

Programming Languages

Console I/O



import java.io.*;

❑ the basic package for operating with input/output data streams

❑ a structure of a Java program with keyboard input

1. import java.io.*;

2. In the main class,

- the method that performs the read **MUST** handle **IOException**

- an object needs to be defined – input stream, that will read the data input

through the keyboard

```
InputStreamReader isr = new InputStreamReader(System.in);
```

```
BufferedReader input = new BufferedReader(isr);
```

```
import java.io.*;
class input{
```

Example

Handling of an IO exception



```
    public static void main (String[]a) throws IOException{
        InputStreamReader isr = new InputStreamReader(System.in);
        BufferedReader input = new BufferedReader(isr);
```

```
        System.out.println("Input something");
        String text = input.readLine();
```

The object **input** reads from the keyboard

```
        System.out.println("You have entered: "+text);
```

Here, the **readLine()** method of the **input** object is invoked. This function returns the string that is input through the keyboard

```
    }
```

```
}
```

```
C:\WINDOWS\system32\cmd.exe

C:\>java input
Input something
Hello, how are you?
You have input Hello, how are you?
C:\>
```



Example

```
import java.io.*;
class input2{
    public static void main (String[]a) {
        InputStreamReader isr = new InputStreamReader(System.in);
        BufferedReader input = new BufferedReader(isr);
        String text = "";
        System.out.println("Input something");
        try{
            text = input.readLine();
        }catch(Exception e){
            System.out.println("An exception has occurred");
        }
        System.out.println("You have input: " + tekst);
    }
}
```

Trying for an IO
exception

Most commonly, IO problems are handled with
try-catch blocks for handling such exceptions

Input parsing



- The **readLine()** method returns a **String** type object.
- **Parsing** = conversion of the input String into an integer, floating-point etc.

Ex.

```
int number;
```

```
float decimal_number;
```

```
//conversion of the input into an integer
```

```
number=Integer.parseInt(input.readLine());
```

```
//conversion of the input to a floating-point
```

```
decimal_number = Float.parseFloat(input.readLine());
```



Assignment 1

1. Input your name, year of birth, height (in meters) and weight (in kilograms).
2. The program should output your name, how old you are and your body-mass index.

$$\text{body_mass_index} = \text{weight} / (\text{height} * \text{height})$$

```
C:\> C:\WINDOWS\system32\cmd.exe

C:\>java weight
What is your name? Adrijan
Which year were you born in? 1977
How tall (in meters) are you? 1.91
What is your weight (in kilograms)? 88
Your name is Adrijan
This year you'll turn 32
Your body-mass index is 24.122147

C:\>_
```



Assignment 2

1. Input from the keyboard
 1. Dimensions of a two-dimensional field
 2. Integer elements in the two-dimensional field
2. The program should calculate the sum and the average of the field.
3. It should also catch exceptions upon inputting elements in the field.

```
C:\WINDOWS\system32\cmd.exe

C:\>java field
Number of rows: 2
Number of columns: 3
el. [0][0]= 4
el. [0][1]= 7
el. [0][2]= 1
el. [1][0]= 9
el. [1][1]= 3
el. [1][2]= 2
The sum = 26
The average = 4.3333335
C:\>java field
Number of rows: 4
Number of columns: 7
el. [0][0]= 3
el. [0][1]=
An exception has occurred
The sum = 3
The average = 0.10714286
C:\>
```



Assignment 3

1. Input text from the keyboard until END gets input. For every input text, output its length and “weight”

```
...  
do{  
    // text inputting code  
    //check whether “END” was input  
    //process the input text  
}while (! text.equals(“END”));  
....
```

```
....  
// text inputting code  
while (! text.equals(“END”)){  
    // text processing code  
    //text inputting code  
} ....
```

```
....  
while (true){  
    //text inputting code  
    //if “END” was input, then  
    //break the cycle  
    // text processing code}....
```


Assignment 3 - output



```
C:\WINDOWS\system32\cmd.exe

C:\>java input

Input the text. To finish inputting, input END:hello
hello has 5 characters, and its weight is 532

Input the text. To finish inputting, input END:how are you?
how are you? has 12 characters, and its weight is 1122

Input the text. To finish inputting, input END:this is a slightly longer text
this is a slightly longer text has 30 characters, and its weight is 2897

Input the text. To finish inputting, input END:
has 0 characters, and its weight is 0

Input the text. To finish inputting, input END:eNd
Thank you for using the program.

C:\>_
```

- Create class **ACCOUNT** with

- attributes

- Bank : String
 - amount: int

- Constructor with two parameters

- Methods

- getBank() и getAmount()
 - add(plus) – increases the amount by plus
 - withdraw(minus) – if minus<=amount, decreases the amount by minus, otherwise sends a message that there is not a sufficient amount.

- In the main class **ATM**

- Create an object myAccount with initial values for a bank and amount input from the keyboard
 - While amount>0, enable the user to operate using a “menu” with the following operations:
 - **Bank** – gives the name of the bank
 - **State** – gives the current amount
 - **Add** – the client inputs a value by which the amount is increased, i.e. the method add() is invoked
 - **Withdraw** – the client inputs a value by which the amount is decreased, i.e. the method withdraw() is invoked
 - **Clear account** – an operation by which the client removes the entire amount from the account.

Assignment 4





A™

```
C:\WINDOWS\system32\cmd.exe
```

```
C:\>java ATM
Input a bank:Tutunska banka
Input the initial amount:1000

Choose from the menu

1 = Bank overview
2 = Amount overview
3 = Cash addition
4 = Cash withdrawal
5 = Account clearing

1
Your account is in the bank: Tutunska banka

Choose from the menu

1 = Bank overview
2 = Amount overview
3 = Cash addition
4 = Cash withdrawal
5 = Account clearing

2
On your account you have 1000 den.

Choose from the menu

1 = Bank overview
2 = Amount overview
3 = Cash addition
4 = Cash withdrawal
5 = Account clearing

3
Input the amount to add:9000

Choose from the menu

1 = Bank overview
2 = Amount overview
3 = Cash addition
4 = Cash withdrawal
5 = Account clearing

2
On your account you have 10000 den.
```

```
Choose from the menu

1 = Bank overview
2 = Amount overview
3 = Cash addition
4 = Cash withdrawal
5 = Account clearing

4
Input the amount to withdraw:3000

Choose from the menu

1 = Bank overview
2 = Amount overview
3 = Cash addition
4 = Cash withdrawal
5 = Account clearing

4
Input the amount to withdraw:10000
There is not sufficient amount on the account.

Choose from the menu

1 = Bank overview
2 = Amount overview
3 = Cash addition
4 = Cash withdrawal
5 = Account clearing

2
On your account you have 7000 den.

Choose from the menu

1 = Bank overview
2 = Amount overview
3 = Cash addition
4 = Cash withdrawal
5 = Account clearing

5
You have withdrawn the entire amount from the account.
C:\>
```

