# Лаб: Условни конструкции

Задачи за упражнение в клас и за домашно към курса ["Основи на програмирането" @ СофтУни](https://softuni.bg/courses/programming-basics).

Тествайте решенията си в **Judge** системата: <https://judge.softuni.org/Contests/2389/Conditional-Statements-Lab>

<https://softuni.bg/trainings/resources/video/69884/video-05-march-2022-veronika-vaneva-programming-basics-with-java-march-2022/3630>

## Отлична оценка

Напише **конзолна програма**, която **чете оценка** (цяло число), въведена от потребителя и отпечатва "Excellent!**"** ако оценката е **5** или по-висока.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 6 | Excellent! | 4 | *(няма изход)* | 5 | Excellent! | 3 | *(няма изход)* |

### Насоки:

1. Създайте **нов клас** в съществуващия IntelliJ проект. Кликнете с десен бутон на мишката върху **папката "**src**"**. Изберете [New] 🡪 [Java Class]: (Package-if we want to create first Package and inside this new class to create ! )

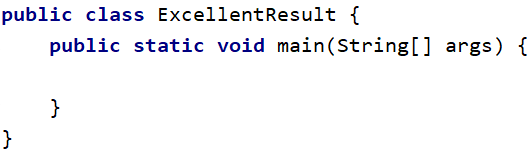
A screenshot of a social media post

Description automatically generatedA screenshot of a cell phone

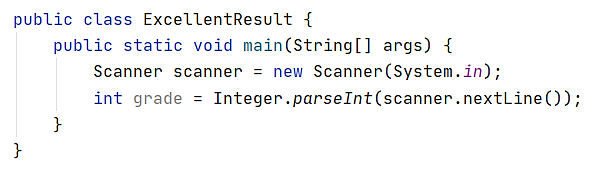
Description automatically generated

Вече имате проект с един клас в него. Остава да напишете кода за решаване на задачата.

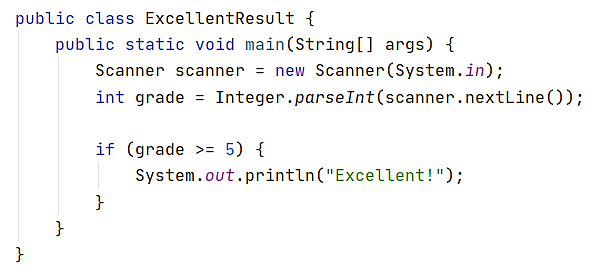
1. Създайте main **метод** като отидете в класа "ExcellentResult" (между квадратните скоби) и напишете:



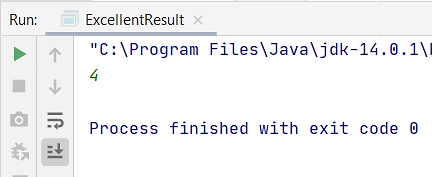
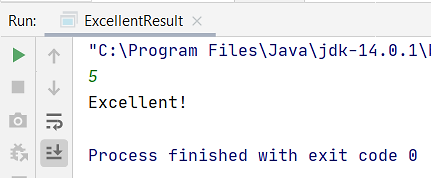
1. Отидете в тялото на метода main(String[] args) (между къдравите скоби). Създайте Scanner обект, с който да четете от конзолата и прочетете едно **реално** **число** - оценката:



1. Направете проверка за стойността на оценката. Ако тя е по-голяма или равна на 5.50, отпечатайте изхода по условие:



1. **Стартирайте** програмата с [Ctrl+Shift+F10] и я **тествайте** с различни входни стойности:



## По-голямото число

Да се напише програма, която чете **две цели числа**, въведени от потребителя и отпечатва **по-голямото от двете**.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 5  3 | 5 | 3  5 | 5 | 10  10 | 10 | -5  5 | 5 |

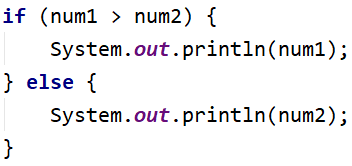
### Насоки:

1. Прочетете две цели числа от конзолата:

A picture containing indoor

Description automatically generated

1. Сравнете дали първото число num1 e по-голямо от второто num2. Отпечатайте по-голямото число.



## Четно или нечетно

Да се напише програма, която чете **цяло число**, въведено от потребителя и печата дали е **четно** или **нечетно**.

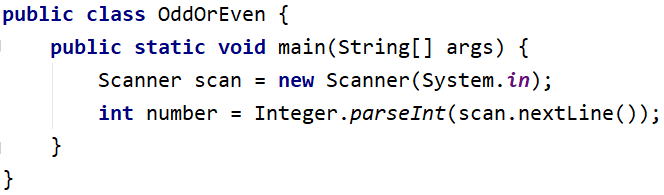
Ако е **четно** отпечатайте "**even"**, ако е нечетно "**odd"**.

### **Примерен вход и изход**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 2 | even | 3 | odd | 25 | odd | 1024 | even |

### Насоки:

1. Първо добавете **нов Java клас** към съществуващия проект.
2. Създайте Scanner обект и прочетете едно цяло число от конзолата:



1. Проверете дали числото е четно като извършите деление на 2 с остатък и проверите дали има остатък от делението. Отпечатайте изхода по условие – текста "even":

A screenshot of a cell phone

Description automatically generated

1. В противен случай отпечатайте "odd":

A screenshot of a cell phone

Description automatically generated

## Познай паролата

Да се напише програма, която **чете парола** (текст), въведена от потребителя и проверява дали въведената парола **съвпада** с фразата "s3cr3t!P@ssw0rd".При съвпадение да се изведе "**Welcome**". При несъвпадение да се изведе "**Wrong password!**".

### Примерен вход и изход

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| qwerty | Wrong password! | s3cr3t!P@ssw0rd | Welcome | s3cr3t!p@ss | Wrong password! |

## Число от 100 до 200

Да се напише програма, която **чете цяло число**, въведено от потребителя и проверява дали е **под 100**, **между 100 и 200** или **над 200**. Ако числото е:

* под 100 отпечатайте: **"Less than 100"**
* между 100 и 200 отпечатайте: **"Between 100 and 200"**
* над 200 отпечатайте: **"Greater than 200"**

### Примерен вход и изход

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 95 | Less than 100 | 120 | Between 100 and 200 | 210 | Greater than 200 |

## Информация за скоростта

Да се напише програма, която **чете скорост** **(реално число)**, въведена от потребителяи отпечатва **информация за скоростта**.

* При скорост **до 10** (включително) отпечатайте **"slow"**
* При скорост **над 10** и **до 50** (включително) отпечатайте **"average"**
* При скорост **над 50** и **до 150** (включително) отпечатайте **"fast"**
* При скорост **над 150** и **до 1000** (включително) отпечатайте **"ultra fast"**
* При по-висока скорост отпечатайте **"extremely fast"**

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 8 | slow | 49.5 | average | 126 | fast | 160 | ultra fast | 3500 | extremely fast |

## Лица на фигури

Да се напише програма, в която потребителят **въвежда вида и размерите на геометрична** фигура и пресмята лицето й. Фигурите са четири вида: квадрат (**square**), правоъгълник (**rectangle**), кръг (**circle**) и триъгълник (**triangle**). На първия ред на входа се чете вида на фигурата (текст със следните възможности: square, rectangle, circle или triangle).

* Ако фигурата е **квадрат (**square): на следващия ред се чете едно дробно число - дължина на страната му
* Ако фигурата е **правоъгълник (**rectangle): на следващите два реда четат две дробни числа - дължините на страните му
* Ако фигурата е **кръг (**circle): на следващия ред чете едно дробно число - радиусът на кръга
* Ако фигурата е **триъгълник (**triangle): на следващите два реда четат две дробни числа - дължината на страната му и дължината на височината към нея

Резултатът да се закръгли до **3 цифри след десетичната запетая**.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| square  5 | 25.000 | rectangle  7  2.5 | 17.500 | circle  6 | 113.097 | triangle  4.5  20 | 45.000 |

“ == “ We compare the addresses into the stack memory ,;

By Referent types using == we compare the addresses into the memory, not the real values of the objects !!!

We use equals() method to compare the referent data values !!!!

Task 1 ;

package conditionalStatements;  
  
import java.util.Scanner;  
  
public class ExcellentResult\_01 {  
 public static void main(String[] args) {  
  
 Scanner scanner = new Scanner(System.*in*);  
 int grade = Integer.*parseInt*(scanner.nextLine());  
  
 if (grade >= 5) {  
 System.*out*.println("Excellent!");  
 }  
 }  
}

Task 2;

package conditionalStatements;  
  
import java.util.Scanner;  
  
public class GreaterNumber\_02 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 int firstNum = Integer.*parseInt*(scanner.nextLine());  
 int secNum = Integer.*parseInt*(scanner.nextLine());  
  
 if (firstNum > secNum) {  
 System.*out*.println(firstNum);  
 } else {  
 System.*out*.println(secNum);  
 }  
 }  
}

Task 3 .

package conditionalStatements;  
  
import java.util.Scanner;  
  
public class EvenOrOdd\_03 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 int num = Integer.*parseInt*(scanner.nextLine());  
  
 if (num % 2 == 0) {  
 System.*out*.println("even");  
 } else {  
 System.*out*.println("odd");  
 }  
 }  
}

Sec Way Task 3 ->

package conditionalStatements;  
  
import java.util.Scanner;  
  
public class EvenOrOdd\_03 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 int num = Integer.*parseInt*(scanner.nextLine());  
 boolean isEven = num % 2 == 0;  
  
 if (isEven) {  
 System.*out*.println("even");  
 } else {  
 System.*out*.println("odd");  
 }  
 }  
}

1.35 Mins !!!

Task 4;

package conditionalStatements;  
  
import java.util.Scanner;  
  
public class PasswordGuess\_04 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 String inputText = scanner.nextLine();  
  
 if (inputText.equals("s3cr3t!P@ssw0rd")) {  
 System.*out*.println("Welcome");  
 } else {  
 System.*out*.println("Wrong password!");  
 }  
 }  
}

Task 5 ..

package conditionalStatements;  
  
import java.util.Scanner;  
  
public class NumberFrom100To200\_05 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 int num = Integer.*parseInt*(scanner.nextLine());  
  
 if (num < 100) {  
 System.*out*.println("Less than 100");  
 } else if (num >= 100 && num <= 200) {  
 System.*out*.println("Between 100 and 200");  
 } else {  
 System.*out*.println("Greater than 200");  
 }  
 }  
}

Taks 5 Sec Way :

package conditionalStatements;  
  
 import java.util.Scanner;  
  
public class NumberFrom100To200\_05 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 int num = Integer.*parseInt*(scanner.nextLine());  
  
 if (num < 100) {  
 System.*out*.println("Less than 100");  
 } else if (num <= 200) {  
 System.*out*.println("Between 100 and 200");  
 } else {  
 System.*out*.println("Greater than 200");  
 }  
 }  
}

Task 6;

///

package conditionalStatements;  
  
import java.util.Scanner;  
  
public class SpeedInfo\_06 {  
 public static void main(String[] args) {  
  
 Scanner scanner = new Scanner(System.*in*);  
 double inputNumber = Double.*parseDouble*(scanner.nextLine());  
  
 if (inputNumber <= 10) {  
 System.*out*.println("slow");  
 } else if (inputNumber <= 50) {  
 System.*out*.println("average");  
 } else if (inputNumber <= 150) {  
 System.*out*.println("fast");  
 } else if (inputNumber <= 1000) {  
 System.*out*.println("ultra fast");  
 } else {  
 System.*out*.println("extremely fast");  
 }  
 }  
}

Task 7 ;

package conditionalStatements;  
  
import java.util.Scanner;  
  
public class AreaOfFigures\_07 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 String inputFigure = scanner.nextLine();  
 double side = 0;  
 double height = 0;  
 double area = 0;  
  
  
 if (inputFigure.equals("square")) {  
 side = Double.*parseDouble*(scanner.nextLine());  
 area = side \* side;  
 } else if (inputFigure.equals("rectangle")) {  
 side = Double.*parseDouble*(scanner.nextLine());  
 height = Double.*parseDouble*(scanner.nextLine());  
 area = side \* height;  
 } else if (inputFigure.equals("circle")) {  
 side = Double.*parseDouble*(scanner.nextLine());  
 area = Math.*PI* \* side \* side;  
 } else if(inputFigure.equals("triangle")) {  
 side = Double.*parseDouble*(scanner.nextLine());  
 height = Double.*parseDouble*(scanner.nextLine());  
 area = (side \* height) / 2;  
 }  
  
 System.*out*.printf("%.3f", area);  
  
 }  
}

Debugging -> Stepwise passing through the code !!!! to find the bug.

Life of the variable is the scope into the block code into which the variable is created and can be used ; its are the {} brackets.

GarbageCollector- deletes from the memory unusued variables and so frees up the memory !