SPRAWOZDANIE

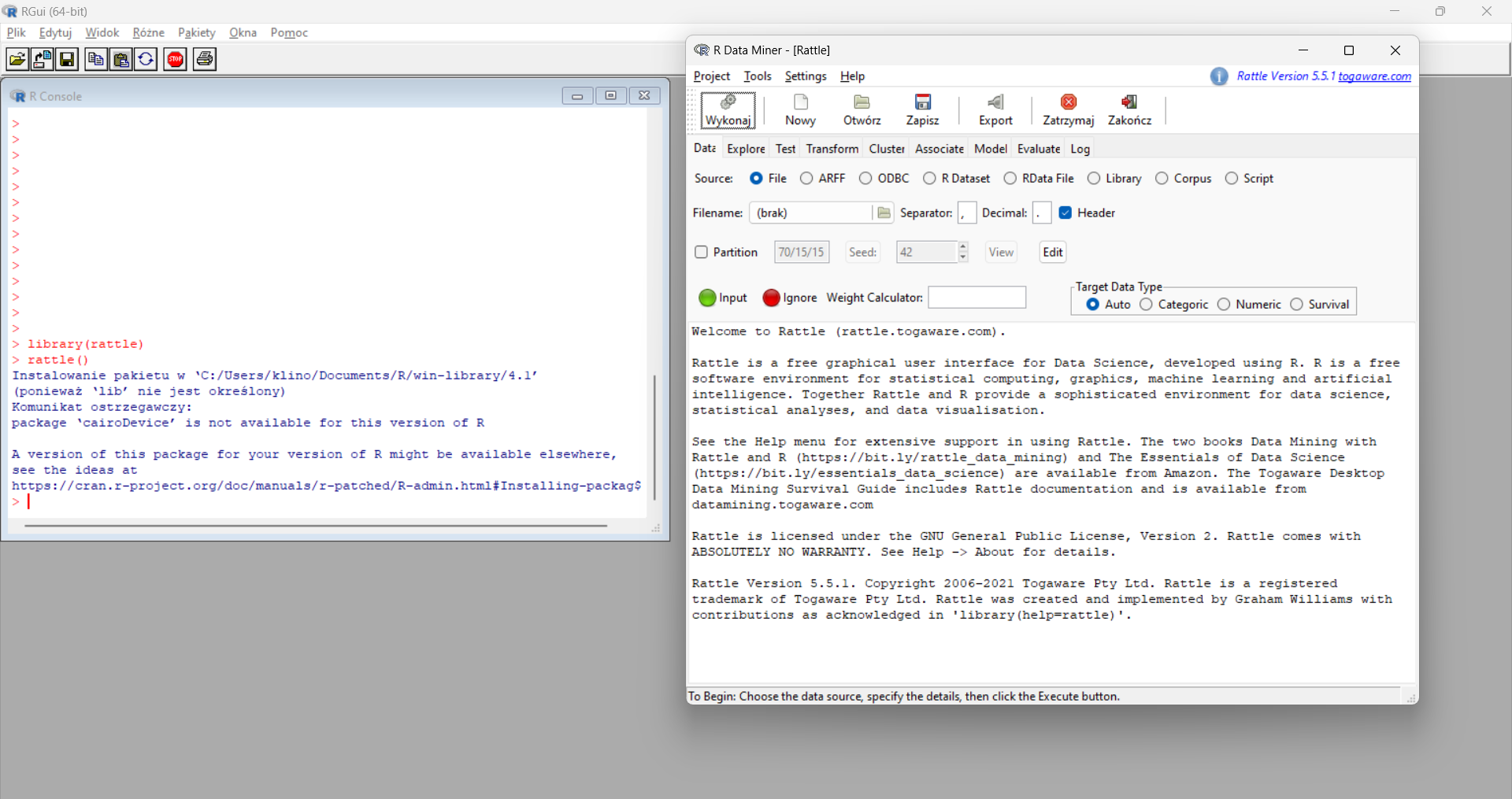
Zajęcia: Zbiory Big Data i Eksploracja Danych

Prowadząca: dr inż. Ruslana Ziubina

|  |  |
| --- | --- |
| Laboratorium nr 1  Data rozpoczęcia: 13.10.2023  Temat: Eksploracja danych – podstawy języka R | Rafał Klinowski  Informatyka  II stopień, stacjonarne,  Semestr 2, gr. a |

# Ćw. 1.

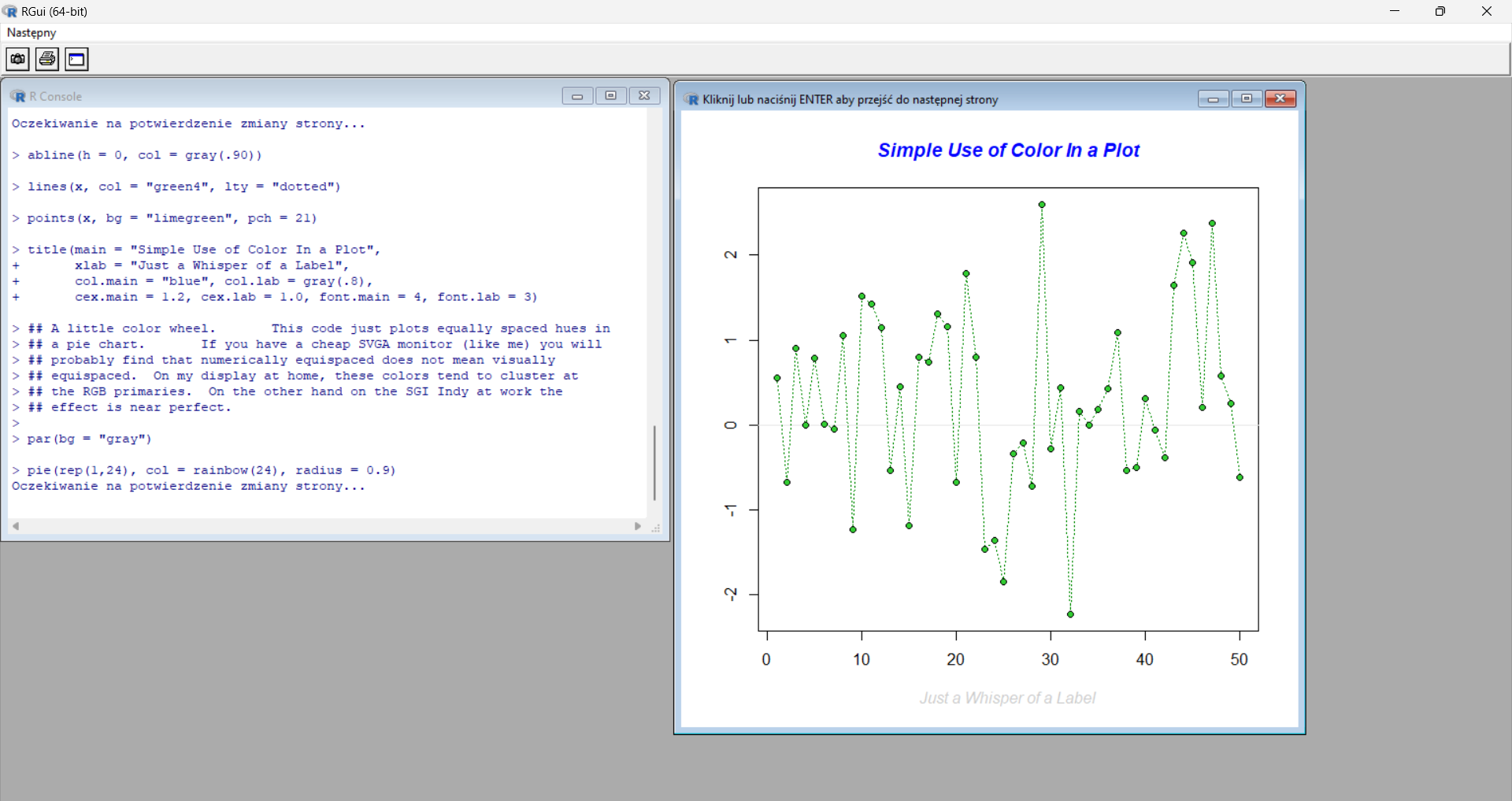
W celu zainstalowania Rattle konieczne było pobranie R w wersji 4.1.3 – w przypadku nowszych wersji występowały problemy z zależnością RGtk2. Po zainstalowaniu Rattle oraz RGtk2 uruchomiono środowisko.



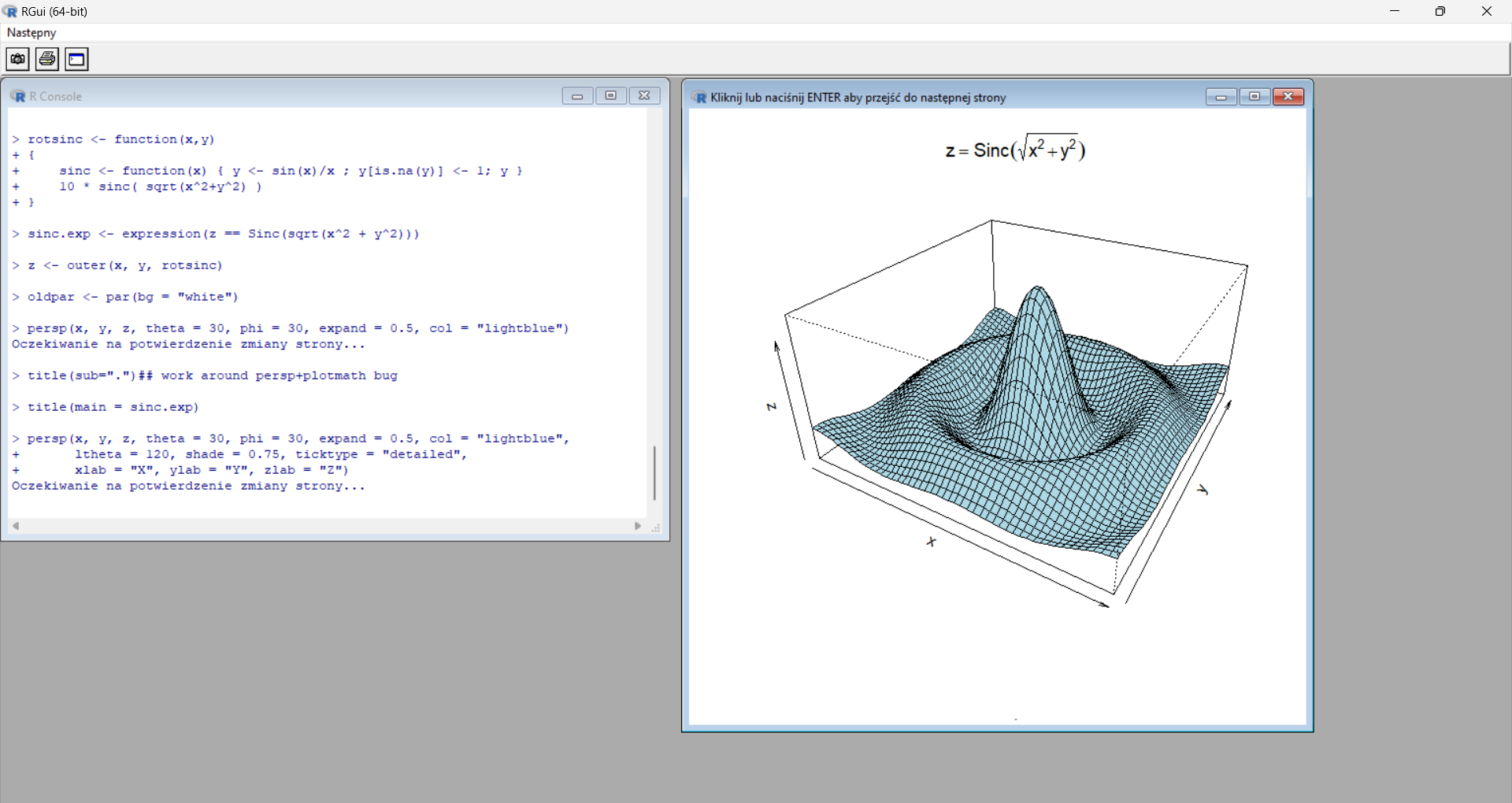
Rysunek . Uruchomione R i Rattle.

# Ćw. 2.

Testowanie niektórych pakietów. W ramach tego ćwiczenia uruchomiono niektóre przykłady z pakietów „graphics” i „persp”.



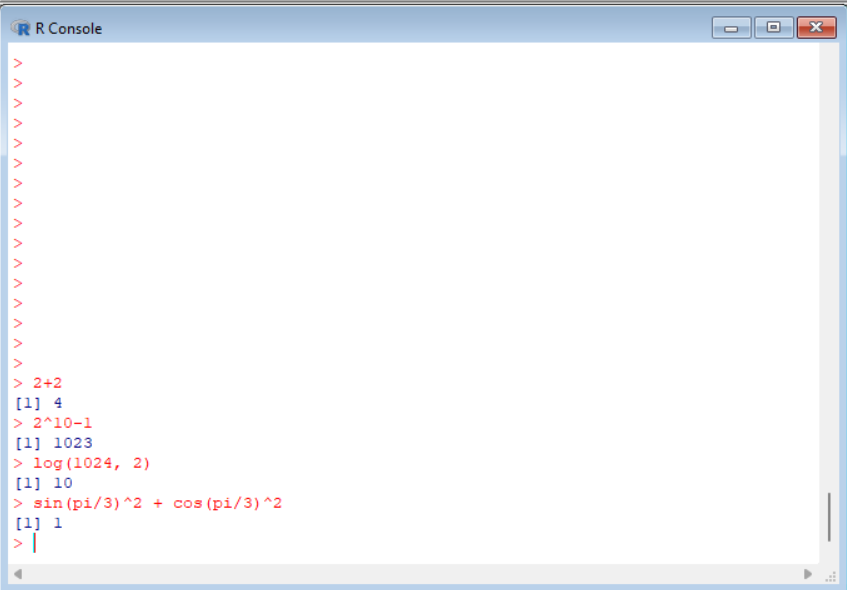
Rysunek . Wynik działania komendy demo(graphics).



Rysunek . Wynik działania komendy demo(persp).

# Ćw. 3.

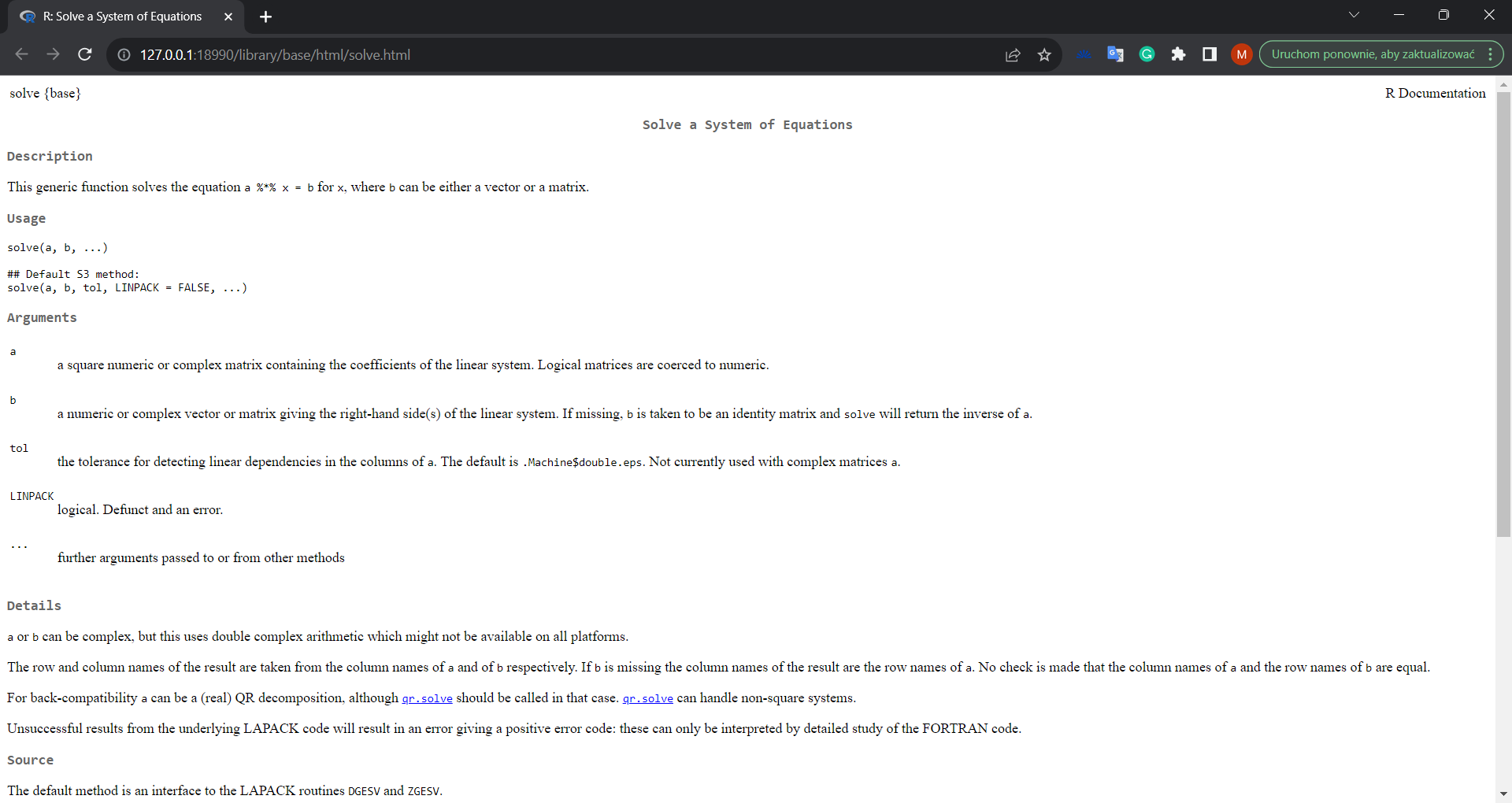
Używanie R jako kalkulatora do wykonywania pewnych operacji matematycznych.



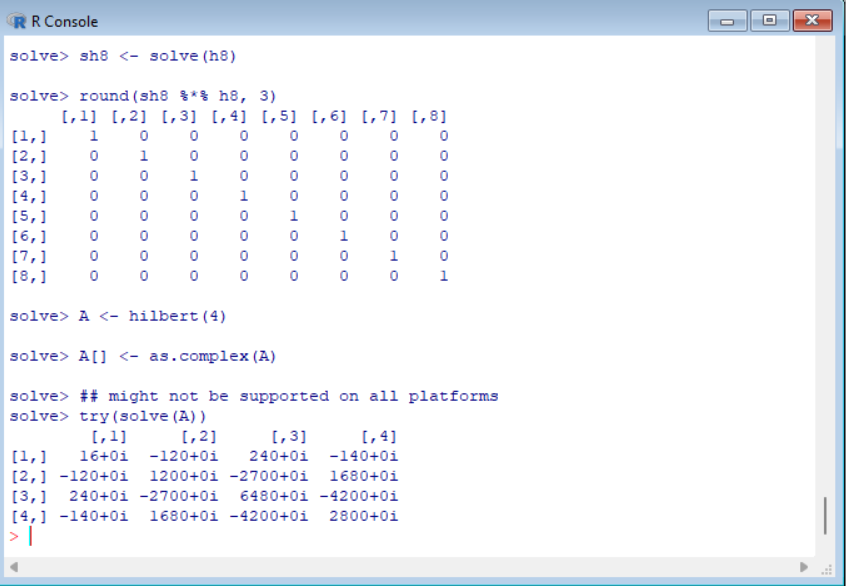
Rysunek . Pewne działania matematyczne w konsoli R.

# Ćw. 4.

Uzyskiwanie pomocy dotyczącej pakietów R – w przeglądarce lub wewnątrz konsoli.



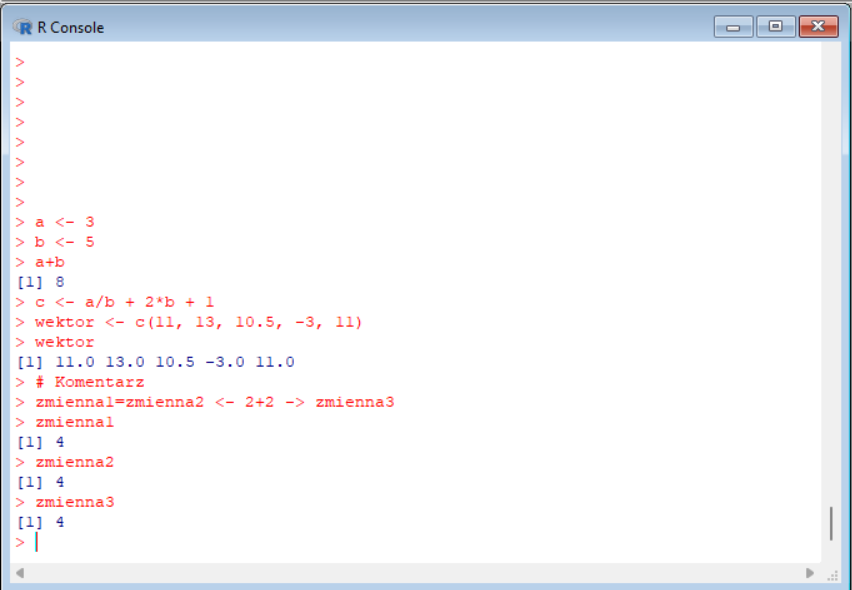
Rysunek . Interfejs pomocy w przeglądarce internetowej po wpisaniu komendy help(solve).



Rysunek . Przykłady użycia biblioteki po wpisaniu komendy example(solve).

# Ćw. 5.

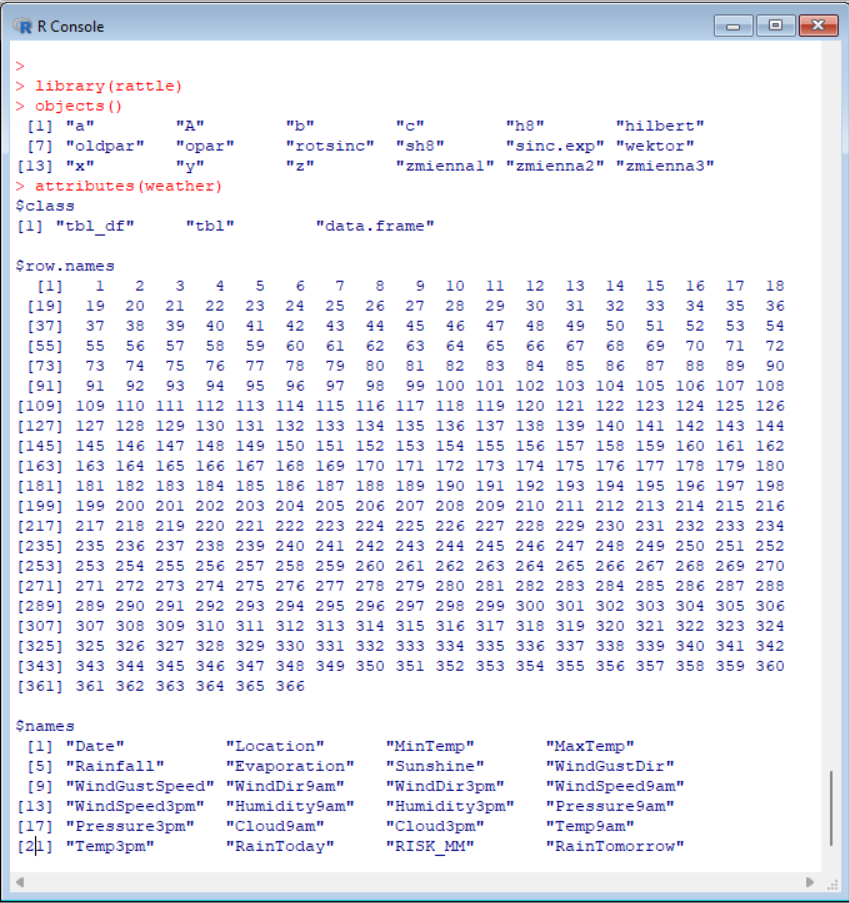
Definiowanie i wykorzystywanie zmiennych oraz komentarze.



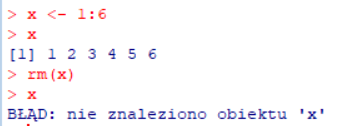
Rysunek . Wynik w konsoli po wykonaniu ciągu instrukcji z instrukcji laboratoryjnej.

# Ćw. 6.

Obiekty w przestrzeni roboczej R oraz atrybuty.



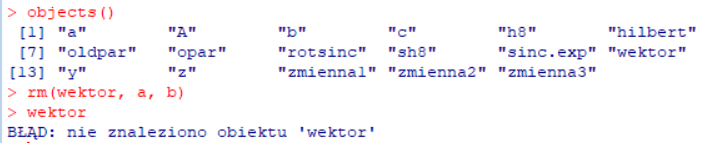
Rysunek . Podgląd obiektów w przestrzeni R oraz atrybutów.



Rysunek . Sposób usunięcia obiektu z pamięci.

# Ćw. 7.

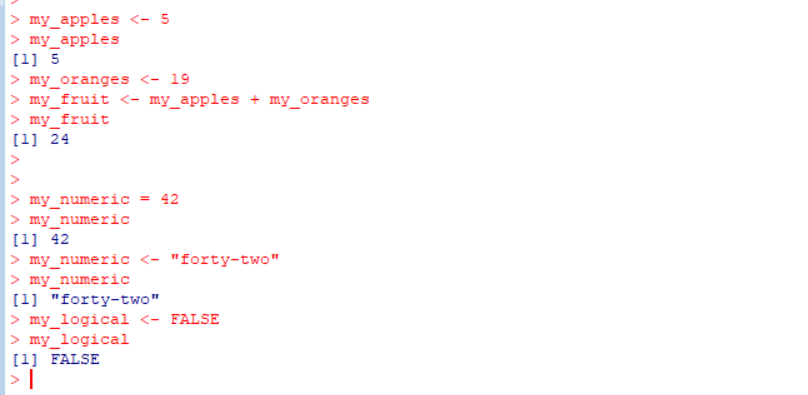
Usuwanie obiektów z pamięci R.



Rysunek . Wyświetlenie oraz usunięcie niektórych obiektów z pamięci. Usunięte obiekty nie mogą być już wykorzystywane (np. wyświetlone).

# Ćw. 8.

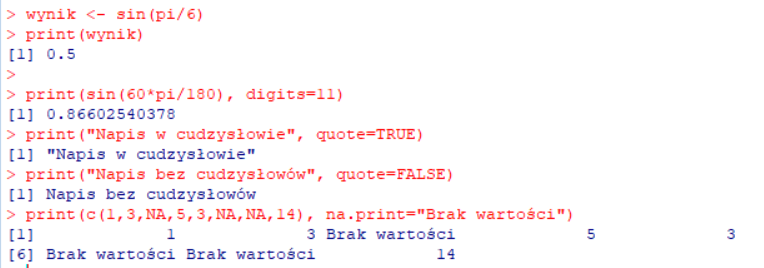
Operator przypisania, typy zmiennych.



Rysunek . Wykonanie pewnych prostych operacji na zmiennych, oraz zapisywanie wartości różnego typu do zmiennych.

# Ćw. 9.

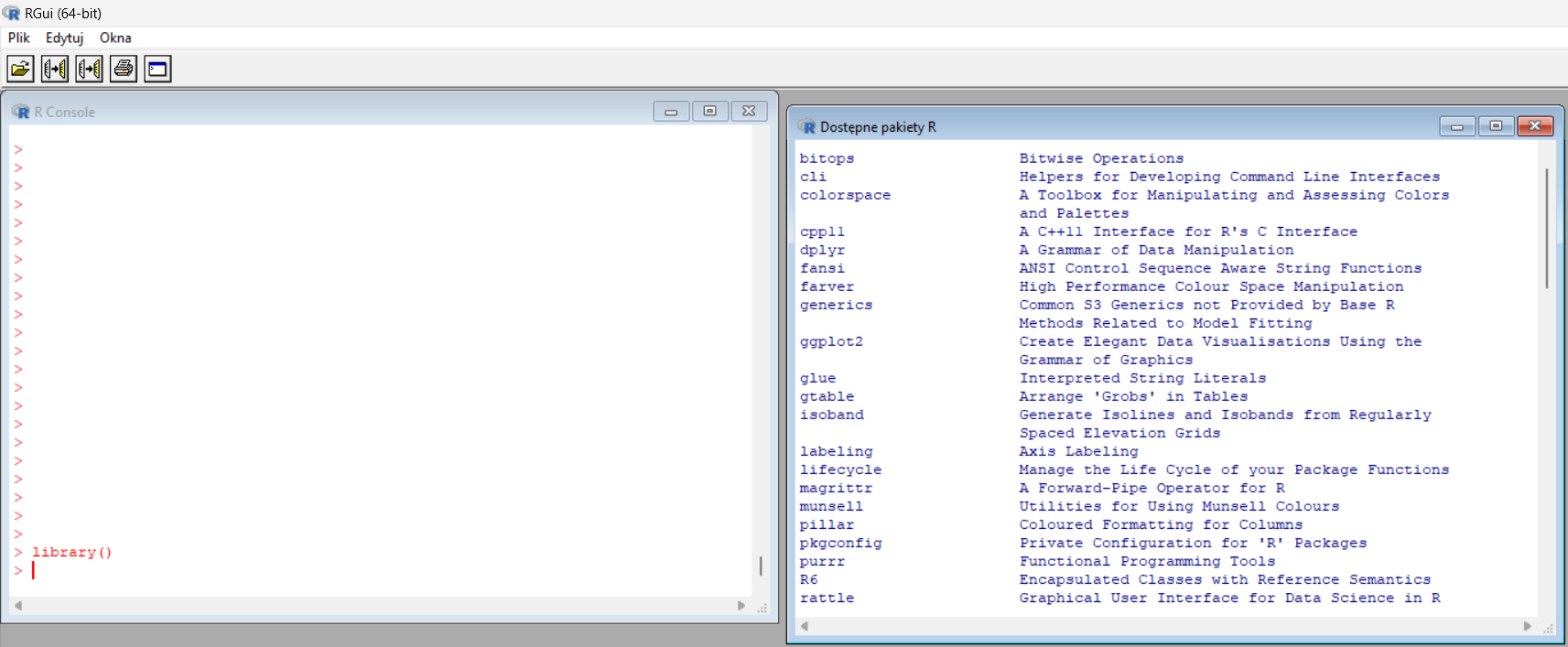
Wykorzystanie funkcji print do wypisywania informacji na ekran.



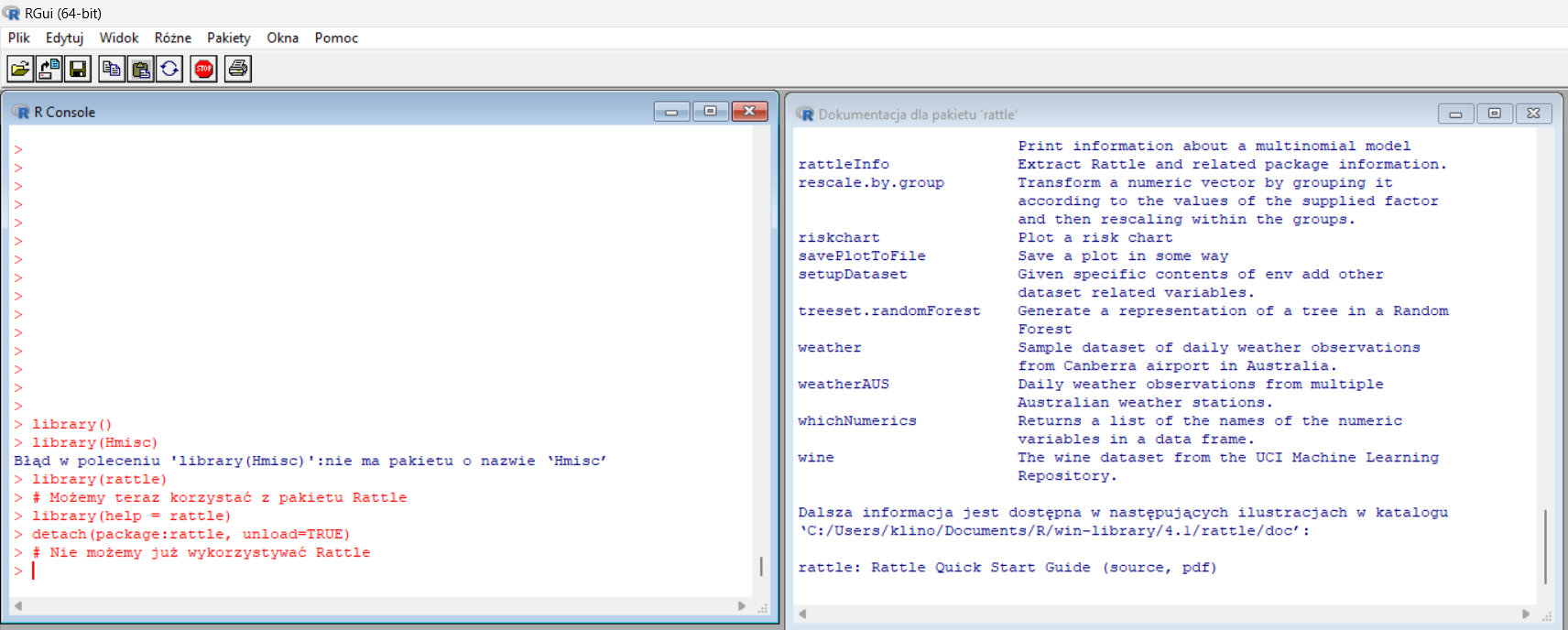
Rysunek . Wyniki wykorzystania funkcji print() w kilku różnych przypadkach i z dodatkowymi argumentami.

# Ćw. 10.

Pakiety w R i ich załadowanie.



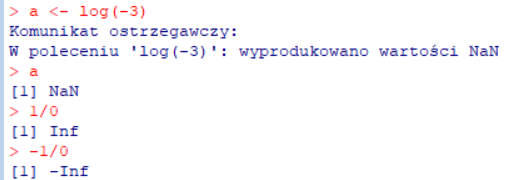
Rysunek . Wynik działania polecenia library().



Rysunek . Ładownie pakietu, uzyskanie pomocy związanej z konkretnym pakietem i usuwanie pakietu z pamięci.

# Ćw. 11.

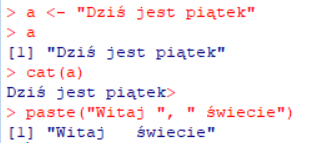
Liczby, NaN, nieskończoności w wyniku niektórych operacji.



Rysunek . Uzyskane wartości NaN oraz +/- nieskończoność przy podaniu nieprawidłowych argumentów do operacji matematycznych.

# Ćw. 12.

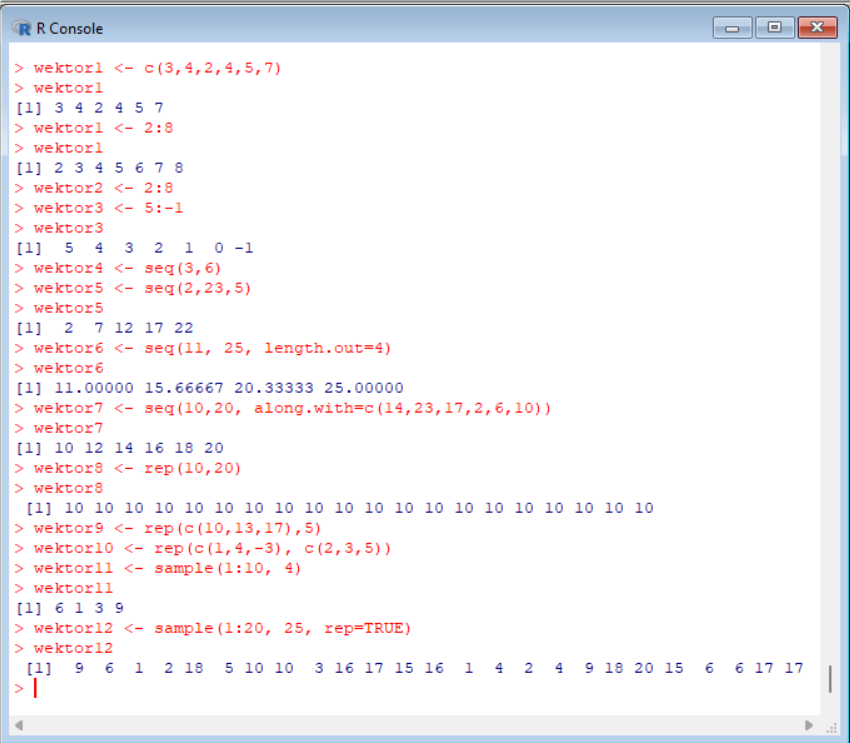
Łańcuchy znaków.



Rysunek . Wypisanie łańcucha znaków z formatowaniem i bez formatowania, skorzystanie z funkcji paste() łączącej łańcuchy.

# Ćw. 13.

Wektory w R.



Rysunek . Sposoby tworzenia wektorów w R – skorzystanie z funkcji: c, seq, rep, sample – wraz z przykładowymi wartościami.

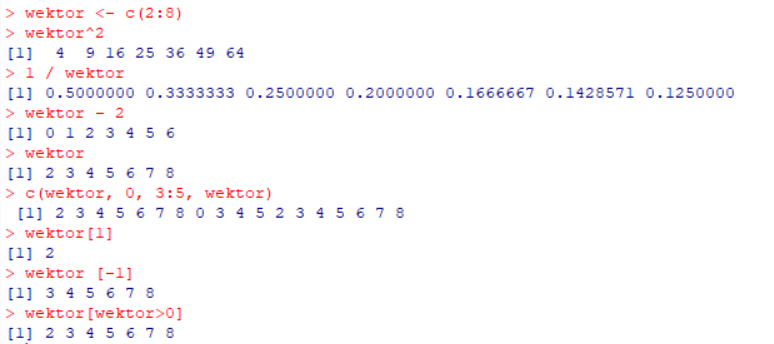
* c() – pozwala tworzyć wektor z podanych wartości, w tym innych wektorów
* seq() – pozwala tworzyć wektor z liczb w danym przedziale od-do, opcjonalnie z danym krokiem
* rep() – pozwala tworzyć wektor z danej ilości powtórzeń danego elementu
* sample() – pozwala tworzyć wektor liczb losowych z danego przedziału, z powtórzeniami lub bez



Rysunek . Uzyskanie długości wektora.



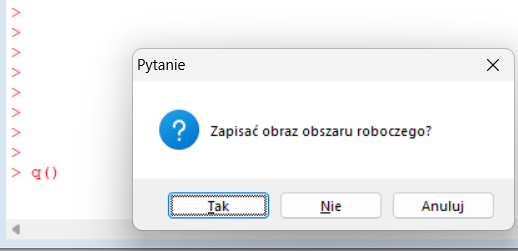
Rysunek . Utworzenie wektorów liczb, znaków oraz wartości logicznych. Typów nie można mieszać – w przypadku wektora mixed\_vector, liczba została przekonwertowana na tekst.

=

Rysunek . Operacje na wektorach. Wypisywanie danych elementów wektora.

# Ćw. 14.

Wyjście ze środowiska R – możliwe zapisanie danych z pamięci.



Rysunek . Próba wyjścia z R – pytanie o zapisanie obszaru roboczego (obiektów w pamięci).

# Ćw. 15.

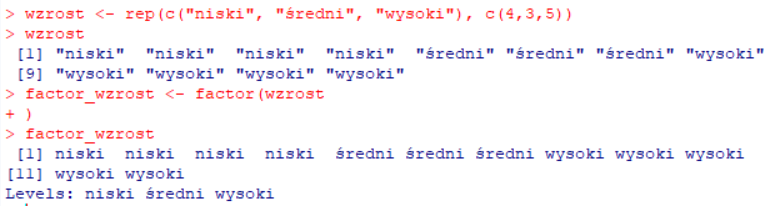
Tworzenie małych zbiorów danych. Zbiory zostały utworzone z wykorzystaniem funkcji c().



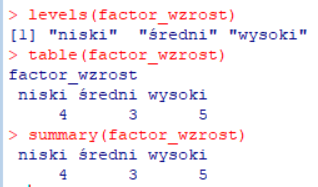
Rysunek . Operacje na małym zbiorze danych – utworzenie macierzy, uzyskiwanie niektórych kolumn lub wierszy, sumowanie elementów, itp.

# Ćw. 16.

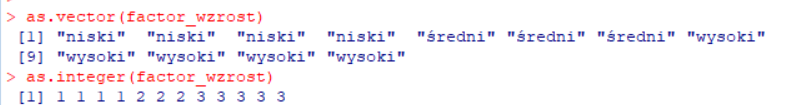
Typ danych „factor” - czynniki.



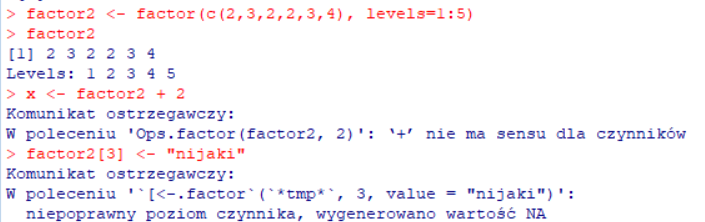
Rysunek . Konwersja wektora na factor.



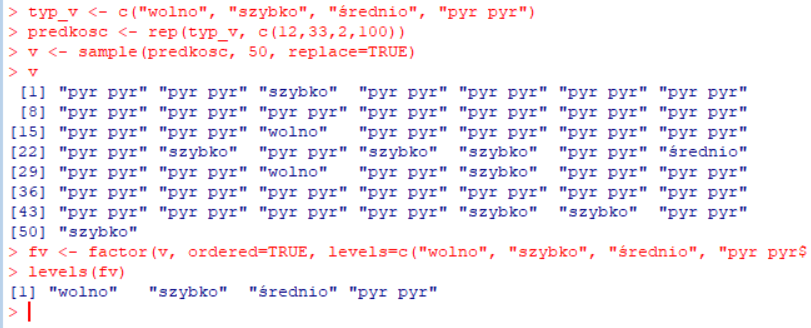
Rysunek . Operacje na typie factor.



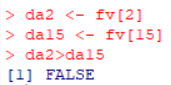
Rysunek . Konwersja odwrotna – z typu factor na wektor lub liczby.



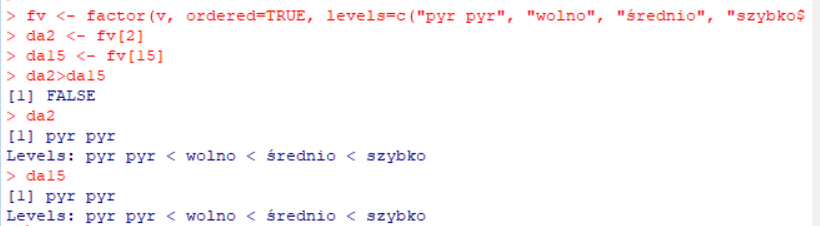
Rysunek . Operacje arytmetyczne czy próba podstawienia pod zmienną kończą się błędem.



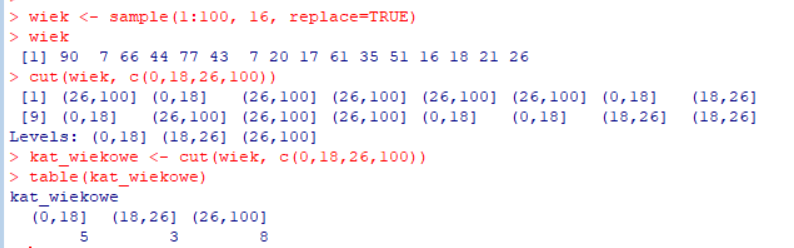
Rysunek . Ustalenie porządku podczas tworzenia factor.



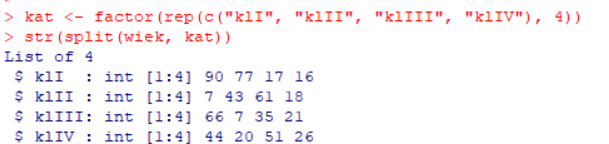
Rysunek . Porównywanie elementów factor zgodnie z kolejnością.



Rysunek . Porównanie po zmianie kolejności.



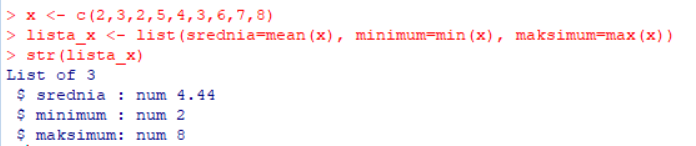
Rysunek . Podział wektora za pomocą funkcji cut().



Rysunek . Podział na grupy z wykorzystaniem wektora i factora.

# Ćw. 17.

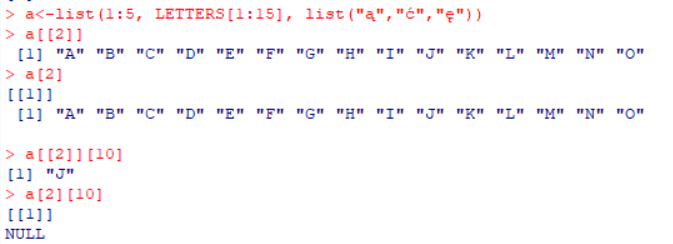
Listy w R.



Rysunek . Utworzenie prostej listy kilku wartości na podstawie wektora.



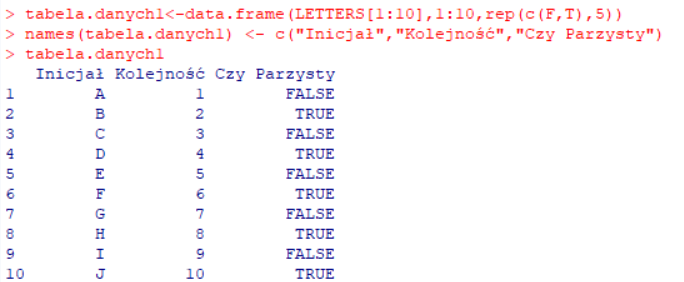
Rysunek . Sposoby odwołania się do pojedynczych elementów listy.



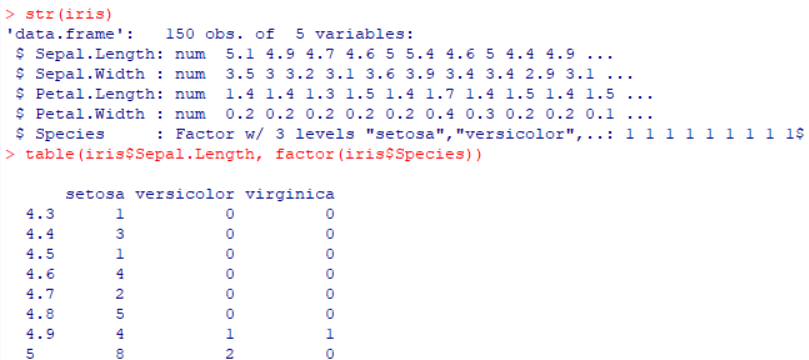
Rysunek . Utworzenie i odwoływanie się do bardziej skomplikowanej listy.

# Ćw. 18.

Ramka danych w R.



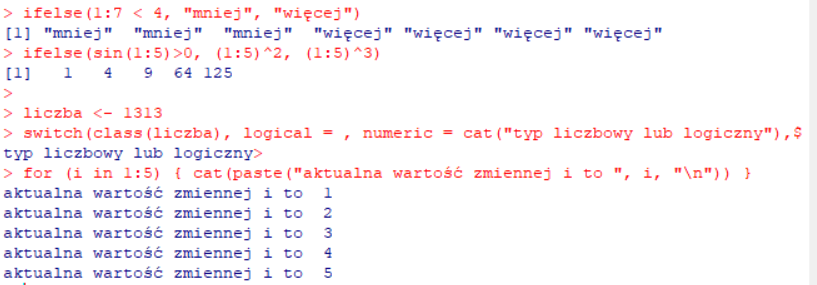
Rysunek . Utworzenie ramki danych i ustawienie nazw kolumn.



Rysunek . Wykorzystanie wbudowanej ramki danych zawierającej zbiór danych „Iris”.

# Ćw. 19.

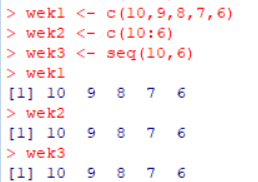
Instrukcje w R.



Rysunek . Wykonywanie podstawowych instrukcji w R – instrukcje warunkowe, pętle.

# Ćw. 20.

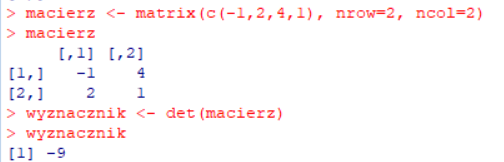
Utwórz na trzy różne sposoby wektor składający się z elementów 10, 9, 8, 7, 6.



Rysunek . Utworzone trzy identyczne wektory na trzy różne sposoby.

# Ćw. 21.

Oblicz wyznacznik macierzy:



Rysunek . Utworzenie macierzy i obliczenie jej wyznacznika.

# Ćw. 22.

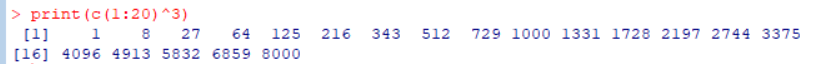
Dokonaj mnożenia macierzy:



Rysunek . Utworzenie dwóch macierzy i ich mnożenie.

# Ćw. 23.

Wyświetl za pomocą jednej instrukcji sześciany liczb od 1 do 20.



Rysunek . Jedna instrukcja użyta do wyświetlenia sześcianów liczb od 1 do 20.

# Ćw. 24.

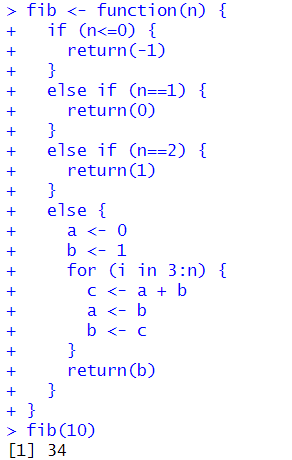
Wyświetl za pomocą jednej instrukcji kwadraty liczb parzystych od 16 do 40.



Rysunek . Jedna instrukcja użyta do wyświetlenia sześcianów liczb parzystych od 16 do 40.

# Ćw. 25.

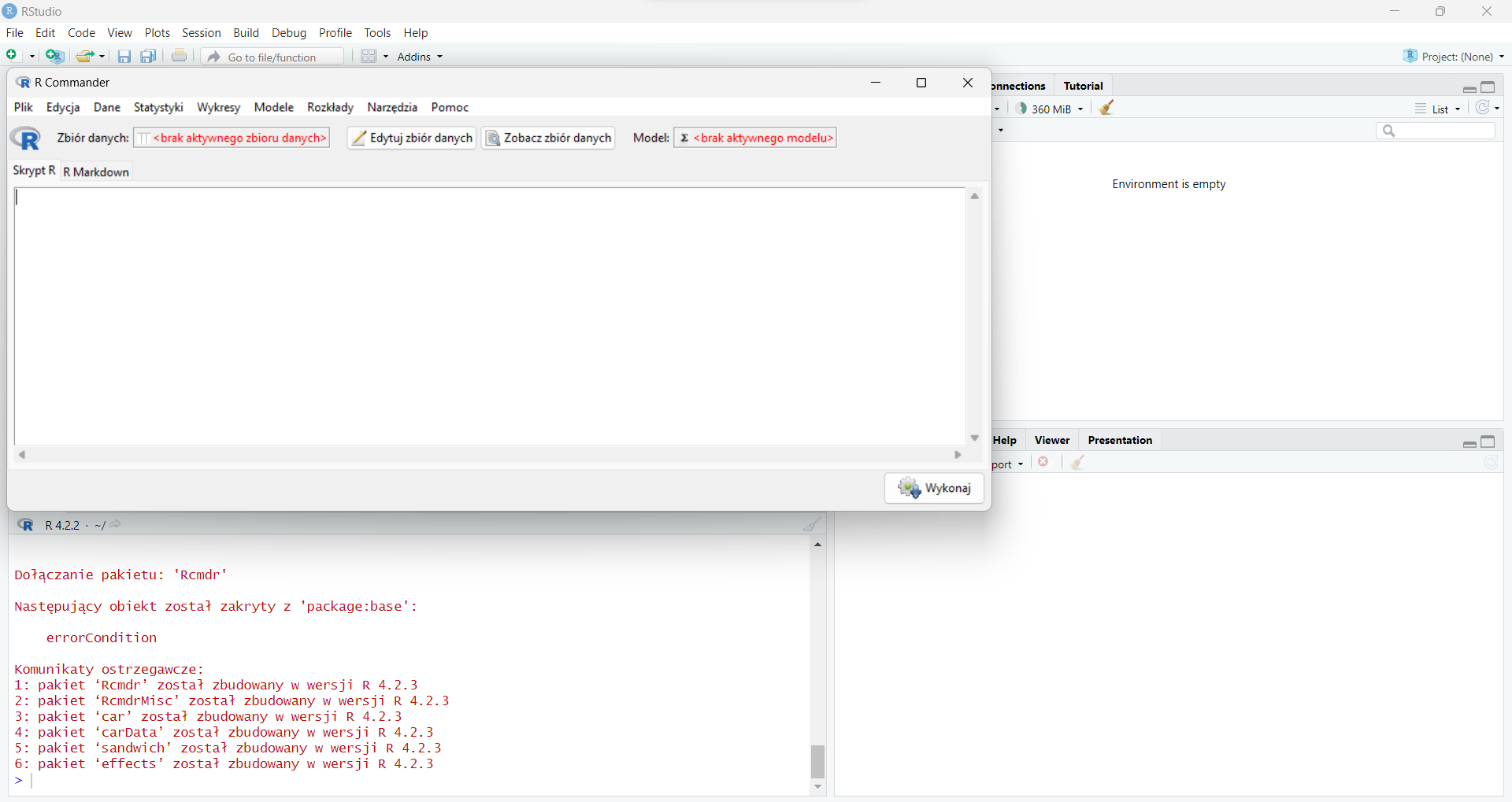
Napisać funkcję liczącą wartość ciągu Fibonacciego dla podanego n.



Rysunek . Stworzona funkcja i przykład jej użycia dla n=10. Funkcja została zapisana w osobnym pliku źródłowym w celu prostszego użycia.

# Ćw. 26.

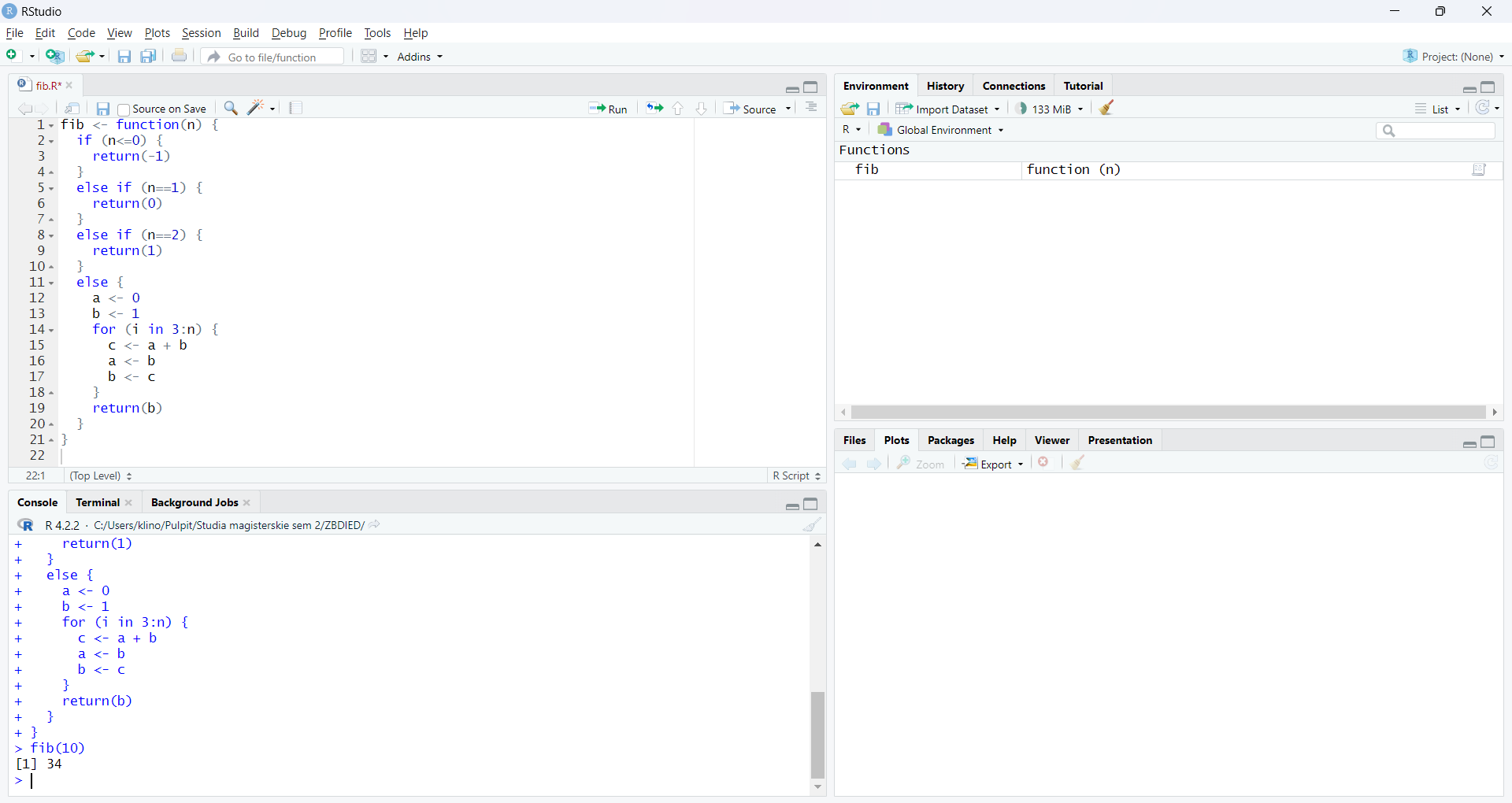
Pakiet R Commander – środowisko graficzne z poziomu R.



Rysunek . Ekran po instalacji i uruchomieniu środowiska graficznego R Commander.

# Ćw. 27.

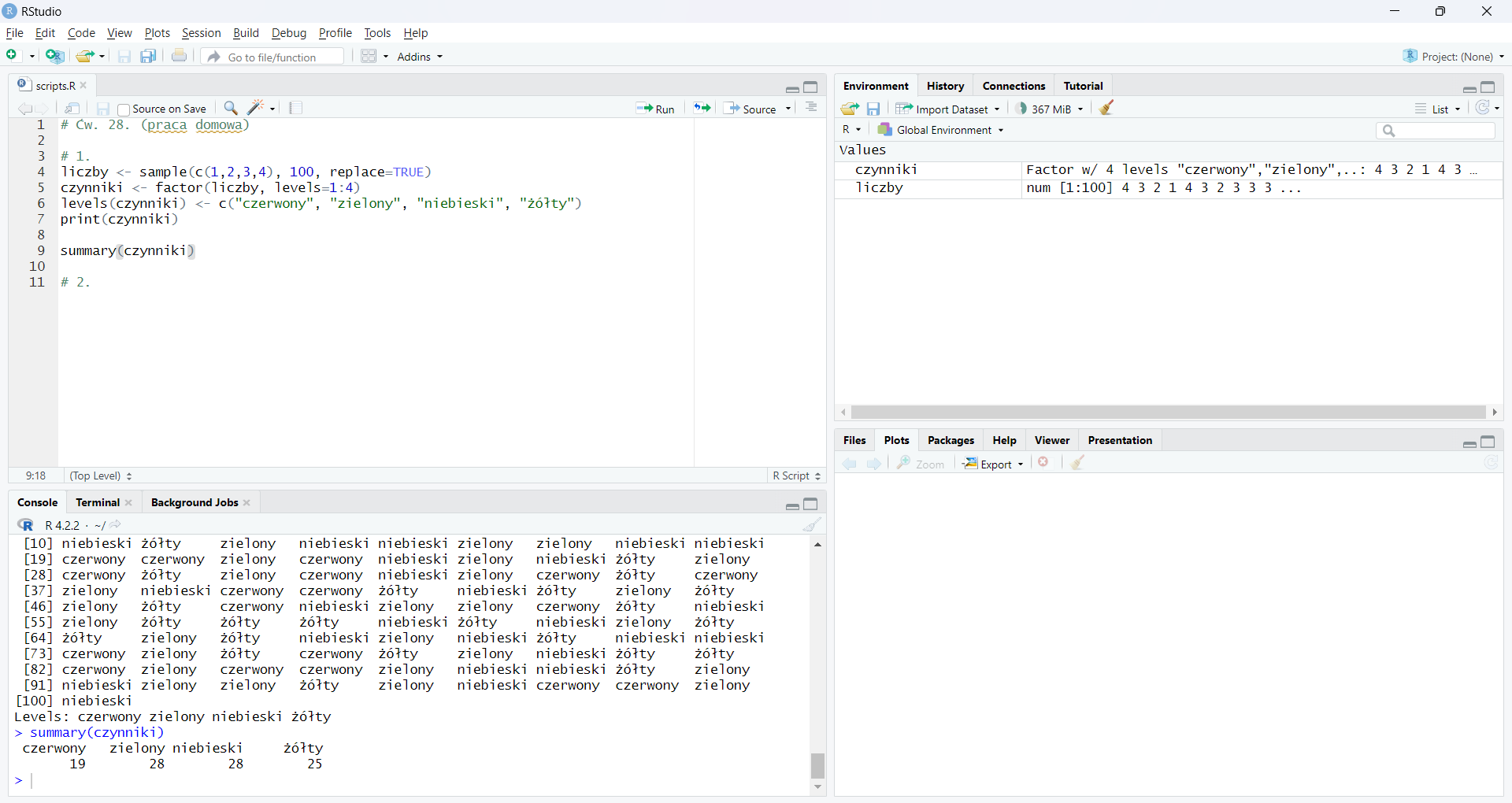
Środowisko RStudio.



Rysunek . Uruchomione środowisko RStudio z utworzonym wcześniej plikiem zawierającym funkcję z ciągiem Fibonacciego. Środowisko posiada podstawowe kolorowanie składni, konsolę oraz podgląd zmiennych w pamięci (w tym tabeli). Wyświetlają się w nim również utworzone wykresy.

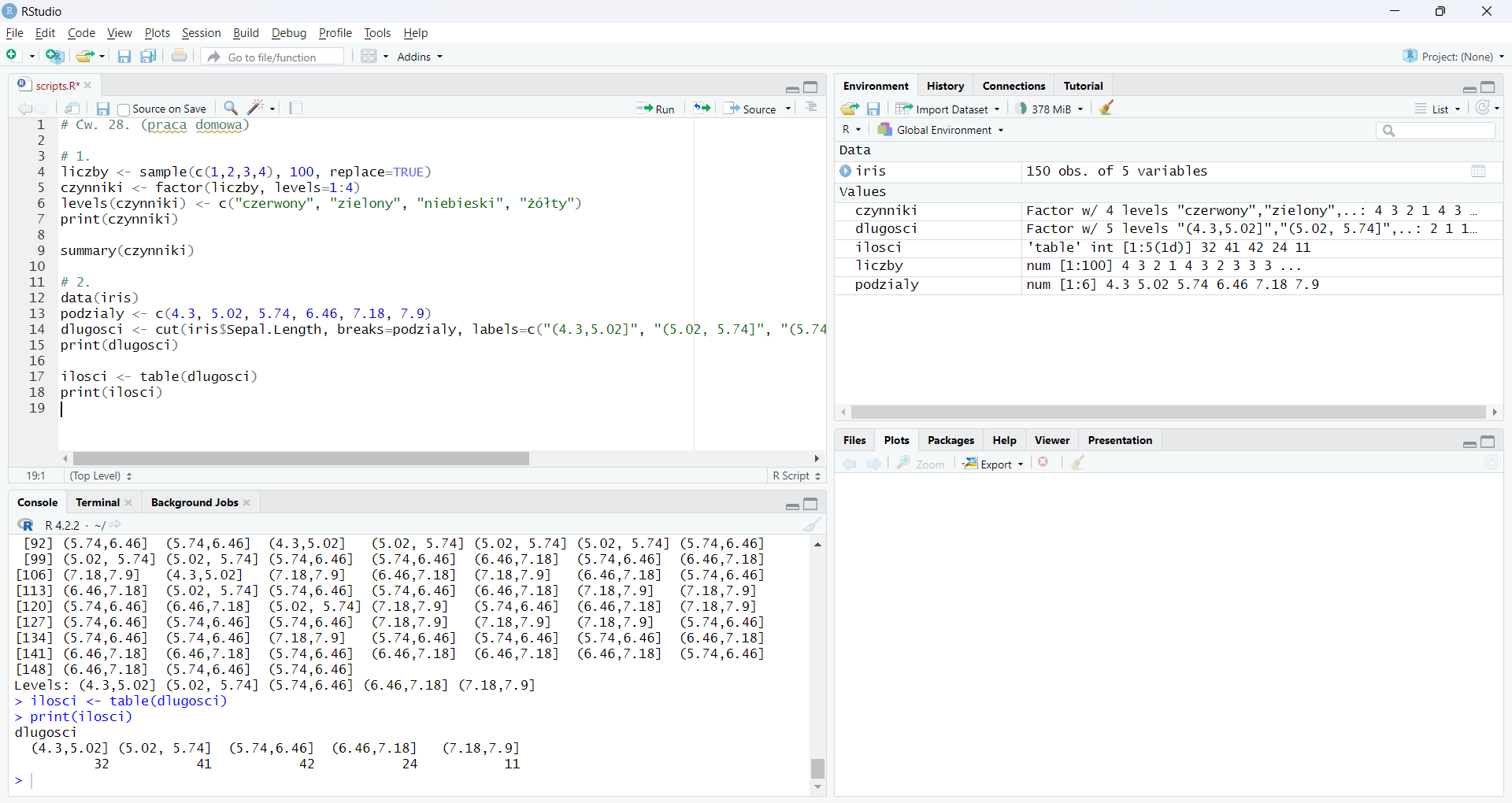
# Ćw. 28.

1. Utwórz wektor 100 liczb naturalnych…



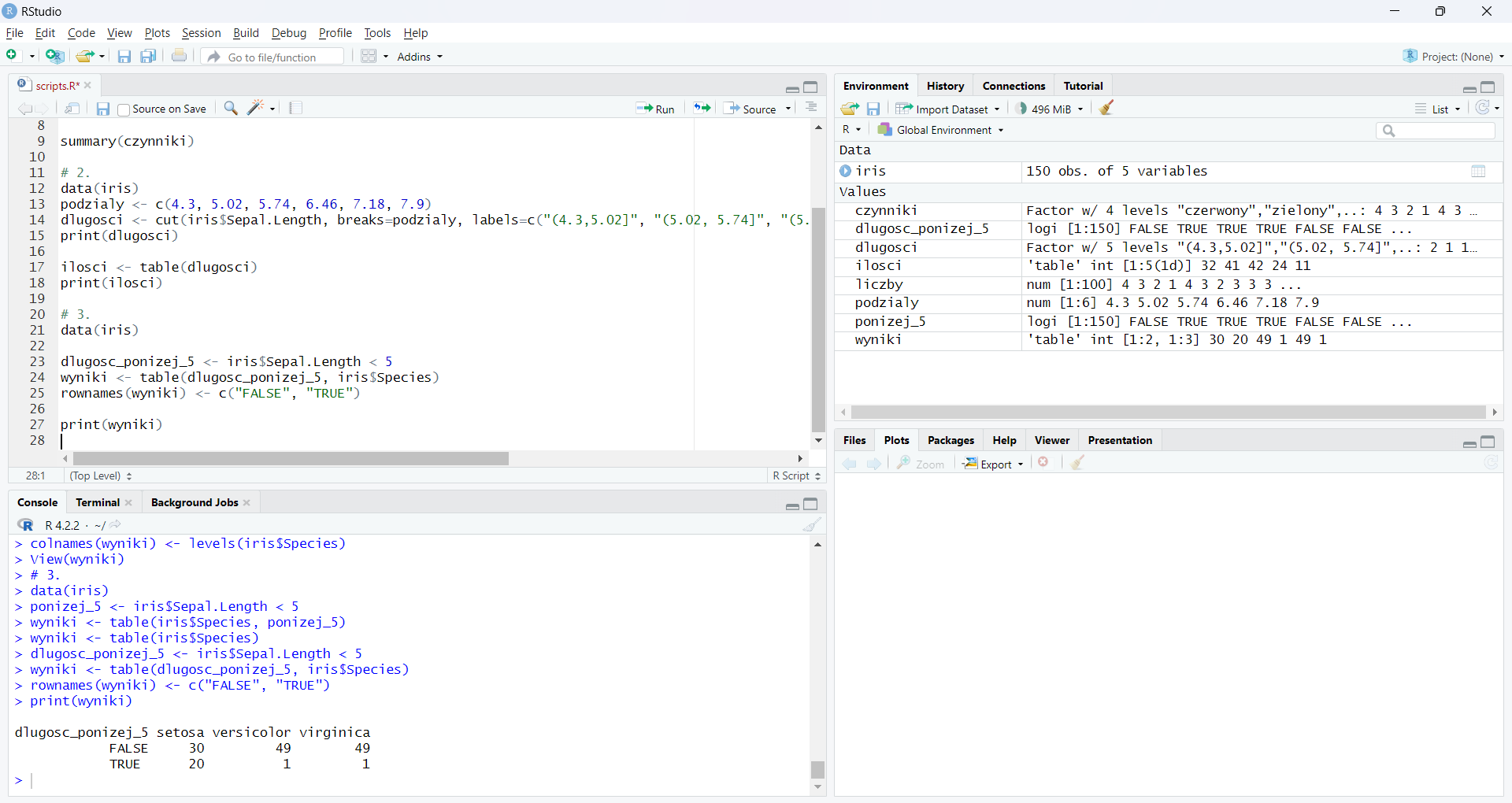
Rysunek . Ekran RStudio z kodem rozwiązującym ten podpunkt oraz wynikami działania.

1. Podziel Sepal.Length na 5 grup…



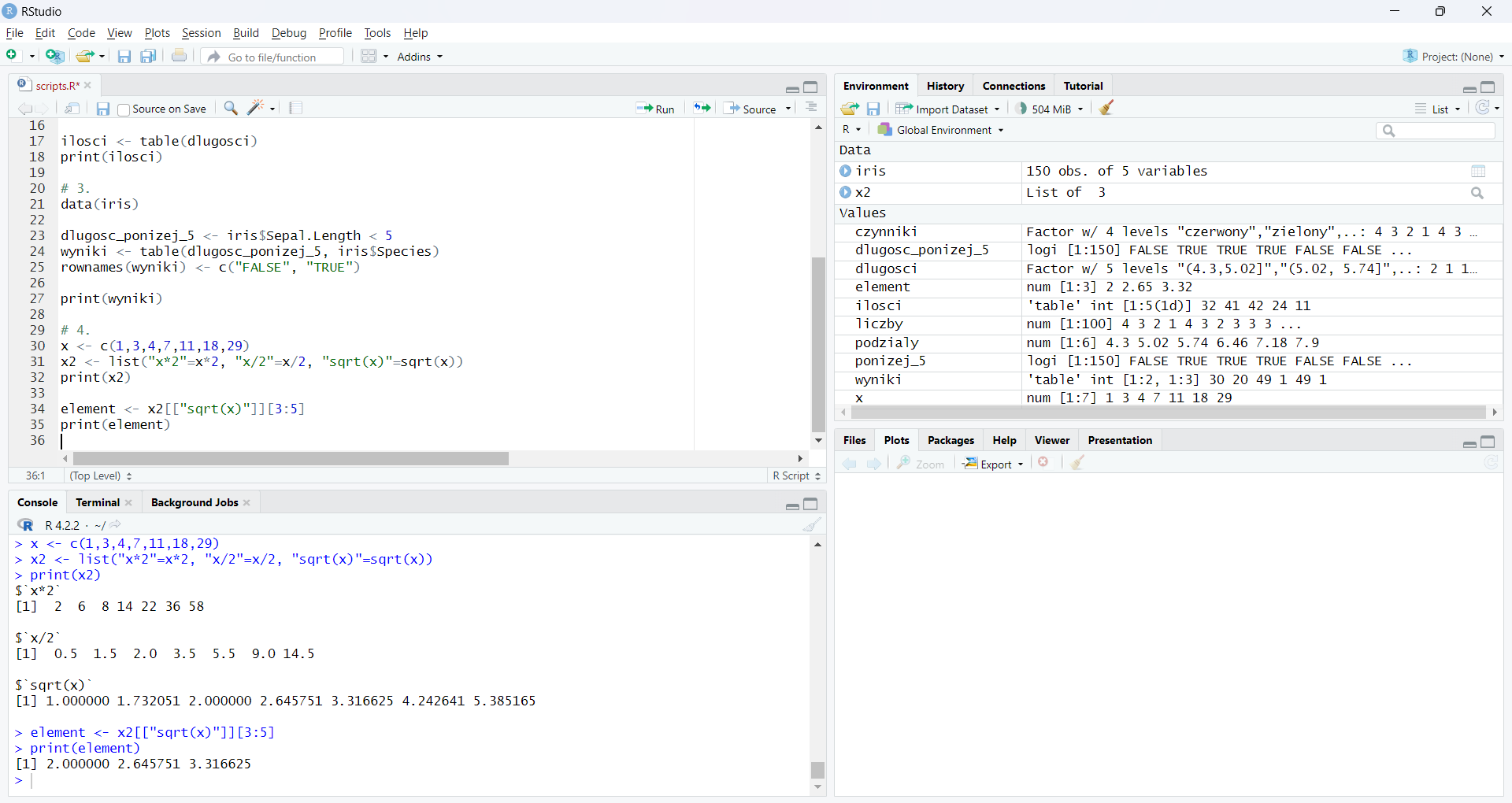
Rysunek . Kod źródłowy i wyniki działania kodu wykonującego to polecenie.

1. Podaj częstość występowania Sepal.Length…



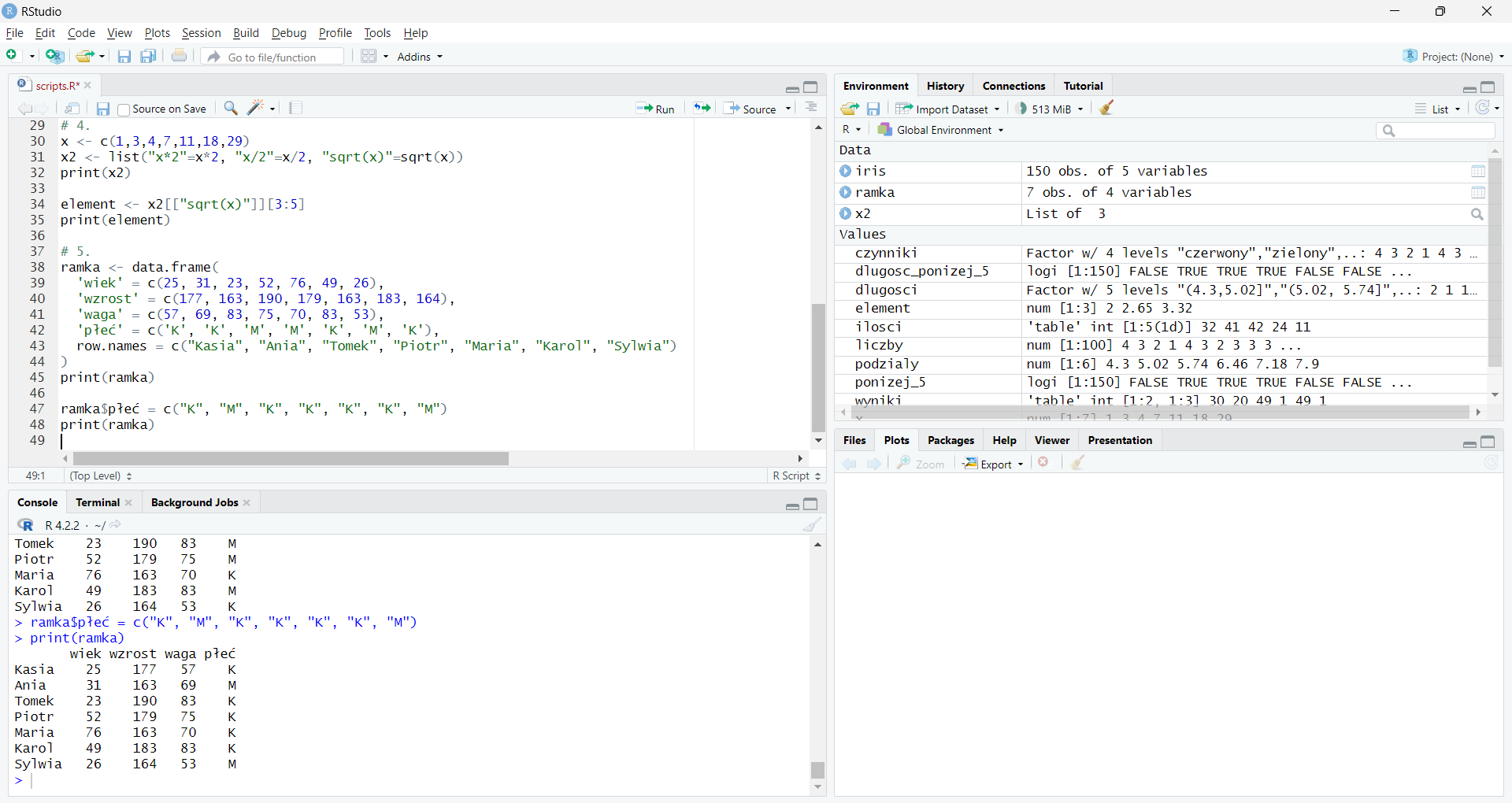
Rysunek . Kod tworzący tabelę o dwóch wierszach – TRUE i FALSE – oraz kolumnach odpowiadających gatunkom kwiatów.

1. Utwórz listę dla wektora…



Rysunek . Utworzenie listy z nazwanymi elementami oraz wyświetlenie trzech kolejnych wartości pierwiastka z x zgodnie z instrukcją.

1. Utwórz data.frame…



Rysunek . Utworzenie data.frame zgodnie z instrukcją i zmiana kolumny z płcią.

1. Zbiór danych swiss…



Rysunek . Kod źródłowy rozwiązujący polecenie 6.

Poniżej można znaleźć cały kod źródłowy napisany do wykonania Ćwiczenia 28.



# Wnioski.

Instalacja pakietu Rattle była wyzwaniem – w przypadku użycia nowszej wersji R występowały problemy z zależnością Rattle „RGtk2”. Dopiero po zainstalowaniu wersji R 4.1.3 i zainstalowaniu zgodnie z instrukcją na stronie pakietu, było możliwe poprawne uruchomienie Rattle.

Podczas laboratorium zapoznałem się z podstawami języka R, który jest w dużym stopniu zorientowany wokół danych – wektorów, list, ramek danych, itp. Standard języka zawiera sporo funkcjonalności do manipulacji danymi, i to jest największym plusem tego środowiska. Dodawanie nowych elementów do wektora, kolumn czy wierszy do ramki danych i tym podobne operacje są niezwykle proste i intuicyjne, a wiele wbudowanych operatorów działa nie tylko dla pojedynczych wartości, ale całych kolekcji danych. W kontekście w jakim używamy R, to takie operacje są bardziej istotne i częściej wykorzystywane niż standardowe instrukcje programistyczne.

W niektórych częściach ćwiczenia laboratoryjnego wykorzystano RStudio, będące najpopularniejszym środowiskiem programistycznym dla języka R. Zawiera ono podstawowe funkcje takie jak: podpowiadanie nazw, kolorowanie składni czy łatwy dostęp do środowiska roboczego, posiada wbudowany podgląd zmiennych w pamięci, wyświetlanie utworzonych wykresów i, poza obsługą plików źródłowych, konsolę R.