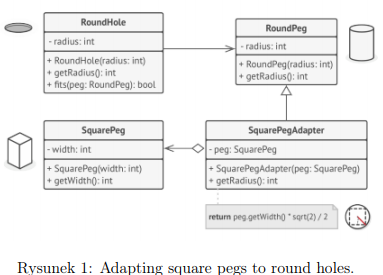
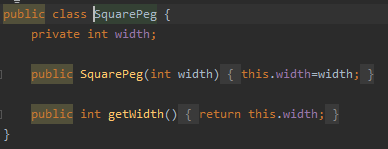
Paweł Kiełbasa, Wojciech Kosztyła

Sprawozdanie – Wzorce Projektowe 2

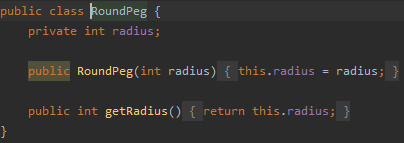
1. Zadanie 1
   1. Zaimplementowaliśmy aplikację według schematu:



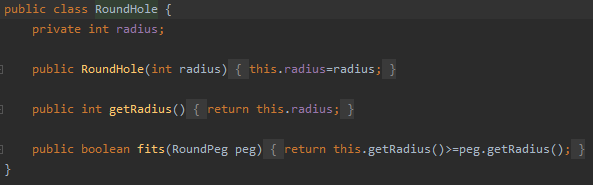
* 1. Klasa SquarePeg



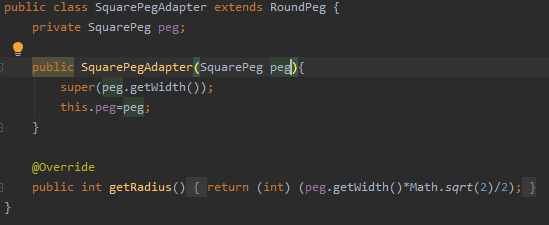
* 1. Klasa RoundPeg



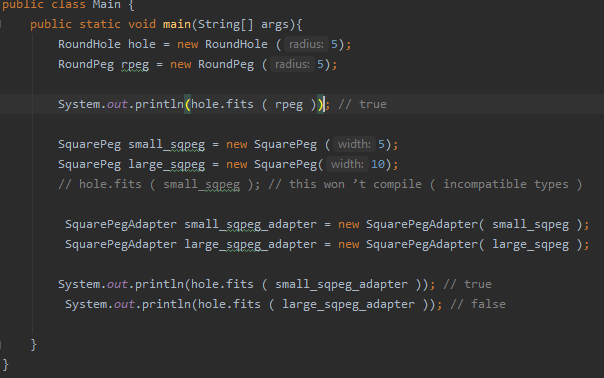
* 1. Klasa RoundHole



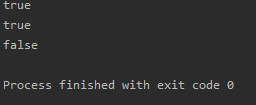
* 1. Klasa SquarePegAdapter



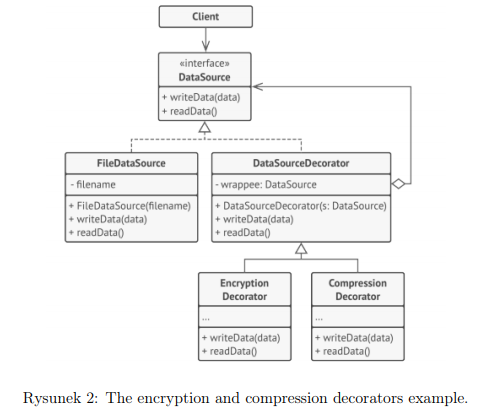
* 1. Klasa Main

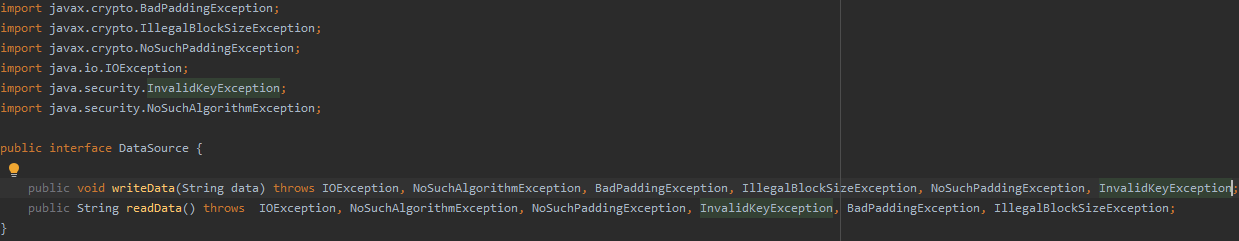
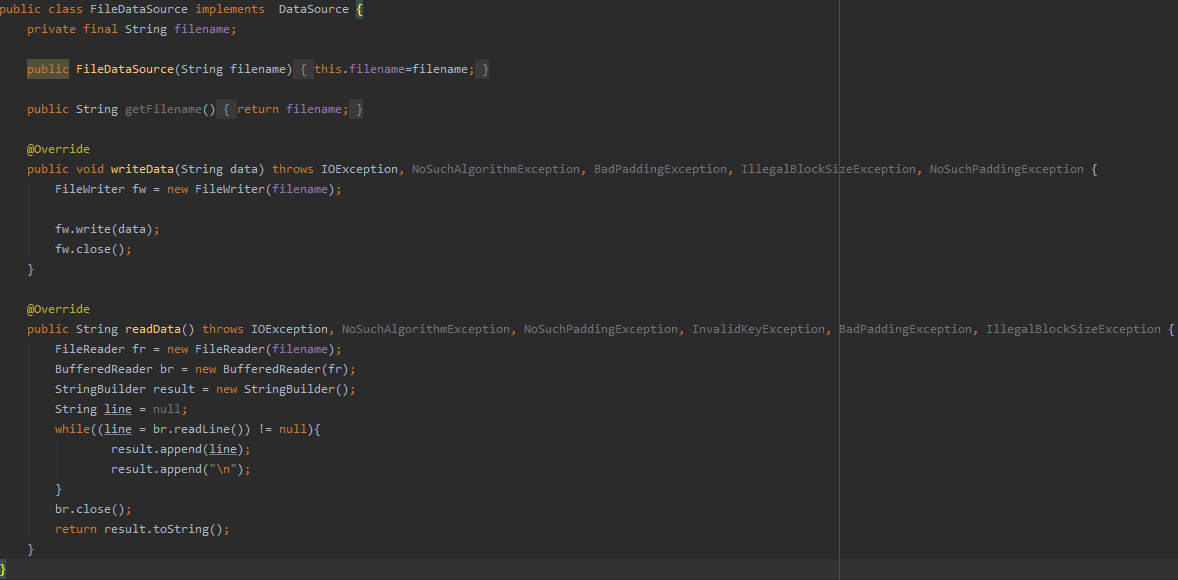
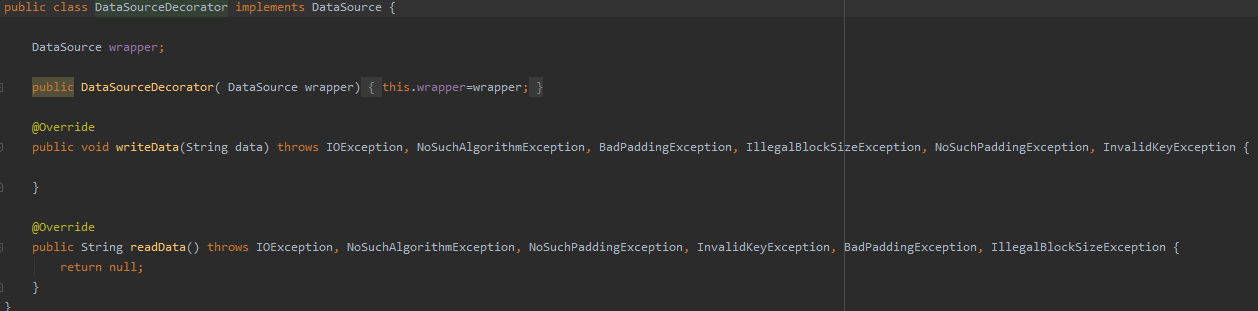
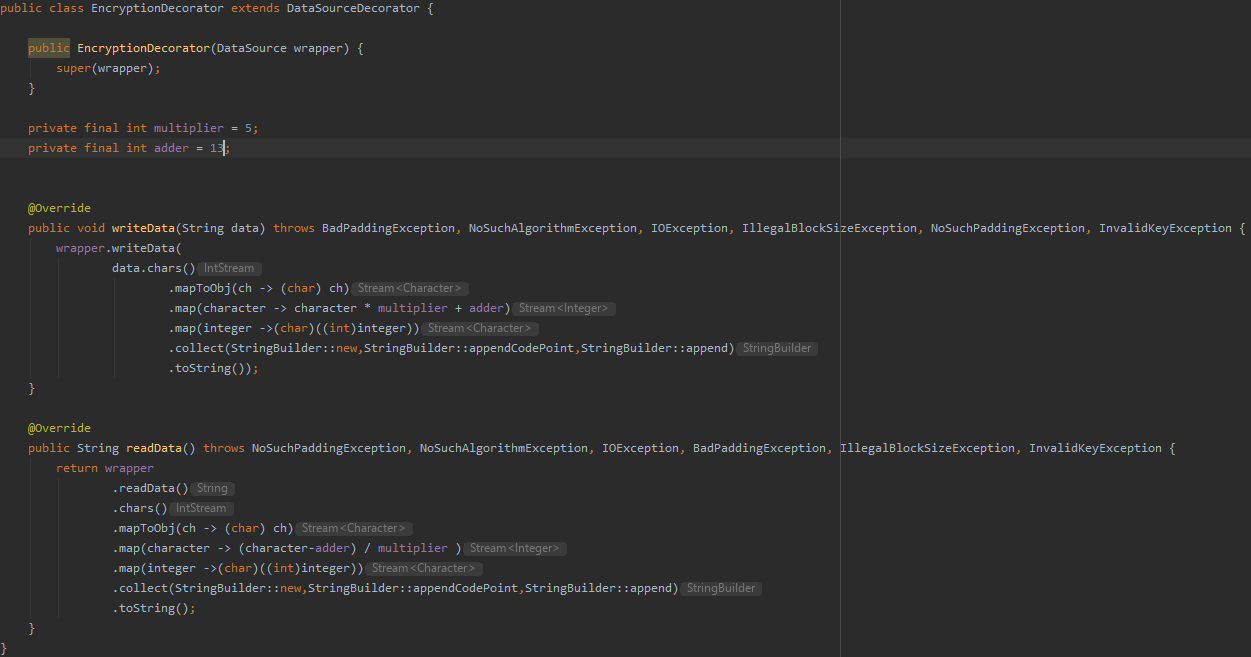
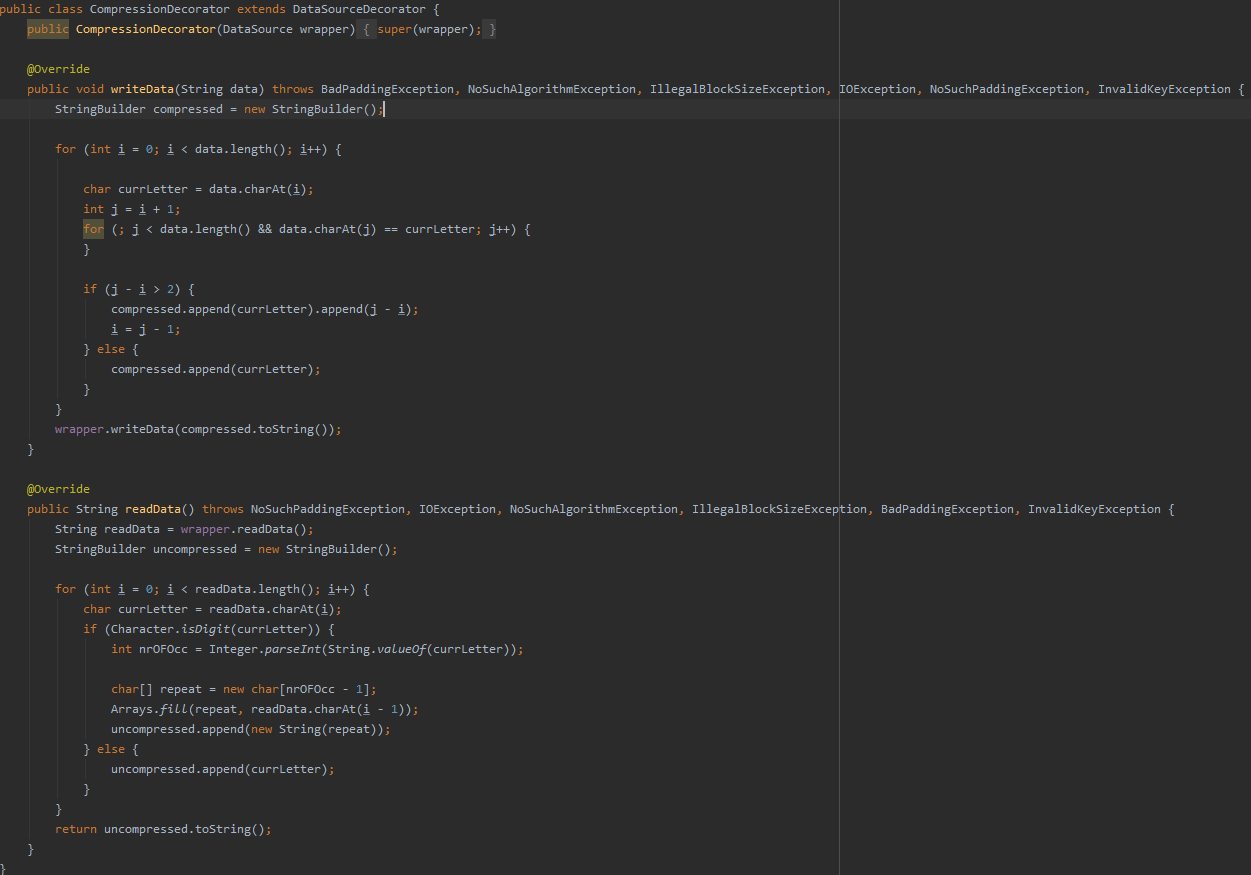


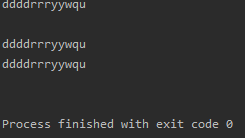
* 1. Efekt wykonania



1. Zadanie 2
   1. Zaimplementowaliśmy aplikację według poniższego schematu



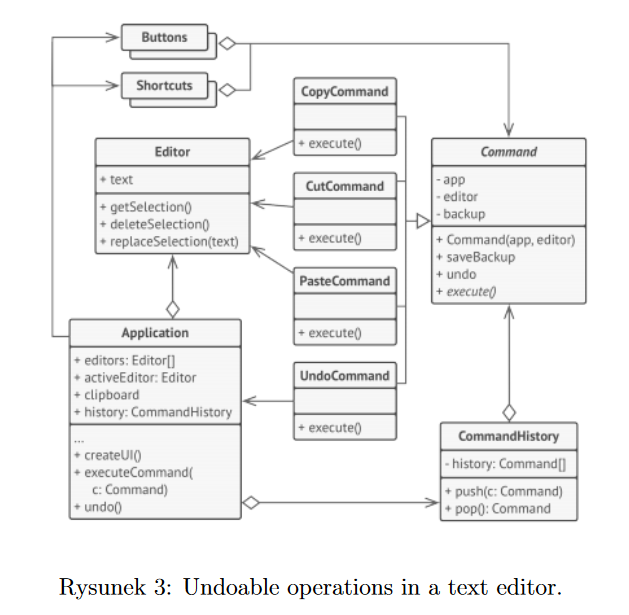
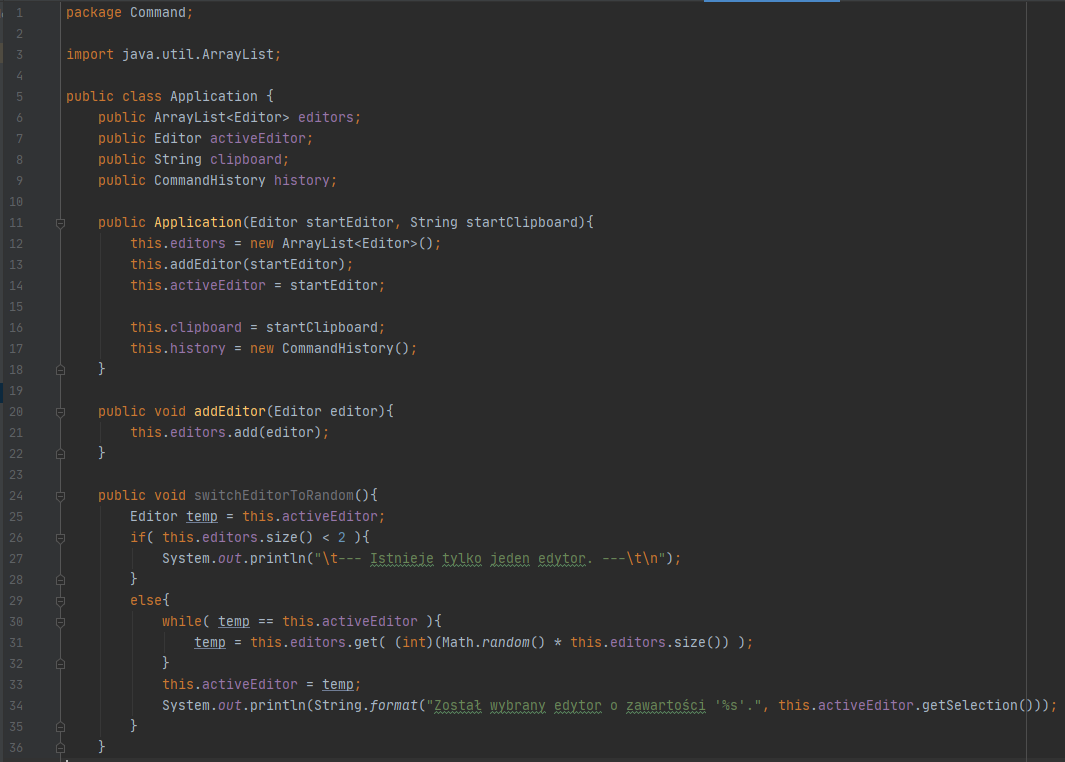
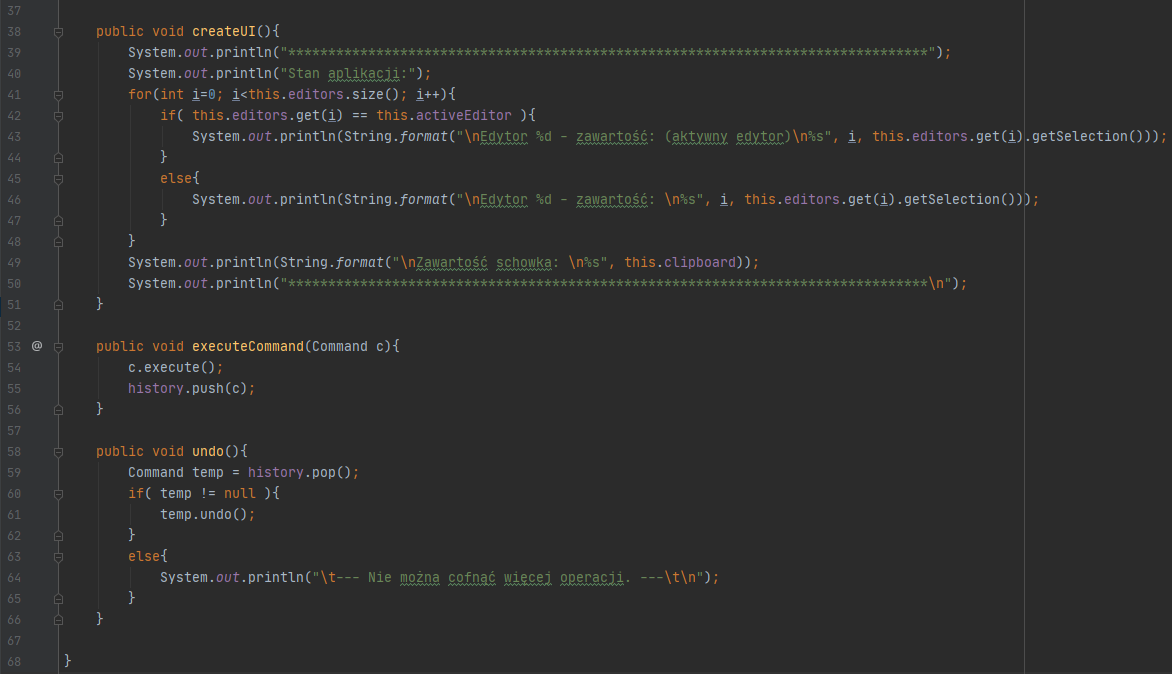
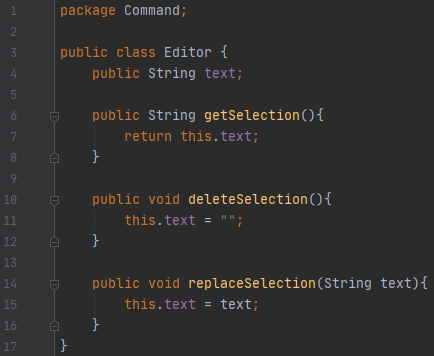
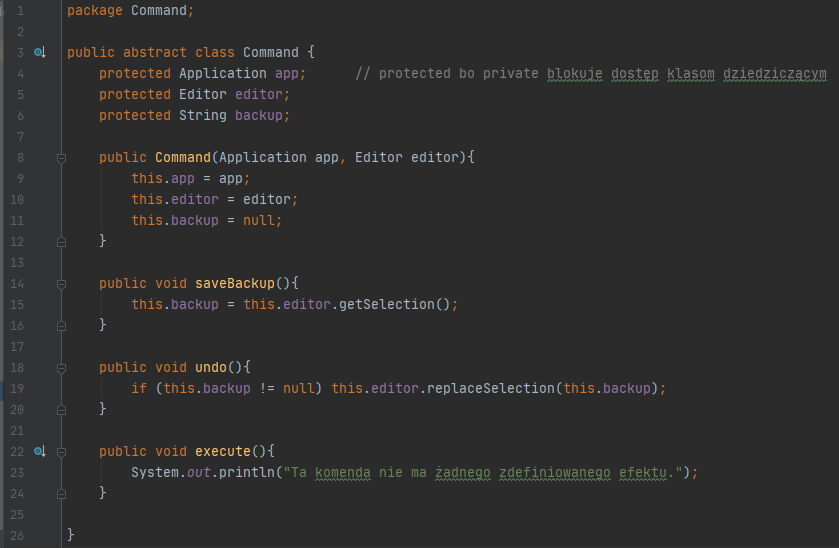
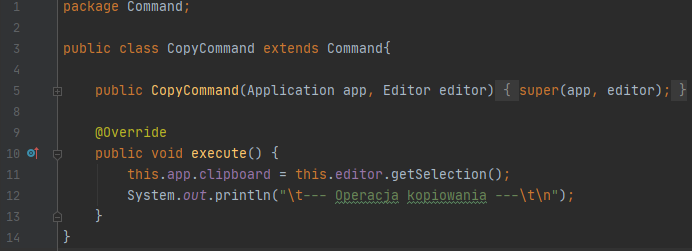
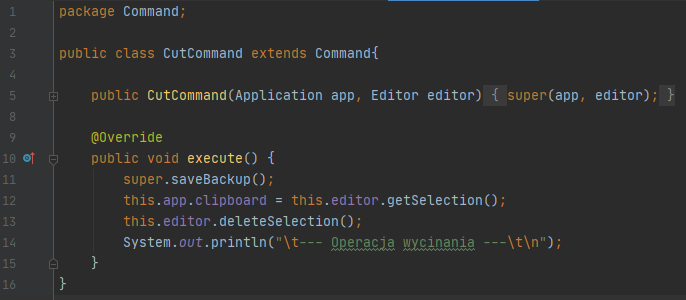
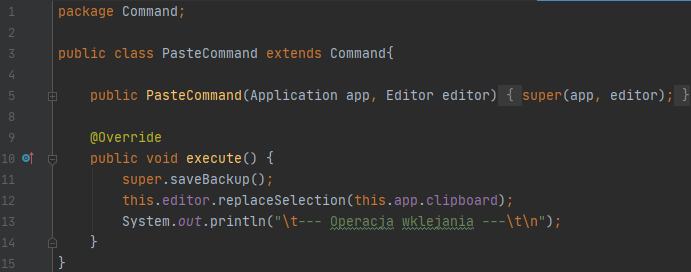
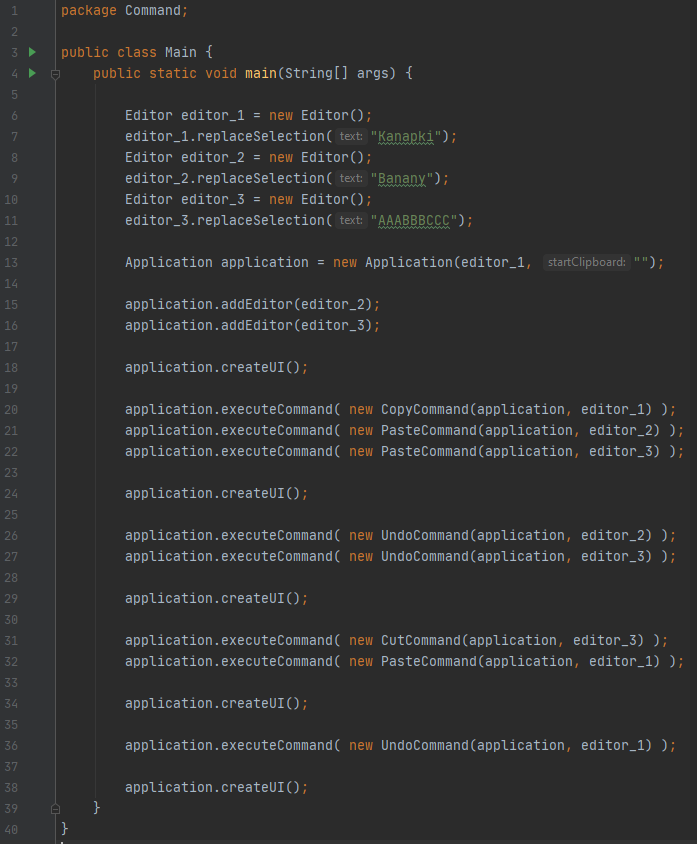
* 1. Interfejs DataSource
  2. Klasa FileDataSource
  3. Klasa DataSourceDecorator
  4. Klasa EncryptionDecorator – ze względu na problemy z gotowymi rozwiązaniami postanowiliśmy stworzyć własny algorytm. Algorytm ten przyjmuje tylko jako argumenty wyrazy nieposiadające żadnej liczby.
  5. Klasa CompressionDecorator
  6. Klasa Main
  7. Efekt wykonania









1. Zadanie 3
   1. Zaimplementowaliśmy aplikację według schematu:  
      
   2. Klasa Application:  
        
      
   3. Klasa Editor:  
      
   4. Klasa Command:  
      
   5. Podklasa CopyCommand:  
      
   6. Podklasa CutCommand:  
      
   7. Podklasa PasteCommand:  
      
   8. Podklasa UndoCommand:  
      
   9. Klasa CommandHistory:  
      
   10. I klasa Main wywołująca mockupowy program.  
       
       1. Efekt wywołania:  
          