**Form Diskusi Pertemuan 7**

**Exception Handling**

**Tujuan diskusi:**

* Mahasiswa mampu mengimplementasi exception handling dalam pemrograman dengan menggunakan bahasa Java.
* Mahasiswa mampu menuliskan user-defined exception untuk suatu kasus tertentu.

**Diskusi 1: Exception Handling 01**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. Buatlah project dengan nama **Diskusi71** 2. Buat sebuah kelas **Bilangan** dengan code program berikut ini:   import java.util.\*;  public class Bilangan {  public static void main (String args[]) {  Scanner in = new Scanner(System.in);  int array[]={0,2,5};  int idx\_1, idx\_2;  System.out.print("Masukkan idx\_1 : ");  idx\_1 = in.nextInt();  System.out.print("Masukkan idx\_2 : ");  idx\_2 = in.nextInt();  System.out.println("Hasil Pembagian : " +  array[1]/array[idx\_1]);  System.out.println("Hasil Penjumlahan : " +  (array[2]+array[idx\_2]));  }  }   1. Eksekusi program dengan memasukkan nilai dengan kombinasi sebagai berikut:  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | No | Idx\_1 | Idx\_2 | array[idx\_1] | array[idx\_2] | Hasil eksekusi | | 1 | 1 | 1 | **2** | **2** | **1, 7** | | 2 | 1 | 3 | **2** | **-** | **1, Index 3 out of bounds for length 3** | | 3 | 0 | 1 | **0** | **2** | **Divided by zero, 7** | | 4 | 0 | 3 | **0** | **-** | **Divided by zero, Index 3 out of bounds for length 3** |  1. Dari langkah 3, pada poin **no. 2 dan 3,** terdapat beberapa exception yang terjadi. Berdasarkan exception tersebut, ubahlah kode program pada langkah 02 di atas menjadi kode program yang dilengkapi dengan exception handling sesuai dengan exception yang terjadi. Untuk setiap terjadi exception, program menampilkan pesan “Terjadi exception dengan tipe exception [nama exception]”. Anda dapat memanfaatkan daftar fungsi berikut.  |  |  | | --- | --- | | fungsi | Kegunaan | | public String getMessage() | Returns a detailed message about the exception that has occurred. | | public String toString() | Returns the name of the class concatenated with the result of getMessage(). | | public void printStackTrace() | Prints the result of toString() along with the stack trace to System.err, the error output stream. |  1. Eksekusi program dengan memasukkan nilai dengan kombinasi sebagai berikut:  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | No | Idx\_1 | Idx\_2 | array[idx\_1] | array[idx\_2] | Hasil eksekusi | | 1 | 1 | 1 |  |  |  | | 2 | 1 | 3 |  |  |  | | 3 | 0 | 1 |  |  |  | | 4 | 0 | 3 |  |  |  |  1. Kode program dengan exception …. |
| **Tuliskan code yang telah diubah di sini (screenshot)….** |
|  |

**Diskusi 2: ExceptionHandling 02**

|  |
| --- |
| 1. Tetap menggunakan project pada diskusi 1 di atas, lakukan eksekusi dengan memberikan inputan : **idx\_1 = e, idx\_2 = 2** 2. Eksekusi kode program. 3. Exception yang terjadi adalah …. |
| **Tuliskan hasil yang terjadi di sini (screenshot)….** |
| 1. Ubah kode program hasil diskusi 1 di atas, sehingga exception yang terjadi dapat ditangani. |
| **Tuliskan code yang telah diubah di sini (screenshot)….** |
| 1. Eksekusi kode program. 2. Hasil Eksekusi dari instruksi nomor 5 |
| **Tuliskan hasil eksekusi kode dari instruksi nomor 5 di sini (screenshot)..** |

**Diskusi 3: User-defined Exception 01**

|  |
| --- |
| 1. Buatlah project dengan nama **Diskusi73** 2. User-defined exception merupakan exception yang dibuat oleh pengguna dengan tujuan untuk membuat tipe baru yang dapat memberikan informasi diagnostik yang lebih baik. User-defined exception class merupakan turunan dari kelas **Exception**. 3. Buat sebuah kelas **Circle** dengan code program berikut ini:   class Circle  {  double x, y, r;  public Circle (double centreX, double centreY, double radius )  {  x = centreX ; y = centreY; r = radius;  }  }   1. Buat sebuah kelas **CircleTest** dengan code program berikut ini:   public class CircleTest {  public static void main(String[] args) {  Circle c1 = new Circle(10, 10, 2);  System.out.println("Circle created");  }  }   1. Eksekusi kode program tersebut 2. Catat hasil pengamatan Anda |
| **Tuliskan hasil pengamatan anda disini (screenshot)….** |
| 7. Modifikasilah instruksi nomor 4 dengan mengganti nilai dari x, y, dan r sebagai berikut :  Circle c1 = new Circle(5, 5, -1);  8. Eksekusi kode program tersebut  9. Jika code anda benar, maka hasil yang didapat adalah **“Circle created”**  10. Bagaimana jika kita ingin membatasi nilai radius yang diinputkan tidak boleh 0 (nol) atau negative? Untuk melakukan hal ini kita bisa menggunakan user defined exception sebagai berikut.  11. Buatlah kelas **InvalidRadiusException** berisi code program berikut.  import java.lang.Exception;  class InvalidRadiusException extends Exception {  private double r;  public InvalidRadiusException(double radius){  r = radius;  }  public void printError(){  System.out.println("Radius [" + r + "] tidak boleh nol");  }  }  12. Buatlah kelas **NegativeRadiusException** berisi code program berikut.  import java.lang.Exception;  class NegativeRadiusException extends Exception {  private double r;  public NegativeRadiusException (double radius){  r = radius;  }  public void printError(){  System.out.println("Radius [" + r + "] tidak boleh negative");  }  }  13. Modifikasilah konstruktor yang ada pada kelas Circle dengan code berikut.  public Circle (double centreX, double centreY, double radius ) **throws InvalidRadiusException, NegativeRadiusException**  {  **if (radius < 0 )**  **{**  **throw new NegativeRadiusException(radius);**  **}**  **else if (radius == 0 )**  **{**  **throw new InvalidRadiusException(radius);**  **}**  else  {  x = centreX ; y = centreY; r = radius;  }  }  14. Modifikasilah kelas CircleTest dengan code berikut.  public class CircleTest {  public static void main(String[] args) {  **try {**  Circle c1 = new Circle(10, 10, -1);  System.out.println("Circle created");  **}**  **catch(InvalidRadiusException e)**  **{**  **e.printError();**  **}**  **catch(NegativeRadiusException e)**  **{**  **e.printError();**  **}**  }  }  15. Eksekusilah kode program tersebut dengan mengisikan pada konstruktornya  Circle c1 = new Circle(10, 10, -5); |
| 16. Eksekusilah kode program tersebut dengan mengisikan pada konstruktornya  Circle c1 = new Circle(10, 10, 0); |
| 17. Ceritakan perbedaan hasil eksekusi pada instruksi nomor 15 dan 16. |
| **Tuliskan di sini….** |

**Diskusi 4: User-defined Exception 02**

|  |
| --- |
| 1. Buatlah project dengan nama **Diskusi74** 2. Buatlah 3 kelas yang ada dalam hirarki pewarisan pada gambar 1, yaitu kelas **Bentuk2D, Segitiga, PersegiPanjang, Lingkaran, dan Bujursangkar.** Atribut, tipe atribut beserta dengan fungsi yang ada pada masing-masing kelas dapat dilihat pada gambar 1 tersebut. **Kode program dapat di-download di situs kuliah.**   Gambar 1. Hirarki bentuk dua dimensi.   1. Diinginkan, untuk setiap pembuatan objek dari kedua kelas **PersegiPanjang** dan **Lingkaran**, **nilai atribut-atribut tidak diperbolehkan bernilai negatif.** 2. Buatlah Sebuah kelas exception, **InvalidWideLengthException**. Lihat contoh diskusi 3 mulai langkah 10 di atas. 3. Lakukan perubahan di fungsi kontruktor pada kelas **PersegiPanjang**. Perubahan dilakukan sebagai berikut:  * Jika nilai parameter aktual (panjang dan lebar) pada fungsi konstruktor kelas **PersegiPanjang** bernilai positif, maka lakukan inisialisasi nilai panjang dan lebar di kelas **PersegiPanjang**. Jika tidak maka terjadi sebuah exception (throw exception, lihat contoh di diskusi 3).  1. Buatlah kelas **UserDefinedExceptionHandlingTest** berisi kode program berikut.   import java.util.\*;  public class UserDefinedExceptionHandlingTest {  public static void main(String[] args) {  try {  double panjang, lebar;  PersegiPanjang pp;    Scanner in = new Scanner(System.in);  System.out.print("Masukkan panjang : ");  panjang = in.nextInt();  System.out.print("Masukkan lebar : ");  lebar = in.nextInt();    Bentuk2D b = new PersegiPanjang(panjang, lebar);  b.display();  }  catch (InvalidWideLengthException e1) {  e1.printError();  }  }  }   1. Eksekusi kode program. 2. Buat lagi sebuah kelas exception, **InvalidRadiusException**. Lihat langkah 4 di atas. 3. Lakukan perubahan di fungsi kontruktor pada kelas **Lingkaran**. Perubahan dilakukan sebagai berikut:  * Jika nilai parameter aktual (radius) pada fungsi konstruktor kelas **Lingkaran** bernilai positif, maka lakukan inisialisasi nilai radius di kelas **Lingkaran**. Jika tidak maka terjadi sebuah exception (throw exception, lihat contoh di diskusi 3).      1. Lakukan perubahan terhadap **UserDefinedExceptionHandlingTest** berisi kode program berikut.   import java.util.\*;  public class UserDefinedExceptionHandlingTest {  public static void main(String[] args) {  try {  double panjang, lebar, **radius**;  PersegiPanjang pp;  **Lingkaran ls;**    Scanner in = new Scanner(System.in);  System.out.print("Masukkan panjang : ");  panjang = in.nextInt();  System.out.print("Masukkan lebar : ");  lebar = in.nextInt();  **System.out.print("Masukkan radius : ");**  **radius = in.nextInt();**    Bentuk2D b = new PersegiPanjang(panjang, lebar);  b.display();    **ls = new Lingkaran(radius);**  **b = ls;**  **b.display();**  }  catch (InvalidWideLengthException e1) {  e1.printError();  }  **catch (InvalidRadiusException e2) {**  **e2.printError();**  **}**  }  }   1. Eksekusi kode program. 2. Tuliskan secara singkat cara menyusun user-defined exception dan cara menggunakan exception tersebut. |
| **Tuliskan disini….** |

**No Kelompok :**

**Anggota :**

|  |  |
| --- | --- |
| **NPM Anggota** | **Nama Anggota** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |