



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота №3

Exceptions

Виконали

студенти групи ІА-23:

Волошин Вадім Вікторович

Воронюк Євгеній Владиславович

Савонік Назар Анатолійович

Перевірив:

Колеснік Валерій Миколайович

Київ 2023

Хід роботи:

Ознайомитись з javadoc для наступних класів: Throwable; Error; Exception; RuntimeException. А також повторити конструкції try-catch-finally, throw, throws.

Проаналізувати предметну область з л/р 10-12. Створити свій власний тип checked exception (підклас класу Exception), що описує порушення одного з бізнес-правил предметної області. Додати throws-декларацію цього exception до відповідних методів, у яких потрібно «викидати» його у разі порушення бізнес-правил. Додати блок try-catch-finally для демонстрації виклику цього метода. Продемонструвати «викидання» та обробку цього exception. Також додати та продемонструвати код для «викидання» кількох стандартних RuntimeException (IllegalArgumentException, NullPointerException, ...).

Виконання завдання:

CheckedException.java {}

```
public class CheckedException extends Exception {  
    public CheckedException(String message) {  
        super(message);  
    }  
}
```

Company.java {}

```
import java.util.*;  
public class Company {  
    private static final int max = 4;  
    ArrayList<Employee> comp = new ArrayList<>();  
  
    public void addComp(Employee employee) throws CheckedException {  
        if(comp.size() >= max) {  
            throw new CheckedException("Too many employees in the list");  
        }  
        comp.add(employee);  
    }  
    private String getEmployee() {  
        return comp.toString();  
    }  
    public static void main(String[] args) {  
        Company company = new Company();  
        try {  
            company.addComp(new Employee("John"));  
            company.addComp(new Employee("Tom"));  
            company.addComp(new Employee("Tommy"));  
            company.addComp(new Employee("Will"));  
            company.addComp(new Employee("Sam"));  
        } catch (CheckedException e) {  
            System.out.println("Error: " + e.getMessage());  
        } finally {  
            System.out.println("Employee list: " + company.getEmployee());  
        }  
    }  
}
```

run: Company.java {}

```
Error: Too many employees in the list  
Employee list: [John, Tom, Tommy, Will]  
  
Process finished with exit code 0
```

Employee.java {}

```
public class Employee {  
    private String name;  
  
    public String toString() {  
        return name;  
    }  
    Employee(String name) {  
        this.name = name;  
    }  
}
```

RuntimeException.java {}

```
public class RuntimeException {  
    public static void main(String[] args) {  
        try {  
            int a = 1 / 0;  
        } catch (ArithmeticException e) {  
            System.out.println("Error: Dividing by zero");  
        }  
        int[] arr = new int[] {1, 2, 3, 4, 5, 6, 7};  
  
        try {  
            System.out.println(arr[8]);  
        } catch (ArrayIndexOutOfBoundsException e) {  
            System.out.println("Error: Invalid array index");  
        } finally {  
            System.out.println("Array length is " + arr.length);  
        }  
    }  
}
```

run: RuntimeException.java {}

```
Error: Dividing by zero  
Error: Invalid array index  
Array length is 7  
  
Process finished with exit code 0
```

Висновок: під час виконання даної лабораторної роботи ми проаналізували предметну область з л/р 10-12. Створили свій власний тип checked exception, що описує порушення одного з бізнес-правил предметної області - [Error: too many employees in the list]. Додали

блок try-catch-finally для демонстрації виклику цього метода. Також додали код для «викидання» стандартного RuntimeException – ArrayIndexOutOfBoundsException. Ознайомились із javadoc для наступних класів: Throwable; Error; Exception; RuntimeException; повторили конструкції try-catch-finally, throw, throws та відповіли на контрольні запитання.