

SOFTIE
SZKOŁA IT

Tester manualny z wprowadzeniem
do automatyzacji
Zadania domowe

Task 1.

Write a program that displays "This course is easy!" 5 times. Each time in a new line.

Task 2.

Uncomment one line so that the program displays the numbers 30 and 5.

```
public static void main(String[] args) {  
    int x = 5;  
    int y = 7;  
  
    // y = x * y;  
    // y = x + y;  
  
    x = y - x;  
    y = y - x;  
  
    System.out.println(x);  
    System.out.println(y);  
}
```

Task 3.

Write a program that displays the square of current year on the screen.

```
public static void main(String[] args) {  
    //your code  
}  
  
public static int square(int a) {  
    return a * a;  
}
```

Task 4.

In the print method, display the passed string 5 times.

```
public static void main(String[] args) {  
    print("I want to learn Java!");  
    print("Being an automation engineer is nice!");  
}  
  
public static void print(String s) {  
    //your code  
}
```

Task 5.

Write a function that computes the minimum of 4 numbers.

Use the following code. The function minimum(a, b, c, d) should use the function min(a, b).

```

public static void main(String[] args) throws Exception {
    System.out.println(min(-10, -5));
    System.out.println(min(-10, -5, -50, -90));
    System.out.println(min(-20, -10, -30, 90));
    System.out.println(min(-80, -1, -75, 40));
}
public static int min(int a, int b, int c, int d) {
    //your code
}

public static int min(int a, int b) {
    //your code
}

```

Task 6.

Add the public static void printPassedString(String s) method, and write code that will make it display the passed string.

```

public static void main(String[] args) {
    printPassedString("Hello World!");
}

```

//your code

Task 7.

Write code for the convertUsdToPln method, which converts dollars to polish zlotys at a given exchange rate.

Use a return statement to return the result from the method. In the main method call the method 3 times and display results.

```

public static void main(String[] args) {
    //your code
}

public static double convertUsdToPln(int usd, double exchangeRate) {
    //your code
}

```

Task 8.

Display a 10 x 10 multiplication table in the following form:

```

1 2 3 4 ...
2 4 6 8 ...
3 6 9 12 ...
4 8 12 16 ...
...

```

Task 9.

Enter the number and name from the keyboard. Display the following string:

<name> will be an automation tester in <number> months!

Task 10.

Write the code for `sumDigits(int number)`. You need to calculate the sum of four digits of this number, and then return the result.

Hint: Use `'%10'` operation to check the last digit.

```
public static void main(String[] args) {
    System.out.println(sumDigits(4912));
}

public static int sumDigits(int number) {
    //your code
}
```

Task 11.

Fix one of the most common errors in the code below so that the person's age is changed.

```
public static void main(String[] args) {

    Person person = new Person();
    System.out.println("Age: " + person.age);
    person.changeAge(person.age);
    System.out.println("New age: " + person.age);
}

public static class Person {
    public int age = 20;

    public void changeAge(int age) {
        age = age + 20;
        System.out.println("The age in changeAge() is " + age);
    }
}
```

Task 12.

Calculate the total cost of beer.

```
public static void main(String[] args) {
    Beer beer1 = new Beer();
    beer1.addPrice(5);
    Beer beer2 = new Beer();
    beer2.addPrice(12);
    System.out.println("The cost of beer is " + Beer.beerPrice);
}
```

```

public static class Beer {
    public static int beerPrice = 0;

    public static void addPrice(int beerPrice) {
        //your code
    }
}

```

Task 13.

Use the keyboard to enter a number. If the number is positive, then double it. If the number is negative, raise it to power 2. If the entered number is zero, don't change it. Display the result on the screen.

Task 14.

Use the keyboard to enter a year, and then determine the number of days in the year. For a leap year – 366, for an ordinary one – 365.

Task 15.

Use the keyboard to enter three numbers a, b, and c (the lengths of the sides of the proposed triangle). Determine whether a triangle with these sides can exist.

Display the result as follows:

"The triangle is possible." – if a triangle with these sides could exist.

"The triangle is not possible." – if a triangle with these sides cannot exist.

Task 16.

Use the keyboard to enter an integer. Display information on the console whether it's a positive/negative and even/odd number, or 0. Run the program until user provides number 1000.

Input: -15

Output: Negative, odd number.

Task 17.

Create a method which will return the number of digits in a passed number.

Input: 91231

Output: 5

Task 18.

Use the keyboard to enter two digits: a and b. Using **for** loop display a x b size rectangle made of 0's.

Input: a = 5, b = 7

Output:

0000000

0000000

0000000

0000000

0000000

Task 19.

Use the keyboard to enter 3 numbers and display the middle one (not the average).

Task 20.

Read numbers from the keyboard and after typing number 0, display the sum of all previously provided numbers.

Task 21.

Use keyboard to enter a name and 3 numbers (year, month and a day). Display data in the following format:

"My name is {name}, I was born on day/month/year."

Task 22*.

Create a Soldier class. The soldier should have 4 attributes: age, weight, strength and agility. Also implement boolean fight(Soldier anotherSoldier) method that will return true if a fight was won and false if it was lost. Design your own fight algorithm.

Task 23.

Use keyboard to enter numbers until user enters -1. Then display the average of all entered numbers excluding the last one.

Task 24.

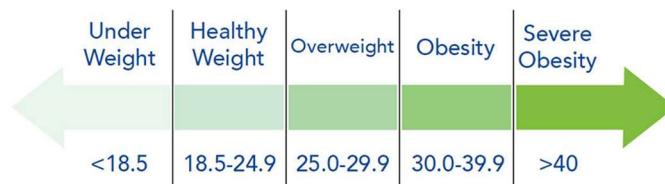
Rewrite the last exercise but this time display the average after user types "exit".

Task 25.

Use keyboard to enter height in meters and weight in kilograms and calculate BMI of the person. Display the category based on the graphics.

Body Mass Index = weight / (height * height)

Weight Categories Based on BMI



Task 26.

Implement a method that will take two arguments – String *s* and int *n*, and return *s* multiplied by *n* times.

Input: "Course", 3

Output: CourseCourseCourse

Task 27.

Implement a method that will take an array of integers as a parameter and will return it sorted descending.

Task 28.

Create a method that will take a list of Strings and an integer *n* as parameters and return a list of Strings. Within the method take the last element of the list and put it at the beginning for *n* times. Display returned list.

Task 29.

Create a Cat class with fields: name, colour, hasOwner, isHungry. Generate 5 cats and add them to a HashSet. Create a method that displays all cat's names from the passed set.

Task 30.

Implement a method that will take a list of Strings as a parameter and return a list with doubled values.

Input: "uno", "dos", "tres"

Output: "uno", "uno", "dos", "dos", "tres", "tres"

Task 31.*

Initialize a Set of integers. Using Random class generate an integer between 1 and 1000 and add it to the Set. Do it as long as you get the full Set of all numbers from 1 to 1000. Calculate time the algorithm needed to generate 1000 different random numbers.

Hint: Use *System.currentTimeMillis()* method before and after running the method. Then subtract first measurement from the second one and display your result.

Task 32.*

Create a list of Strings. Use the keyboard to enter 5 Strings and add them to the list. Now find the longest String in the list and display it. If there are more than 1 longest Strings, display all of them.