|  |  |  |
| --- | --- | --- |
| Grupa ćwicz. **2** | Data wykonania 12.04.2018 | Nr. Scenariusza  **6** |
| **Temat ćwiczenia:** Platformy agentowe w języku JADE. | | |
| Imię i nazwisko  **Kamil Szczurkowski** | | Ocena i Uwagi |

# Zadania do wykonania:

1. Utwórz pliki compiljade.bat oraz runjade.bat
2. Uruchomić platformę JADE na komputerze lokalnym
3. Skompiluj klasę HelloAgend.java
4. Dokonać modyfikacji pliku HelloAgent.java tak, aby agent się nie usuwał.
5. Do uruchomionej platformy dodaj kontener a następnie:
   1. Uruchom agenta w nowym kontenerze
   2. Przenieś agenta do innego kontenera
   3. Sklonuj agenta
   4. Usuń agenta
6. Zmodyfikuj przykład TimeAgent.java tak, aby agent usuwał się dopiero po 2 minutach.
7. Utwórz kontener wraz z agentem TimeAgent
   1. Sprawdź co się stanie przy próbie kopiowania agenta .
   2. Wykonaj migracje agenta
8. Prześledź sytuacje wyjątkowe – co się dzieje gdy brak jest klasy tworzonego agenta
9. Dokonaj modyfikacji pliku HelloAgent.java tak, aby agent wypisywał dotychczasowy komunikat tyle razy, ile zostanie podane w parametrze agenta.
10. Stwórz program tak, aby wszystko co było w metodzie main() zostało wywołane w klasie agent.

# Wykonanie zadania:

1. Utworzyłem pliki compiljade.bat oraz runjade.bat dzięki którym będę mógł kompilować kod i uruchamiać platformę



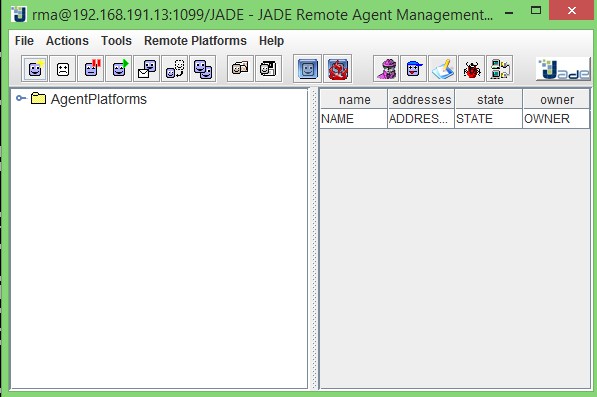
Rysunek 1 - Pliki do kompilacji i uruchamiania.

1. Uruchomiłem platformę w trybie tekstowym



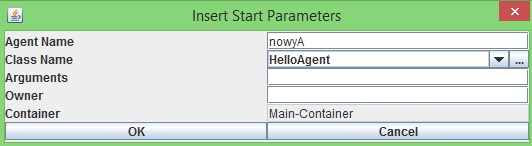
Rysunek 2 Tryb tekstowy.

Następnie uruchomienie w trybie graficznym za pomocą polecenia java jade.Boot –gui

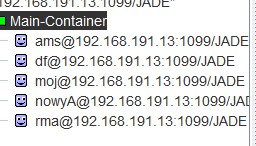


Rysunek 3 - Gui.

1. Skompilowałem i uruchomiłem klasę HelloAgend.java za pomocą naszych plików bat.  
   Następnie w Gui utworzyłem nowego agenta



Rysunek 4 - Utworzenie nowego agenta.



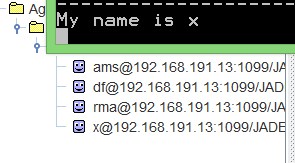
Rysunek 5 - Agenci

Teraz uruchomimy agenta z konsoli za pomocą polecenia java jade.Boot -gui x:HelloAgent



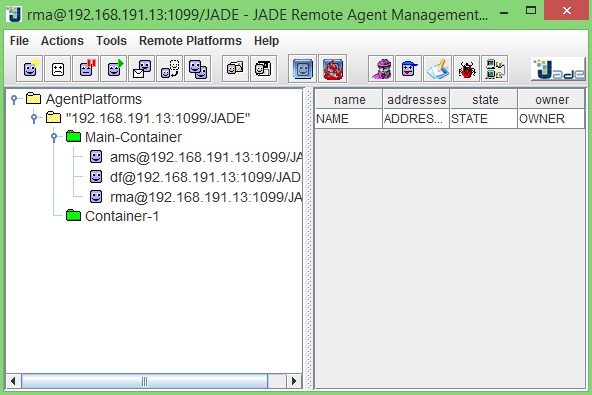
Rysunek 6 – Rezultat

1. Po usunięciu linijki odpowiedzialnej za usuwanie agenta otrzymaliśmy następujący efekt.



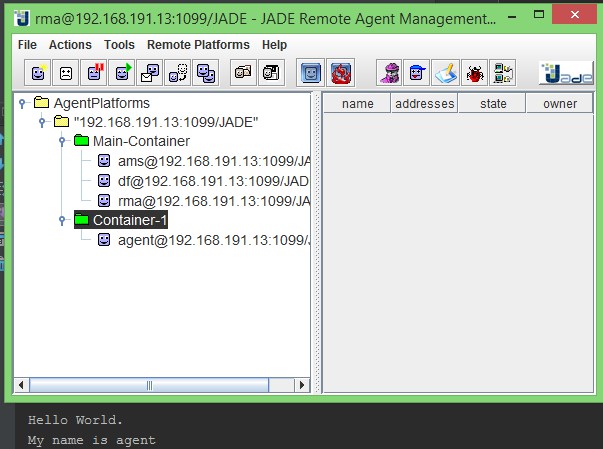
Rysunek 7 - Po uruchomieniu HelloAgent.java

1. Do uruchomionej platformy dodałem kontener:



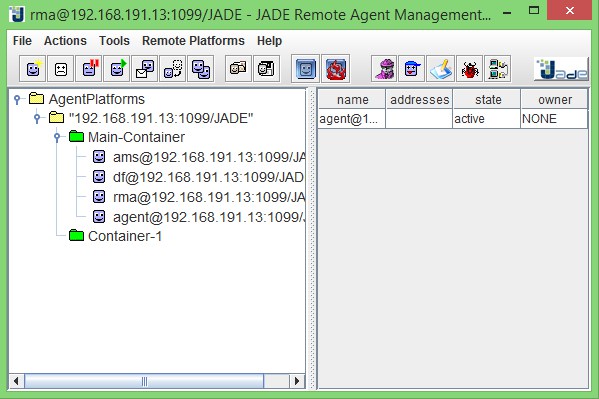
Rysunek - Dodanie drugiego kontenera

1. Utworzyłem agenta w nowym kontenerze



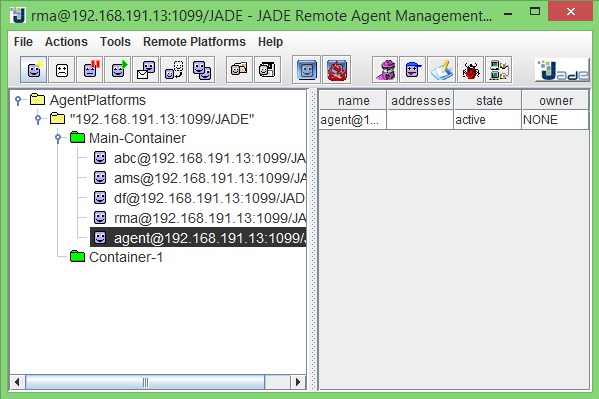
Rysunek - Agent w nowym kontenerze

1. Przeniosłem agenta do innego kontenera



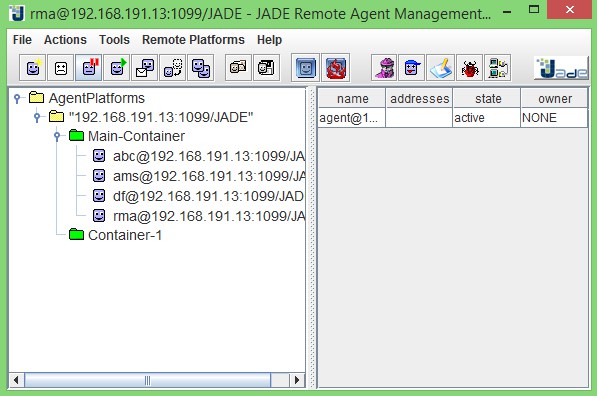
Rysunek - Przenoszenie pomiędzy kontenerami

1. Sklonowałem agenta



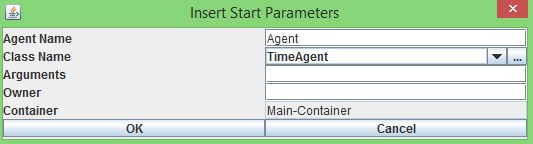
Rysunek - Sklonowanie agenta

1. Usunąłem agenta

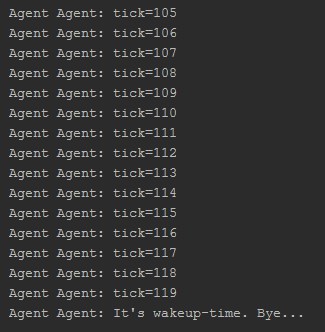


Rysunek - Usuwanie agenta

1. Zmodyfikowałem plik TimeAgent.java tak ze agent usuwa się

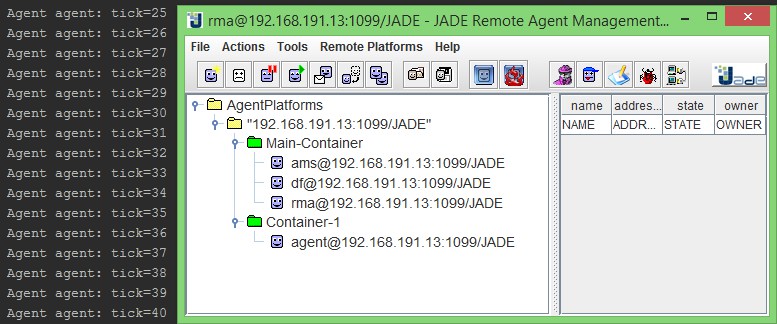


Rysunek 13 - Utworzenie agenta



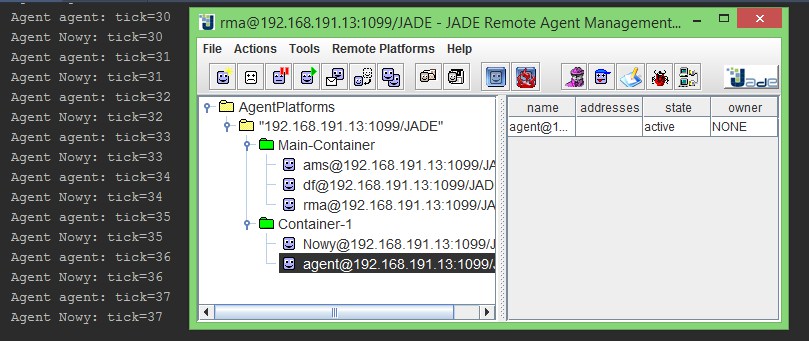
Rysunek 14 - Usuniecie agenta

1. Utworzyłem kontener wraz z agentem



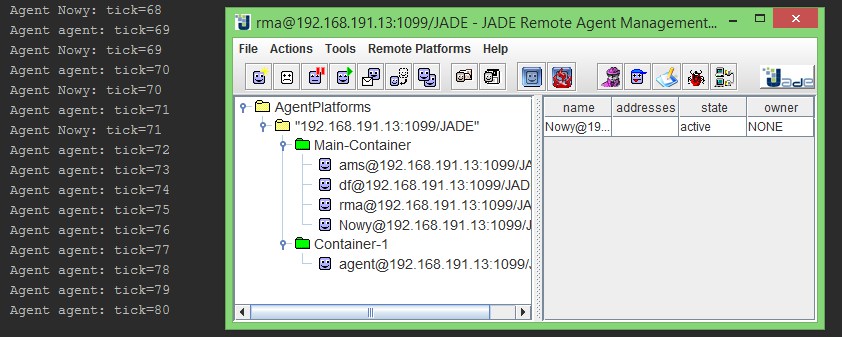
Rysunek 15 - Utworzenie nowego kontenera wraz z agentem

1. Można kopiować agentów podczas pracy. Będą wtedy robić to samo co agent który pracuje. Jeśli agent którego kopiujemy jest na etapie tick = 60 to nowo skopiowany również będzie na tym samym etapie .

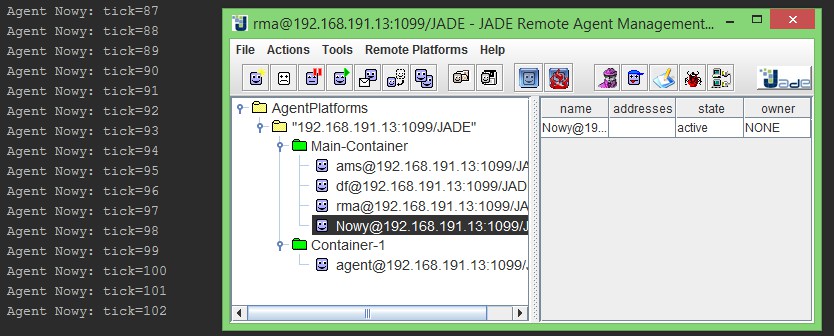


Rysunek 16 - Klonowanie agentów

1. Można również robić migracje agentów pomiędzy rożnymi kontenerami. Wtedy natomiast agenci będą wykonywać swoja prace w osobnych terminalach.

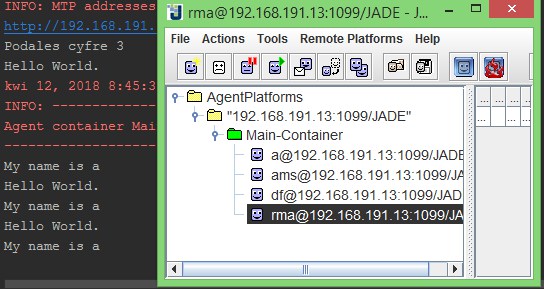


Rysunek 17 - Wynik dla Container 1



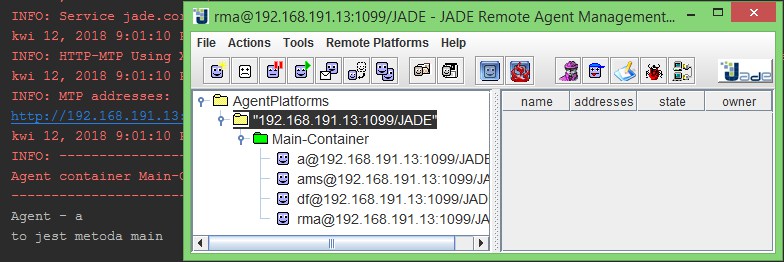
Rysunek 18 - Wynik dla Main - Container

1. Jeśli podczas tworzenia agenta brakuje klasy tworzonego agenta program wyrzuci nam błąd ze dana klasa nie istnieje i agent się nie utworzy podobnie jest z tworzeniem agenta w kontenerze który nie istnieje.
2. Zmodyfikowałem plik HelloAgent tak ze wyświetla się komunikat tyle razy ile wpiszemy w argumencie.



Rysunek 19 - Wynik działania

1. Zmodyfikowałem dotychczasowy program tak ze Agent wywołuje metodę main()



Rysunek 20 - wywołanie przez agenta metody main()