# Wielowątkowość

## Wstęp

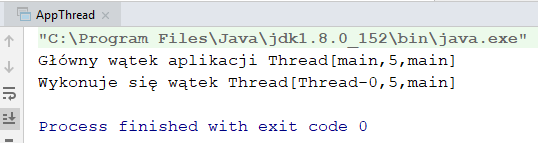
Proces to każdy działający program w pamięci. Proces ma kawałek pamięci dostępny tylko dla siebie. W każdym procesie istnieje co najmniej jeden wątek zwany wątkiem głównym. Z tego wątku możemy tworzyć wątki poboczne. Wszystkie wątki w obrębie jednego procesu współdzielą ze sobą pamięć. Dzięki temu komunikacja między wątkami jest bardzo szybka a to współdzielenie pamięci jest źródłem wszystkich problemów związanym z programowaniem wielowątkowym.

## Sposoby tworzenia wątków

### Rozszerzenie klasy Thread

**public class** MyThread **extends** Thread {  
 @Override  
 **public void** run(){  
 System.***out***.println(**"Wykonuje się wątek "**+MyThread.*currentThread*());  
 }  
}

**public class** AppThread {  
 **public static void** main(String[] args) {  
 System.***out***.println(**"Główny wątek aplikacji**+Thread.*currentThread*());  
 Thread thread=**new** MyThread();  
 thread.start();  
 }  
}



Nazwy main i Thread-0 otrzymamy jeśli wyświetlanie nazwy wątku zapiszemy jako

MyThread.*currentThread*().getName());

#### A jak ustalić własną nazwę dla wątku?

Należy napisać konstruktor i odpowiednio utworzyć obiekt naszego wątku

**public** MyThread(String name) {  
 **super**(name);  
}

Thread thread=**new** MyThread(**"mój wątek"**);

#### Metoda start i run

// częste pytanie na rozmowach kwalifikacyjnych

Metoda **start** uruchamia nowy wątek i uruchamia metodę **run**. Gdybyśmy uruchomili metodę run dla wątku, to nasz kod **nie wykona się w osobnym wątku, ale w wątku głównym aplikacji**.