

Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки

Кафедра автоматики та управління в технічних системах

Лабораторна робота №9

**Основи програмування**

*Варіант 25*

|  |  |  |
| --- | --- | --- |
| Виконав  студент групи ІТ-03: |  | Перевірила: |
|  |  |  |
|  |  |  |
|  |  |  |
| Чабан А. Є. |  | Смолій В. В. |

Київ 2020

**Тема**:String

**Хід роботи**:

1. Ознайомитись з API класу String. Особливу увагу звернути на методи:

- split

- toCharArray

- toLowerCase

- toUpperCase

2. Виконати завдання з таблиці 2 відповідно до свого варіанту у таблиці 1.

- Кожне завдання має бути реалізовано як окремий клас.

- Кожен клас має складатись щонайменше з двох методів:

- public static void main(String[] agrs) - точка входу. Містить код, що кілька разів знаходить результат завдання при різних значеннях аргументів та параметрів. Для перевірки мають бути присутні як дозволені так і заборонені комбінації аргументів та параметрів.

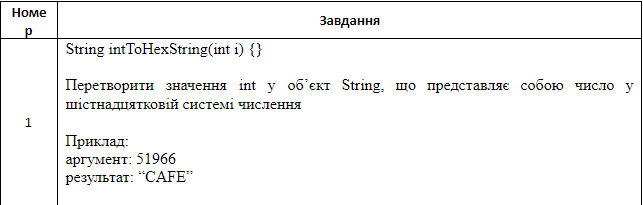
- Метод, що реалізує задане завдання. Метод має перевіряти аргументи та у разі їх помилковості аварійно закінчувати свою роботу шляхом викидання стандартного виключення IllegalArgumentException або NullPointerException (дивись л/р №3). В жодному разі цей метод не повинен напряму взаємодіяти з користувачем через консоль або інший UI (ніколи не змішуйте бізнес-логіку та користувацький інтерфейс).

- Клас може містити інші допоміжні методи.

3. Відповісти на контрольні питання

**Виконання:**

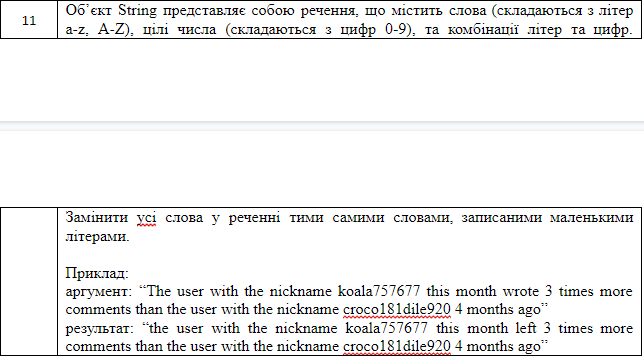
1.

  
і

Реалізуємо завдання в окремому класі:

*package* ua.kpi.fict.acts.it03.lab9;  
  
*public class* ToHex {  
 String intToHexString (*int* i){  
 *if*(i < 0){  
 *throw new* IllegalArgumentException("Argument is less than 0");  
 }  
 String ans = Integer.*toHexString*(i);  
 *return* ans.toUpperCase();  
 }  
}

Також реалізуємо наступне завдання в окремому класі:

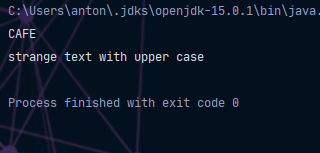


*package* ua.kpi.fict.acts.it03.lab9;  
  
*public class* Task11 {  
 String converted(String s) {  
 *if* (s.length() <= 0) {  
 *throw new* IllegalArgumentException("Bad string");  
 }  
 *return* s.toLowerCase();  
 }  
}

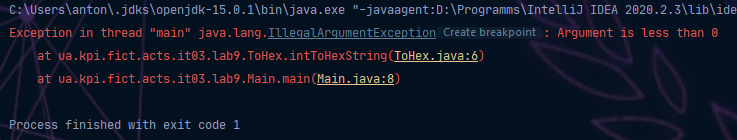
А тепер виконаємо все це в класі main та перевіримо правильність виконання:

*package* ua.kpi.fict.acts.it03.lab9;  
  
*public class* Main {  
 *public static void* main(String[] args) {  
 ToHex hexer = *new* ToHex();  
 Task11 lower = *new* Task11();  
 *int* i1 = 51966;  
 String ans1 = hexer.intToHexString(i1);  
 System.out.println(ans1);  
 *////////////////////////*  
String testStr = "sTrAnGe TExT wITh uPpEr CaSE";  
 String ans2 = lower.converted(testStr);  
 System.out.println(ans2);  
 }  
}

В результаті виконання коду маємо такий результат як і очікувалося:



Результат коду при спрацюванні exception:



**Контрольні запитання:**

1) Immutable об’єкти не змінюються після створення. Об’єкти класу String immutable задля можливості кешування, безпеки та синхронізації. Клас String final, оскільки не потребує змін та розширень.

2) Регулярні вирази – це набір синтаксичних правил за якими можна знайти необхідний рядок.



3) Метод equals() порівнює об’єкти, а оператор «==» порівнює примітиви, тому «==» буде перевіряти не об’єкти, а посилання на них. == - підходить для порівняння чисел, а equals() – підходить для порівняння рядків та об’єктів.

4) StringBuffer, StringBuilder – це не immutable об’єкти, тобто працюють швидше. StringBuffer, StringBuilder дозволяють робити більше операцій над рядками.

5) Можна використати метод trim() класу String.