

1 Class

Class template:

```
public class Klasa{  
    // fields  
    int a;  
    boolean b;  
    //constructor  
    public Klasa(){}  
    //methods  
    public void metoda1(){}  
    public int metoda2{}  
    //metoda main  
    public static void main(String args) {}  
}
```

public class must be stored in a file with the same name as the class name!

2 Naming conventions

- the next word in a name starts with uppercase
- class name starts with uppercase
- fields and methods name starts with lowercase
- constructor name must be equal to class name

3 Basic datatypes

type	name	wrapping class
logic value	boolean	Boolean
char	char	Character
8 bits digit	byte	Byte
16 bits digit	short	Short
32 bits digit	int	Integer
64 bits digit	long	Long
32 bits floating point digit	float	Float
64 bits floating point digit	double	Double

4 Methods

Method is a part of a class that performs some operation usually on a object fields

```
int nazwaMetody(int parametr1, boolean parametr 2){  
    return 0;  
}
```

Before the name of the methods we insert return type. If method do not return anything we use *void*.

4.1 Access methods

There should be methods to set and get the values of the fields.

```
int pole;  
  
int getPole(){  
    return pole;  
}  
  
void setPole(int pole){  
    this.pole = pole;  
}
```

5 Comments

5.1 Single line comment

```
//This comment will end with the new line
```

5.2 Multi-line comment

```
/*  
Comment  
*/
```

5.3 javadoc comment

```
/**  
javadoc comment  
*/
```

- @param – the description of method parameter
- @return – the description of method return value

6 Tasks to complete

1. Make familiar with the java documentation regarding following classes: Boolean, Character, Byte, Short, Integer, Long, Float, Double
2. Use methods from the those classes
3. Make familiar with java documentation regarding Scanner class
4. Propose the fields and methods for classes: Car, Plane, Phone
5. Create the classes from previous task using proper name conventions
6. Create and generate javadoc for all classes from previous task
7. Create program that sets and prints the values from objects of created classes. The access to fields should be only performed by access methods
8. Create program that gets the input from the user and sets the values of created classes
9. Create the constraints on the fields and apply them to the setter methods.
10. By using Date class create solution to determine the date of last modification of an object
11. Create method that resets all values of fields
12. Create mechanism that count the number of updates in the objects
13. Create mechanism that count the number of objects created from class