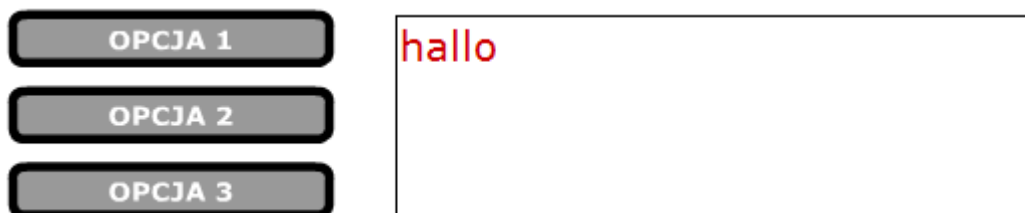


## Ćwiczenie 3 – Funkcje parametryczne

Wykonamy ćwiczenie, które pokazuje zalety stosowania funkcji parametrycznej. Można takiej funkcji używać wielokrotnie, zmieniając tylko parametr przekazywany do funkcji.

Utworzymy trzy przyciski, których wciśnięcie spowoduje wyświetlenie tekstu w polu tekstowym. Każdy przycisk wyświetli inny tekst. Tekst, który ma być wyświetlony będzie parametrem przekazywanym do funkcji.



Pobierz przykład (<http://jsekulska.kis.p.lodz.pl/studia.htm>).

### Krok 1

#### Tworzenie pliku kontenera

Otwórz nowy plik i nie zmieniaj jego wyjściowych ustawień.

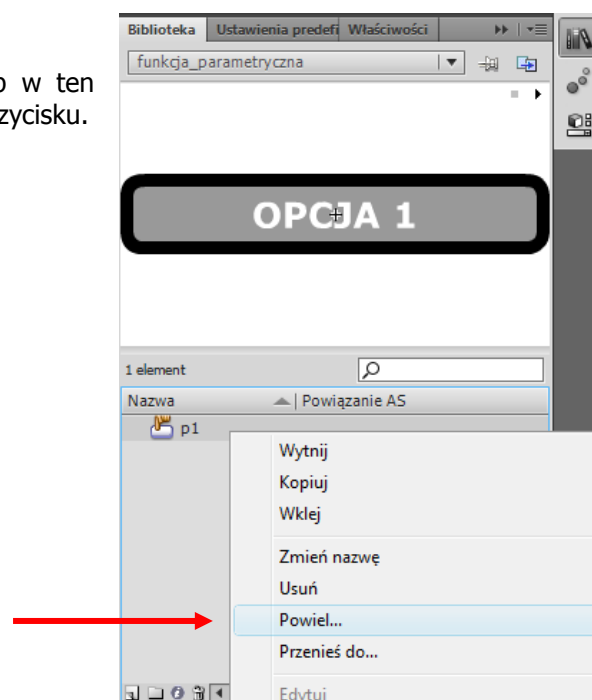
### Krok 2

#### Tworzenie przycisków

Utwórz nowy symbol typu przycisk. Zastosuj własne ustawienia przycisku (kolor tła przycisku, kolor i typ czcionki, zmiany ustawień tych obiektów w stanach przycisku).



Jeśli przycisk jest gotowy powiel go w bibliotece. Zrób w ten sposób dwie kopie. W każdej kopii zmodyfikuj napis na przycisku.



### Krok 3

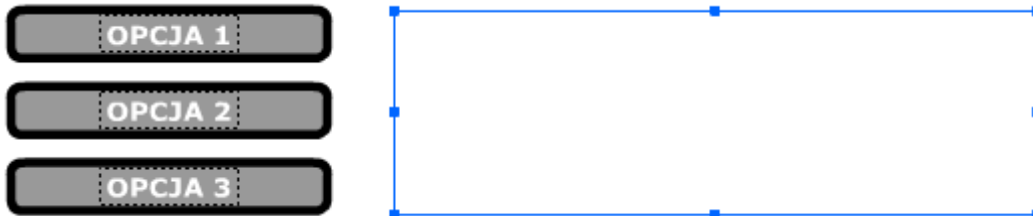
#### Ustawienia przycisków

Gotowe przycisku ustaw na stole montażowym (jak w przykładzie na początku instrukcji).  
Nadaj przyciskom nazwy instancji: **b1\_btn**, **b2\_btn**, **b3\_btn**.

### Krok 4

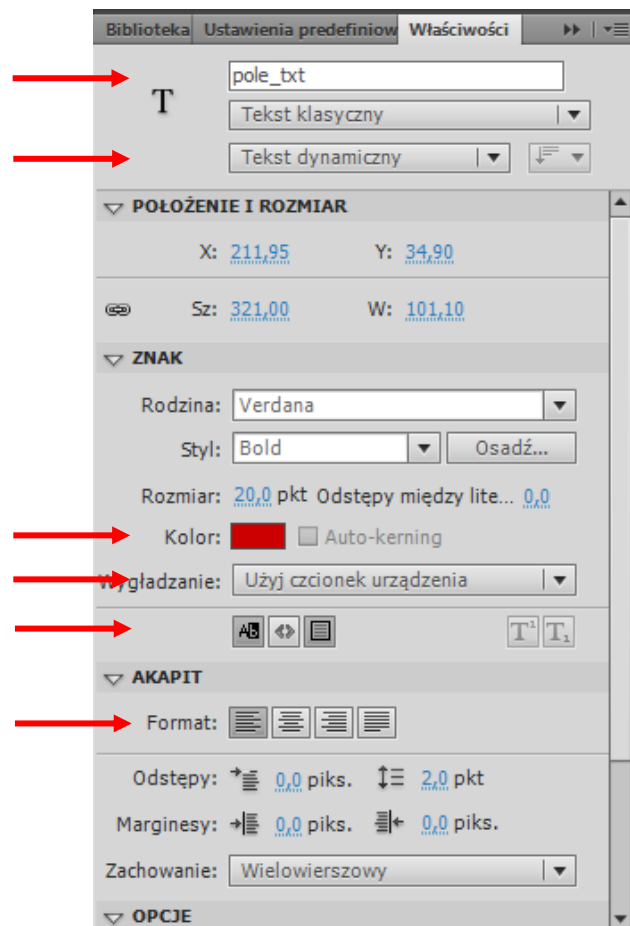
#### Dodanie pola tekstowego

Obok przycisków narysuj narzędziem **Tekst** pole tekstowe.



Ustaw właściwości pola tekstowego:

- nazwa instancji: **pole\_txt**
- tekst dynamiczny
- rozmiar: 20 pkt
- kolor czerwony
- użyj czcionek urzędzenia
- ramka wokół pola tekstowego
- wyrównanie do lewej



### Krok 5

#### Dodanie skryptu

Wstaw nową warstwę na scenie głównej i nazwij ją „akcje”. Będąc w klatce tej warstwy otwórz panel **Operacje**. Zaczynamy pisać kod. Uczynimy to w trzech krokach:

1. Skonstruujemy na początek funkcję parametryczną, która dodaje tekst do pola tekstowego „pole\_txt” ale tekst ten nie jest na razie określony, tylko zastąpiony jest parametrem o nazwie textInserted:

```
function createText(textInserted:String) {
    pole_txt.text=textInserted;
}
```

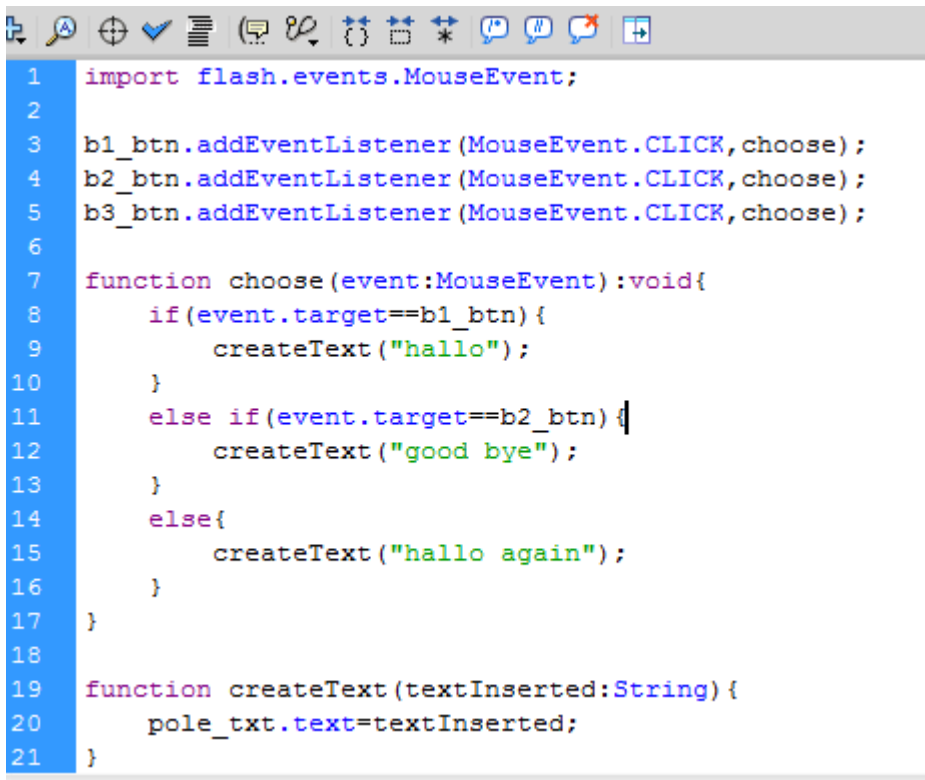
2. Powyżej funkcji **createText** skonstruujemy teraz główną funkcję, która wywoła funkcję **createText** z parametrem zastąpionym właściwym tekstem. Zauważ, że funkcja zostanie wywołana aż trzy razy, przypisując różny tekst w zależności od tego, który przycisk zostanie wciśnięty:

```
function choose(event:MouseEvent):void{
    if(event.target==b1_btn){
        createText("hallo");
    }
    else if(event.target==b2_btn){
        createText("good bye");
    }
    else{
        createText("hallo again");
    }
}
```

3. Musimy jeszcze dodać obiekt nasłuchujący zdarzenia myszy i określić cel zdarzenia, czyli przyciski. Kod piszemy powyżej funkcji **choose**:

```
b1_btn.addEventListener(MouseEvent.CLICK,choose);
b2_btn.addEventListener(MouseEvent.CLICK,choose);
b3_btn.addEventListener(MouseEvent.CLICK,choose);
```

Kod powinien wyglądać teraz następująco:



```
1 import flash.events.MouseEvent;
2
3 b1_btn.addEventListener(MouseEvent.CLICK,choose);
4 b2_btn.addEventListener(MouseEvent.CLICK,choose);
5 b3_btn.addEventListener(MouseEvent.CLICK,choose);
6
7 function choose(event:MouseEvent):void{
8     if(event.target==b1_btn){
9         createText("hallo");
10    }
11    else if(event.target==b2_btn){
12        createText("good bye");
13    }
14    else{
15        createText("hallo again");
16    }
17 }
18
19 function createText(textInserted:String){
20     pole_txt.text=textInserted;
21 }
```

Wszystko jest już gotowe. Przetestuj aplikację.

**Zachowaj plik do kolejnego ćwiczenia.**