilleImpl test = new GrilleImpl(TAILLE_9);
    try {
        test.setValue(0, 0, '1');
        test.setValue(0, 1, '1');
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (CaractereInterditException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
    }
}

@Test
public void testGrilleSetValueValeurDejaAjouteePresentSurColonne() throws
CaractereInterditException {
    GrilleImpl test = new GrilleImpl(TAILLE_16);
    try {
        test.setValue(0, 0, '1');
        test.setValue(1, 0, '1');
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (CaractereInterditException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
    }
}

@Test
public void testGrilleSetValueValeurDejaAjouteePresentSurBloc() throws
CaractereInterditException {
    GrilleImpl test = new GrilleImpl(TAILLE_16);
    try {
        test.setValue(0, 0, '1');
        test.setValue(1, 1, '1');
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (CaractereInterditException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
    }
}

@Test
public void testGrilleNeufGetValue() throws CaractereInterditException,
HorsBornesException {
    GrilleImpl test = new GrilleImpl(TAILLE_9);
    test.setValue(0, 0, '1');
    assertEquals('1', test.getValue(0, 0));
}

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

@Test
public void testGrilleSeizeGetValue() throws ValeurImpossibleException,
CaractereInterditException, HorsBornesException {
    GrilleImpl test = new GrilleImpl(TAILLE_16);
    test.setValue(0, 0, '0');
    assertEquals('0', test.getValue(0, 0));
}


@Test
public void testGrilleNeufGetValueHorsBornes() throws CaractereInterditException,
HorsBornesException {
    GrilleImpl test = new GrilleImpl(TAILLE_9);
    try {
        test.setValue(0, 0, '1');
        assertEquals('0', test.getValue(0, TAILLE_9));
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (HorsBornesException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
    }
}

@Test
public void testGrilleSeizeGetValueHorsBornes() throws CaractereInterditException,
HorsBornesException {
    GrilleImpl test = new GrilleImpl(TAILLE_16);
    try {
        test.setValue(0, 0, '1');
        assertEquals('0', test.getValue(0, TAILLE_16));
        fail("Reaction anormale, une exception a ete ignoree.");
    } catch (HorsBornesException e) {
        System.out.println("C'est normal de se retrouver ici car : \"" + e.getMessage() +
"\");
    }
}

@Test
public void testGrilleDeQuatreComplete() {
    GrilleImpl test = new GrilleImpl(TAILLE_4);
    test.setGrille(grilleDeQuatre);
    assertEquals(true, test.complete());
}

@Test
public void testGrilleNeufComplete() {
    GrilleImpl test = new GrilleImpl(TAILLE_9);
    test.setGrille(grilleDeNeuf);
    assertEquals(true, test.complete());
}

```

	Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE
<u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32	Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri

```

}

@Test
public void testGrilleSeizeComplete() {
    GrilleImpl test = new GrilleImpl(TAILLE_16);
    test.setGrille(grilleDeSeize);
    assertEquals(true, test.complete());
}

@Test
public void testGrilleDeQuatreIncomplete() {
    GrilleImpl test = new GrilleImpl(TAILLE_4);
    grilleDeNeuf[0][1] = EMPTY_CHAR;
    test.setGrille(grilleDeNeuf);
    assertEquals(false, test.complete());
}

@Test
public void testGrilleNeufIncomplete() {
    GrilleImpl test = new GrilleImpl(TAILLE_9);
    grilleDeNeuf[0][1] = EMPTY_CHAR;
    test.setGrille(grilleDeNeuf);
    assertEquals(false, test.complete());
}

@Test
public void testGrilleSeizeIncomplete() {
    GrilleImpl test = new GrilleImpl(TAILLE_16);
    grilleDeSeize[5][8] = EMPTY_CHAR;
    test.setGrille(grilleDeSeize);
    assertEquals(false, test.complete());
}
}

```

3. Ecrivons une implémentation dans une classe “GrilleImpl”.


```

/**
 * Implementation de la classe "GrilleImpl"
 * pour la creation d'une simple grille de sudoku
 * de dimension 4x4, 9x9, 16x16 ou 25x25.
 * @author NDOURDOM Ezechiel.
 */

public class GrilleImpl implements Grille {

    /**
     * Constante representant la dimension de la grille.

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiel Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

    */
    private final int dimension;

    /**
     * Grille de sudoku.
     */
    private char[][] grille;

    /**
     * Constructeur d'initialisation de
     * la grille de sudoku 4x4, 9x9, 16x16 ou 25x25.
     * @param taille de la grille de type int.
     */
    public GrilleImpl(final int taille) {
        this.dimension = taille;
        this.grille = new char[this.getDimension()][this.getDimension()];
        for (int i = 0; i < this.dimension; i++) {
            for (int j = 0; j < this.dimension; j++) {
                this.grille[i][j] = EMPTY;
            }
        }
    }


    /**
     * Accesseur de retour de la grille.
     * @return char [][]
     */
    public char[][] getGrille() {
        return grille;
    }

    /**
     * mutateur d'affectation d'une grille entiere.
     * @param laGrille la grille a affecter.
     */
    public void setGrille(final char[][] newGrille) {
        System.arraycopy(newGrille, 0, this.grille, 0, this.getDimension());
    }

    @Override
    public int getDimension() {
        return this.dimension;
    }

    @Override
    public void setValue(final int x, final int y, final char value)
        throws CaractereInterditException {
        try {
            this.possible(x, y, value);

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

//Si la value n'est pas autorisee dans
//la grille aux vues des autres valeurs
//contenues dans le tableau, on leve une
//exception.
if (!this.estAutorisee(x, y, value)) {
    throw new CaractereInterditException("la valeur est interdite au "
        + "vues des valeurs deja contenues dans la grille.");
}
this.getGrille()[x][y] = value;
} catch (Exception ex) {
    throw new CaractereInterditException(ex.toString());
}
}

```

```

@Override
public char getValue(final int x, final int y)
    throws HorsBornesException {
    if (estHorsBornes(x, y, this.getDimension())) {
        throw new HorsBornesException("x et/ou y est(sont) hors bornes.");
    }
    return this.getGrille()[x][y];
}

```

```

@Override
public boolean complete() {
    for (char[] sousGrille : this.getGrille()) {
        for (char carac : sousGrille) {
            if (carac == EMPTY) {
                return false;
            }
        }
    }
    return true;
}


```

```

@Override
public boolean possible(final int x, final int y, final char value)
    throws HorsBornesException {

    // On verifie que x et y sont dans les bornes autorisees;
    // pour cela, x et y ne doivent pas etre inferieur a 0
    // et ne doivent pas dépasser la taille max de la grille.
    if (estHorsBornes(x, y, this.getDimension())) {
        throw new HorsBornesException("x et/ou y est(sont) hors bornes.");
    }
}

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiel Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

}

//On cree un tableau qui recevra les valeurs
//possibles en fonction du type de la grille.
char[] temp = new char[this.getDimension()];

//Puis on recupere les valeurs possibles
System.arraycopy(possible, 0, temp, 0, this.getDimension());


//on verifie si value est un caractere autorise.
boolean estPresent = false;
for (char s : temp) {
    if (s == value) {
        estPresent = true;
        break;
    }
}
if (!estPresent) {
    throw new HorsBornesException("Cette valeur n'est pas autorisee "
        + "(parmis les valeurs possibles.)");
}
return estPresent;
}

/**
 * Methode pour verifier si x et y sont hors bornes.
 * @param x position x dans la grille.
 * @param y position y dans la grille.
 * @param laDimension taille de la grille.
 * @return true si x et/ou y sont hors bornes,
 *         false si x et y sont dans les bornes.
 */
private boolean estHorsBornes(final int x, final int y,
    final int laDimension) {
    boolean test1 = false;
    if (x < 0 || y < 0) {
        test1 = true;
    } else {
        test1 = false;
    }

    boolean test2 = false;
    if (x >= laDimension || y >= laDimension) {
        test2 = true;
    } else {
        test2 = false;
    }

    if (test1 || test2) {

```

	<p style="text-align: center;">Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE</p>
<p><u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32</p>	<p style="text-align: center;">Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri</p>

```

    return true;
}
return false;
}

```

```

/**
 * Methode pour verifier si une valeur est autorisee
 * aux vues des autres valeurs de la grille.
 * @param x position x dans la grille.
 * @param y position y dans la grille.
 * @param value valeur a tester.
 * @return true si la valeur est autorisee,
 *         false dans le cas contraire.
 */

```

```

private boolean estAutorisee(final int x, final int y, final char value) {
    // On verifie que l'element n'exite pas
    // sur la meme ligne.
    if (this.presentSurLigne(x, value)) {
        return false;
    }
}

```

```

    // On verifie que l'element n'exite pas
    // sur la meme colonne.
    if (this.presentSurColonne(y, value)) {
        return false;
    }
}

```

```

/*On doit vzrifier que la sous grille
(3x3) ou (4x4) ne contient pas deja
la valeur.*/

```

```

int racine = (int) Math.sqrt(this.getDimension());
if (presentSurBloc(x, y, value, racine)) {
    return false;
}
return true;
}

```

```


/**
 * Methode pour verifier si une valeur est presente
 * sur une ligne.
 * @param x numero de la ligne.
 * @param value valeur a tester.
 * @return true si la valeur est presente,
 *         false dans le cas contraire.
 */

```

```

private boolean presentSurLigne(final int x, final char value) {
    for (int i = 0; i < this.getDimension(); i++) {
        if (this.getGrille()[x][i] == value) {
            return true;
        }
    }
}

```


	Consortium International e-Miage & Université de Picardie Jules Verne d'Amiens Master MIAGE
<u>Nom</u> : NDOURDOM <u>Prénom</u> : Ezéchiél Devoir n°2, Décembre 2021 Nombre de pages : 32	Code du Module : C306 Intitulé du module : Ingénierie du logiciel (ISI_05) Nom du tuteur : Nissrine Nehiri

```

    }
}
return false;
}

/**
 * Methode pour verifier si une valeur est presente
 * sur une colonne.
 * @param y numero de la colonne.
 * @param value valeur a tester.
 * @return true si la valeur est presente,
 *         false dans le cas contraire.
 */
private boolean presentSurColonne(final int y, final char value) {
    for (int i = 0; i < this.getDimension(); i++) {
        if (this.getGrille()[i][y] == value) {
            return true;
        }
    }
    return false;
}

/**
 * Methode pour verifier si une valeur est presente
 * dans une sous grille ou un bloc de la grille.
 * @param x numero de la ligne.
 * @param y numero de la colle.
 * @param value valeur a chercher.
 * @param racine racine carree de la dimension.
 * @return true si la valeur est presente,
 *         false dans le cas contraire.
 */
private boolean presentSurBloc(final int x, final int y,
    final char value, final int racine) {
    int posX = x - (x % racine);
    int posY = y - (y % racine);
    // ou encore : posX = 3*(i/3), posY = 3*(j/3);

    for (int i = posX; i < posX + racine; i++) {
        for (int j = posY; j < posY + racine; j++) {
            if (this.getGrille()[i][j] == value) {
                return true;
            }
        }
    }
    return false;
}
}

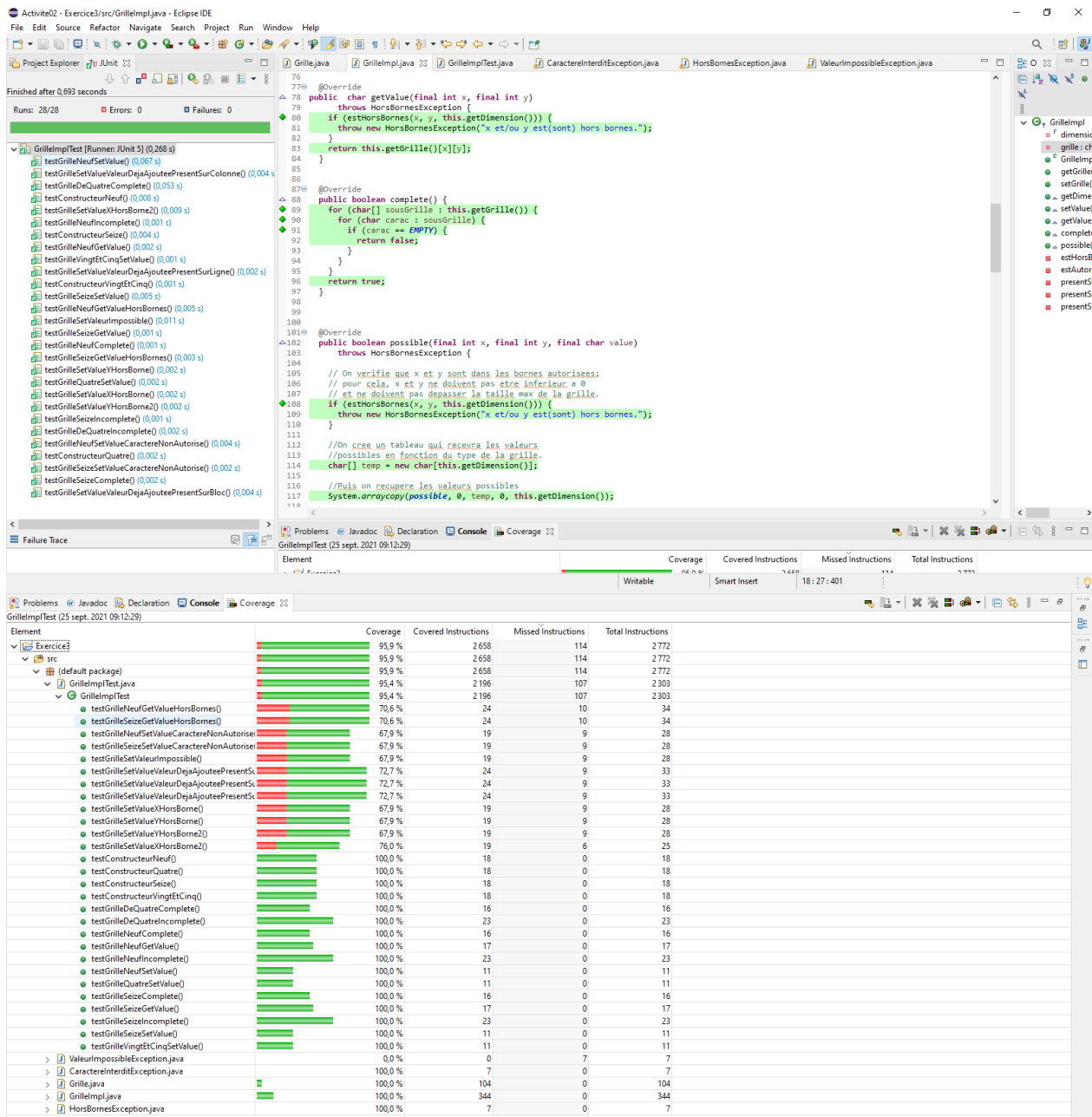
```



Nom : NDOURDOM
Prénom : Ezéchiél
Devoir n°2, Décembre 2021
Nombre de pages : 32

Code du Module : C306
Intitulé du module : Ingénierie du logiciel (ISI_05)
Nom du tuteur : Nissrine Nehiri

4. Donnons le résultat des tests unitaires, la couverture de code sur l'exécution de ces tests unitaires, les rapports checkstyle, Spotbugs et PMD.





Consortium International e-Miage
&
Université de Picardie Jules Verne d'Amiens
Master MIAGE

Nom : NDOURDOM
Prénom : Ezéchiél
Devoir n°2, Décembre 2021
Nombre de pages : 32

Code du Module : C306
Intitulé du module : Ingénierie du logiciel (ISI_05)
Nom du tuteur : Nissrine Nehiri

Activite02Exercice03

Documentation sur le projet

- Info Projet
- Rapports Projet
 - JaCoCo
 - Checkstyle
 - Sources Xref
 - Sources Xref des Tests
 - SpotBugs
 - CPD
 - PMD
 - Surefire Report

Built by

Résultats Checkstyle

Le document suivant contient les résultats de Checkstyle 6.18 avec les règles sun_checks.xml.

Résumé

Fichiers	Infos	Avertissements	Erreurs
6	0	0	0

Fichiers

Fichier	I	A	E
---------	---	---	---

Règles

Catégorie	Règle	Violations	Sévérité
-----------	-------	------------	----------

Détails

Copyright © 2021.All rights reserved.

Activite02Exercice03

Documentation sur le projet

- Info Projet
- Rapports Projet
 - JaCoCo
 - Checkstyle
 - Sources Xref
 - Sources Xref des Tests
 - SpotBugs
 - CPD
 - PMD
 - Surefire Report

Built by

Rapport SpotBugs de détection des bugs

Le document suivant contient les résultats de SpotBugs

La version de SpotBugs est 4.0.4

La limite est *medium*

L'effort est *default*

Résumé

Classes	Bugs	Erreurs	Classes Manquantes
5	1	0	0

Fichiers

Classe	Bugs
Grille	1

Grille

Bug	Catégorie	Détails	Ligne	Priorité
Grille.POSSIBLE devrait être sorti de l'interface et mis en package protected	MALICIOUS_CODE	MS_OOI_PKGPROTECT	12	Medium

Copyright © 2021.All rights reserved.

Activite02Exercice03

Documentation sur le projet

- Info Projet
- Rapports Projet
 - JaCoCo
 - Checkstyle
 - Sources Xref
 - Sources Xref des Tests
 - SpotBugs
 - CPD
 - PMD
 - Surefire Report

Built by

Résultats PMD

Le document suivant contient les résultats de PMD 6.29.0.

PMD found no problems in your source code.

Copyright © 2021.All rights reserved.