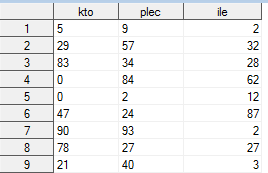
**data** a;

infile 'C:\Users\Marta\Desktop\Marta\studia\rok5\SAS\5\lab05\p.txt';

input kto $ plec $ ile;

**run**;



\*Uwaga ogolna: zasadniczo INPUT dziala podobnie do instrukcji SET - kazde wykonanie instrukcji INPUT czyta, o ile to mozliwe, pierwsza nieczytana linijke z pliku wejsciowego. Innymi slowy, kolejne INPUTy domyslnie przechodza do nowych linijek. Kazdy INPUT domyslnie czyta cala linijke, az do znalezienia znaku konca linii EOL;

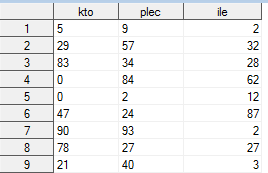
filename plik 'C:\Users\Marta\Desktop\Marta\studia\rok5\SAS\5\lab05\p.txt';

**data** a;

infile plik;

input kto $ plec $ ile;

**run**;



**data** a;

input kto $ plec $ ile;

cards;

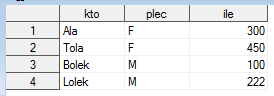
Ala F 300

Tola F 450

Bolek M 100

Lolek M 222

**run**;



**data** a;

infile datalines; \*ta linia jest opcjonalna, ale moze sie przydac, gdy bedziemy zmuszeni do uzycia instrukcji INFILE, patrz pozniej;

input kto $ plec $ ile;

cards;

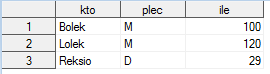
Bolek M 100

Lolek M 120

Reksio D 29

; /\* liczba spacji oddzielajacych elementy pliku dowolna, zalozenie: co najmniej jedna spacja \*/

**run**;



**data** a;

infile datalines missover;

input kto $ plec $ ile;

cards;

Bolek M 100

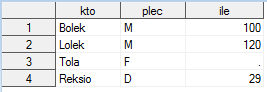
Lolek M 120

Tola F

Reksio D 29

;

**run**;



**data** a;

infile datalines missover;

input kto $ plec $ ile;

cards;

Bolek M 100

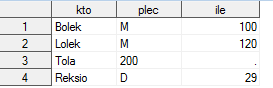
Lolek M 120

Tola 200

Reksio D 29

;

**run**;



**data** a;

input kto $ plec $ ile;

cards;

Bolek M 100

KrolikBugs M 100

; /\*tu trzeba najpierw zadeklarowac dlugosc, bo Krolik przekracza 8 bajtow\*/

**run**;



**data** a;

length kto $ **30**;

input kto $ plec $ ile;

cards;

Bolek M 100

KrolikBugs M 100

;

**run**;



**data** a;

input kto $ **1**-**6** plec $ **7** ile **8**-**10**;

cards;

Bolek\*M100

Lolek\*M120

Tola\*\*F222

ReksioD29

;

**run**;



**data** a;

input kto $ **1**-**6** plec $ **7** ile **8**-**10**;

cards;

Bolek\*M100

Lolek\*M120

Tola\*\* 200

ReksioD29

; /\*tu oczywiscie nie ma problemow z brakujaca plcia Toli \*/

**run**;



**data** a;

input kiedy;

cards;

2005/12/20

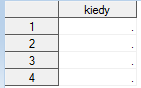
2003/09/08

2000/07/03

1997/11/30

;

**run**;



\* To sie oczywiscie nie moze udac w ten sposob - zmienna KIEDY jest numeryczna, a w jej

wartosciach pojawiaja sie zmaki / ;

\* Trzeba poinformowac SASa, jak powinien interpretowac wczytywane znaki;

\* Sluza do tego INFORMATY (w tym przypadku informat yymmdd10.);

**data** a;

input kiedy yymmdd10.;

cards;

2005/12/20

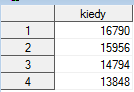
2003/09/08

2000/07/03

1997/11/30

;

**run**;



**data** a;

input kiedy yymmdd10.;

cards;

2005/12/20

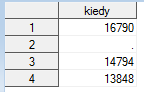
2003/09/08

2000/07/03

1997/11/30

;

**run**;



\*To sie wysypie, bo SAS czyta dokladnie 10 znakow (tyle, ile zadeklarowano w dlugosci informatu), w tym poczatkowa spacje.;

**data** f;

input kto $6. +**2** ile comma10.2 +**2** kiedy yymmdd8.;

cards;

Bolek\*\*\*100,222.20\*\*60/01/02

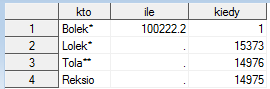
Lolek\*\*\*\*78,444.33\*\*02/02/02

Tola\*\*\*\*\*\*\*\*\*12.89\*\*01/01/01

Reksio\*\*\*\*\*\*\*\*1.00\*\*00/12/31

;

**run**;



**data** a;

input kto=$ ile=;

cards;

kto=Bolek ile=100

kto=Lolek ile=0

**run**;



**data** g;

input kto $ ile;

cards;

Jan Kowalski 100

Jan Nowak 200

;

**run**;



\*Tak sie oczywiscie wysypuje;

\*W takiej sytuacji pomaga ampersand: & (po zmiennej, ktorej dotyczy);

**data** g;

input kto $ & ile;

cards;

Jan Kowalski 100

Jan Nowak 200

;

**run**;



**data** h;

input pensja commax5.2 kto $;

cards;

12,11 Bolek

2,2 Lolek

;

**run**;



**data** h;

input pensja :commax5.2 kto $;

cards;

12,11 Bolek

2,2 Lolek

;

**run**;



/\*

Symbol :

Uzyty przed nazwa informatu powoduje, ze podczas czytania zmiennej uwzgledniane sa tylko

znaki do najblizszego separatora, a nie tyle, ile wynikaloby z dlugosci informatu.

\*/

**data** f;

infile cards;

input indyk $ x1 @;

if indyk='T' then input x2;

else input @**6** x3;

cards;

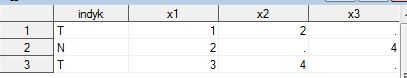
T 1 2 3

N 2 3 4

T 3 4 5

;

**run**;



\* @ na koncu linijki z instrukcja INPUT zatrzymuje rekord wejsciowy w buforze na czas wykonania danego obrotu petli glownej;

\* Uwaga: malpka w @6 ma inne znaczenie - jest to jeden z tzw. Column Pointer Controls, który przesuwa pointer do zadanej kolumny;

**data** i;

infile cards;

input litera $ x @@;

cards;

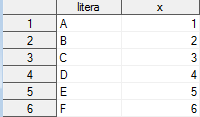
A 1 B 2

C 3

D 4 E 5 F 6

;

**run**;



\* @@ na koncu linijki z instrukcja INPUT zatrzymuja rekord wejsciowy w buforze takze pomiedzy obrotami petli glownej;

**data** j;

infile cards;

input kto $ x / y; \*to samo co dwa inputy;

cards;

Bolek 1

22.4

Lolek 2

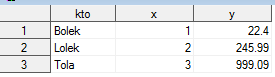
245.99

Tola 3

999.09

;

**run**;



\* Domyslnie INPUT czyta pojedyncza instrukcje. Istnieje jednak mozliwosc wczytywania kilku linijek pliku tekstowego naraz. Sluzy do tego opcja do instrukcji INFILE - N=

(dobrze znalezc i przeczytac odpowiedni fragment z dokumentacji. Przydatny jest w tym kontekscie jeden z Line Pointer Controls, #(liczba) - patrz informacja nt. Line Pointer Controls w opisie instrucji INPUT;