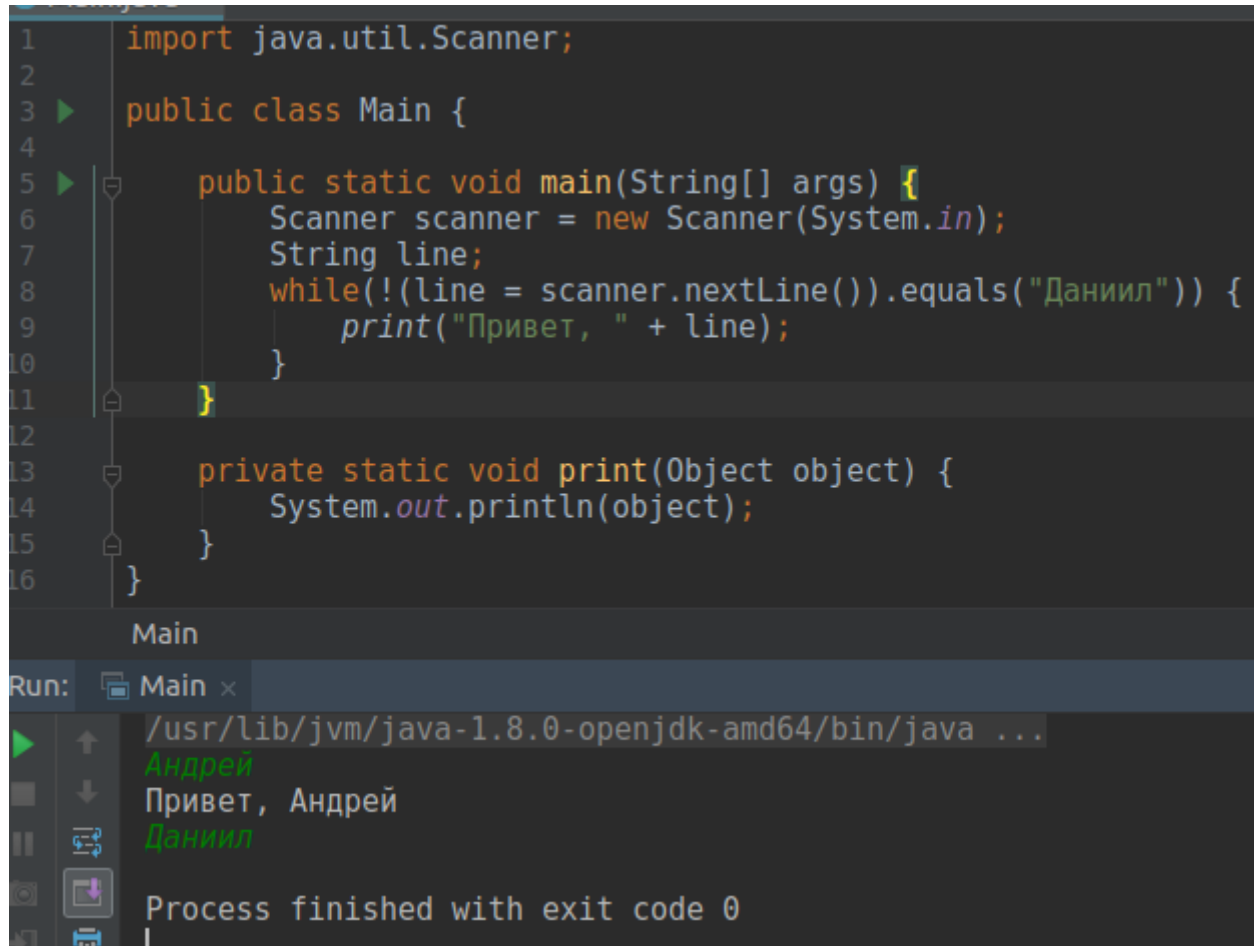


Решения задач

Из лекции 15

Задача номер 1

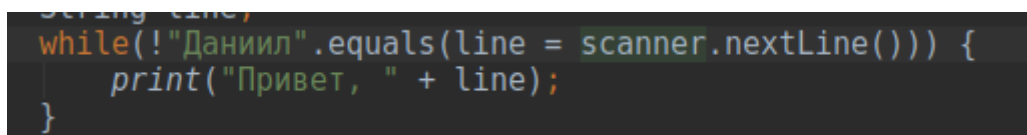


```
1 import java.util.Scanner;
2
3 public class Main {
4
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         String line;
8         while(!(line = scanner.nextLine()).equals("Даниил")) {
9             print("Привет, " + line);
10        }
11    }
12
13    private static void print(Object object) {
14        System.out.println(object);
15    }
16 }
```

Run: Main x

```
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
Андрей
Привет, Андрей
Даниил
Process finished with exit code 0
```

И давайте я вам сразу расскажу как правильно сравнивать строки. Посмотрите на линию 8. Ставим курсор на слово equals и нажимаем Alt+Enter. И выбираем Flip 'equals()'.



```
while(!"Даниил".equals(line = scanner.nextLine())) {
    print("Привет, " + line);
}
```

У нас постоянная с одной стороны и переменная с другой. Запомните, сравниваем постоянную с переменной, а не наоборот. Почему? Расскажу когда пройдем одну важную тему.

Задача номер 2

```
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    int number;
    int count = 0;
    while (count < 20) {

        number = scanner.nextInt();

        if (number % 5 == 0) {
            print(number + " кратно 5");
        } else {
            print(number + " не кратно 5");
        }

        count++;
    }
}
```

Main > main()

un: Main x

/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...

2873761

2873761 не кратно 5

1858747

1858747 не кратно 5

2342340

2342340 кратно 5

Когда будете тестировать этот код, не забывайте что мы вводим числа типа int. Т.е. не выходим за рамки 2 миллиардов. Здесь все довольно просто, перейдем к следующей задаче.

Задача номер 3

```
3 public class Main {
4
5     private static final int CREDIT_MAX = 150_000;
6     private static final int SALARY_MIN = 25_000;
7     private static final int DURATION_MAX = 12;
8     private static final String DENIED = "Извините, но вам отказано в кредите";
9
10    public static void main(String[] args) {
11        Scanner scanner = new Scanner(System.in);
12
13        print("Добрый день. Желаете взять кредит? Какая у вас зарплата?");
14        int salary = scanner.nextInt();
15        if (salary < SALARY_MIN) {
16            print(DENIED);
17        } else {
18            print("На какой срок вы бы хотели получить кредит (в месяцах)?");
19            int duration = scanner.nextInt();
20            if (duration > DURATION_MAX) {
21                print(DENIED);
22            } else {
23                print("Какую сумму вы бы хотели взять в кредит?");
24                int credit = scanner.nextInt();
25                if (credit > CREDIT_MAX) {
26                    print(DENIED);
27                } else {
28                    print("Поздравляем вас! Вам одобрен кредит на сумму " + CREDIT_MAX + "!");
29                }
30            }
31        }
32    }
33 }
```

Run: Main x Main x

/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...

Добрый день. Желаете взять кредит? Какая у вас зарплата?

27000

На какой срок вы бы хотели получить кредит (в месяцах)?

11

Какую сумму вы бы хотели взять в кредит?

130000

Поздравляем вас! Вам одобрен кредит на сумму 150000!

Process finished with exit code 0

Здесь конечно же вам стоит самому придумать логику. Но давайте посмотрим на то, как сделал ее я. Сначала я прошу указать зарплату, если она меньше минимальной, то сразу отказываем в кредите. Иначе же спрашиваем про срок, если все ок, то идем дальше, но если нет, то отказываем. И наконец спрашиваем про сумму кредита.

В чем же минус этого решения? Вложенные условия. Посмотрите на код. Чем больше вопросов тем больше вложенности. Как это решить? Мы бы могли собрать данные сразу и потом уже обработать. Давайте посмотрим как.

```
3 ▶ public class Main {
4
5     private static final int CREDIT_MAX = 150_000;
6     private static final int SALARY_MIN = 25_000;
7     private static final int DURATION_MAX = 12;
8     private static final String DENIED = "Извините, но вам отказано в кредите";
9
10 ▶ public static void main(String[] args) {
11     Scanner scanner = new Scanner(System.in);
12
13     print("Добрый день. Желаете взять кредит? Какая у вас зарплата?");
14     int salary = scanner.nextInt();
15     print("На какой срок вы бы хотели получить кредит (в месяцах)?");
16     int duration = scanner.nextInt();
17     print("Какую сумму вы бы хотели взять в кредит?");
18     int credit = scanner.nextInt();
19     if (salary < SALARY_MIN || duration > DURATION_MAX || credit > CREDIT_MAX) {
20         print(DENIED);
21     } else {
22         print("Поздравляем вас! Вам одобрен кредит на сумму " + CREDIT_MAX + "!");
23     }
24 }
25
```

Main

Run: Main x Main x

/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...

Добрый день. Желаете взять кредит? Какая у вас зарплата?

27000

На какой срок вы бы хотели получить кредит (в месяцах)?

11

Какую сумму вы бы хотели взять в кредит?

140000

Поздравляем вас! Вам одобрен кредит на сумму 150000!

Process finished with exit code 0

Здесь мы получаем все данные от юзера и только потом пишем нашу логику. Вы бы могли написать иначе

```
if (salary >= SALARY_MIN && duration <= DURATION_MAX && credit <+CREDIT_MAX)
```

то есть исходить от обратного. Удовлетворяются ли все условия сразу и только иначе отказ.

Вы можете модифицировать логику как пожелаете, главное это получить данные от юзера.

Задача номер 4

Она аналогична задаче номер 3. Вопросы другие и условие. Суть одна и та же.

Задача номер 5

```
5 public static void main(String[] args) {
6     Scanner scanner = new Scanner(System.in);
7
8     int count = 2;
9     int[] array = new int[10];
10    array[0] = scanner.nextInt();
11    array[1] = scanner.nextInt();
12    final int diff = array[1] - array[0];
13
14    while (count < 10) {
15        array[count] = scanner.nextInt();
16
17        int difference = array[count] - array[count - 1];
18
19        if (diff != difference) {
20            break;
21        }
22        count++;
23    }
24 }
25
```

Main > main()

Run: Main x

/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...

4
5
6
7
9

Process finished with exit code 0

Для арифметической последовательности нам нужны первые 2 члена чтобы высчитать разницу. После чего уже в цикле высчитываем разницу введенного числа и предыдущего. Для этих целей используем массив чисел. Как только разница оказывается другой – прерываем цикл. Да, как и в for цикле, в while тоже можно прервать его с помощью break;

Посмотрите на линию 17. Я только сейчас обнаружил что так не нужно было делать. Старайтесь не создавать внутри цикла переменную. А вынести ее до. Т.е. надо было бы написать так

```
int difference;
```

```
while(count<10)
```

```
difference = array[count] - array[count - 1];
```

На самом деле нам и не нужен массив. Мы бы могли хранить лишь последнее число и первое.

```
5 public static void main(String[] args) {
6     Scanner scanner = new Scanner(System.in);
7
8     int count = 2;
9     int first = scanner.nextInt();
10    int lastItem = scanner.nextInt();
11    final int diff = lastItem - first;
12
13    while (count < 10) {
14        first = scanner.nextInt();
15
16        if (diff != first - lastItem) {
17            break;
18        }
19        lastItem = first;
20        count++;
21    }
22 }
23
```

Main > main()

Run: Main x

/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...

3
5
7
11

Process finished with exit code 0

Даже для разницы нам не нужна переменная. Мы же всегда оперируем 2 числами. Вот и 2 переменные для первого элемента и последнего. И еще 1 переменная счетчик и константа чтобы с ней сравнивать. Старайтесь по мере возможности использовать меньше переменных и не плодить их. Тем более в цикле и тем более массивы.

Согласен, здесь немного запутанно получается при названии переменной `first`. Но вы можете назвать ее `current` и все станет ясней. Старайтесь не просто писать код чтобы он работал, а улучшать его по мере написания. А тем более после того как все верно. Вернитесь через некоторое время и подумайте как его улучшить.

```

5  ▶  public static void main(String[] args) {
6      Scanner scanner = new Scanner(System.in);
7
8      int count = 2;
9      int last = scanner.nextInt();
10     int current = scanner.nextInt();
11     final int diff = current - last;
12
13     while (count < 10) {
14         last = current;
15         current = scanner.nextInt();
16
17         if (diff != current - last) {
18             break;
19         }
20         count++;
21     }
22 }
23

```

Внимание на линию 14. Перед тем как записывать в текущее значение нужно заменить последнее значение текущим. Надеюсь всем понятно. Если нет, то подумайте или же подебажьте. Если не написать 14 линию, то мы сравним третий элемент с первым. А не со вторым. Вот и все.

Задача 6

```
1  import java.util.Scanner;
2
3  public class Main {
4
5      private static final String[] COMMANDS = new String[]{
6          "/help", "/start", "/end", "/getLocation"
7      };
8      private static final String[] RESULTS = new String[]{
9          "", "Bot started!...", "Bot ended", "location is Moscow"
10     };
11
12     public static void main(String[] args) {
13         Scanner scanner = new Scanner(System.in);
14
15         String commands = "commands available for this bot \n";
16         for (String command : COMMANDS) {
17             commands += command + "\n";
18         }
19         RESULTS[0] = commands;
20
21         String input;
22         boolean found;
23         while (!(input = scanner.nextLine()).equals(COMMANDS[2])) {
24             found = false;
25             for (int i = 0; i < COMMANDS.length; i++) {
26                 if (COMMANDS[i].equals(input)) {
27                     print(RESULTS[i]);
28                     found = true;
29                     break;
30                 }
31             }
32             if (!found) {
33                 print("no command found for that input. Try typing " + COMMANDS[0]);
34             }
35         }
36     }
37
38     private static void print(Object object) {
39         System.out.println(object);
40     }
41 }
```

Для начала нам нужны какие-то команды. Для них создаем массив константу и заполняем чем-то. Типичные команды для чатбота: старт, финиш, что-то еще. И конечно же хелп, чтобы понимать какие есть команды. Так же создаем какие-то ответы под эти команды. Важно чтобы массивы были одинаковой длины. После в результат команды хелп добавляем результатом все команды. Проходя циклом по ним. Далее просто пишем цикл: пока юзер не ввел команду финиша – берем и сравниваем ввод с теми командами которые есть. Если нашли команду – прерываем цикл и ставим флаг true. Если ничего не нашли среди команд, то просто выдаем стандартную фразу – не нашлась такая команда, попробуйте команду хелп.

И как видите все работает. Вводим что попало, нам выдает – попробуйте ввести хелп. Вводим хелп и получаем список доступных команд. Вводим что-то и после чего заканчиваем работу.

Вы можете написать еще условие, что была введена команда старт или нет. Все на ваш вкус.

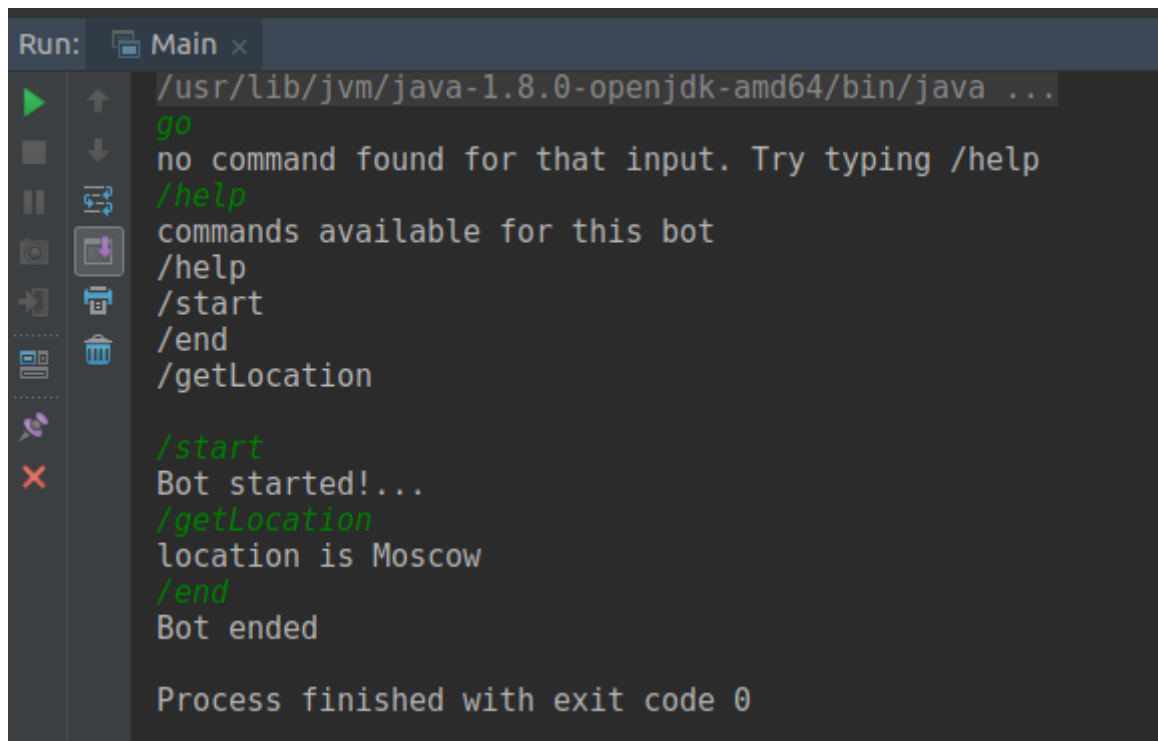

```
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
go
no command found for that input. Try typing /help
/help
commands available for this bot
/help
/start
/end
/getLocation

/start
Bot started!...
/getLocation
location is Moscow
/end

Process finished with exit code 0
```

Но кто-то скажет, что у нас есть результат для команды /end но мы ее никогда не увидим. Как же ее вывести перед завершением? Легко и просто. Напишите бесконечный цикл и внутри уже проверяйте все.

```
11
12 public static void main(String[] args) {
13     Scanner scanner = new Scanner(System.in);
14
15     String commands = "commands available for this bot \n";
16     for (String command : COMMANDS) {
17         commands += command + "\n";
18     }
19     RESULTS[0] = commands;
20
21     String input;
22     boolean found;
23     while (true) {
24         input = scanner.nextLine();
25         found = false;
26         for (int i = 0; i < COMMANDS.length; i++) {
27             if (COMMANDS[i].equals(input)) {
28                 print(RESULTS[i]);
29                 found = true;
30                 break;
31             }
32         }
33         if (COMMANDS[2].equals(input)) {
34             break;
35         }
36         if (!found) {
37             print("no command found for that input. Try typing " + COMMANDS[0]);
38         }
39     }
40 }
41 }
```



The screenshot shows a Java IDE's Run console. The title bar indicates the process is running under 'Main'. The command line shows the Java command: `/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...`. The console output shows the following sequence of events:

```
go
no command found for that input. Try typing /help
/help
commands available for this bot
/help
/start
/end
/getLocation

/start
Bot started!...
/getLocation
location is Moscow
/end
Bot ended

Process finished with exit code 0
```

The left sidebar contains standard IDE icons for running, debugging, and other actions.

Запомните эту задачу. С нее мы начнем следующую лекцию.

И я опять напомню – не стесняйтесь писать в комментариях что вам непонятно, где подробнее остановиться и т.д. Удачи!