

## 1 Klasa

Szkielet klasy:

```
public class Klasa{  
    //pola klasy  
    int a;  
    //konstruktor  
    boolean b;  
    public Klasa(){}  
    //metody  
    public void metoda1(){}  
    public int metoda2{}  
    //metoda main  
    public static void main(String args) {}  
}
```

publiczna klasa musi zostać zapisana w pliku zgodnym z nazwą klasy!

## 2 Konwencje nazewnicze

- we wszystkich przypadkach kolejne słowa w nazwie rozpoczynają się z dużej litery
- nazwa klasy rozpoczyna się z dużej litery
- pola i nazwy metod rozpoczynają się z małej litery
- konstruktor posiada identyczną nazwę z nazwą klasy

## 3 Podstawowe typy danych

rodzaj typu	nazwa	klasa opakowująca
typ logiczny	boolean	Boolean
znak	char	Character
liczba 8 bitowa	byte	Byte
liczba 16 bitowa	short	Short
liczba 32 bitowa	int	Integer
liczba 64 bitowa	long	Long
liczba zmiennoprzecinkowa 32 bity	float	Float
liczba zmiennoprzecinkowa 64 bity	double	Double

## 4 Metody

Metoda jest fragmentem kodu związanym z klasą i operującym na jej polach

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

Nazwa metody poprzedzana jest typem zwracanym. W przypadku gdy metoda nic nie zwraca *void*.

### 4.1 Metody dostępu do pól

W klasie powinny być dostępne metody ustawiające i zwracające wartości pól.

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

## 5 Komentarze

### 5.1 Komentarz obejmujący jedną linię

```
//Komentarz konczacy sie znakiem nowej linii
```

### 5.2 Komentarz obejmujacy wiele linii

```
/*  
Komentarz  
*/
```

### 5.3 Komentarz javadoc

```
/**  
Komentarz javadoc  
*/
```

- @param – opis parametru metody
- @return – opis wartości zwracanej

## 6 Zadania do wykonania

1. Zapoznać się z dokumentacją klas Boolean, Character, Byte, Short, Integer, Long, Float, Double
2. Przetestować metody klas z poprzedniego zadania
3. Zapoznać się z dokumentacją klasy Scanner
4. Zaproponować strukturę klasy dla wybranego obiektu
5. Stworzyć szkielet klasy z zadania 4 z wykorzystaniem właściwych konwencji nazewnictwa
6. Stworzyć i wygenerować dokumentację java doc dla wszystkich pól i metod z poprzedniego zadania
7. Stworzyć program ustawiający a następnie wyświetlający zawartość wszystkich pól danej klasy. UWAGA: Należy pamiętać, że dostęp do pól realizowany powinien być tylko z metod dostępowych
8. Stworzyć program pobierający od użytkownika wartości pól zaproponowanej klasy
9. Ustalić ograniczenia na pola klasy i zabezpieczyć program w taki sposób aby nie można było wpisać wartości spoza zakresu
10. Wykorzystując klasę Date uzupełnić klasę o metodę poznania ostatniej modyfikacji obiektu
11. Stworzyć metodę resetującą wszystkie pola klasy do wartości domyślnych
12. Stworzyć mechanizm liczący liczbę wprowadzonych zmian w obiektach klasy
13. Zaproponować mechanizm obliczania liczby utworzonych obiektów danej klasy