

# Інженерія знань

6 курс, осінь 2021

- Доц. Баклан І.В.
- Email: [iaa@ukr.net](mailto:iaa@ukr.net)
- Web: [baklaniv.at.ua](http://baklaniv.at.ua)

# Лекція 6

**Приклад створення бази  
знань у вигляді мережі  
фреймів**

# Приклад реалізації бази знань родинних відносин

Реалізувати базу знань для родинних відносин. Повинні відслідковуватися відносини: батько, мати, син, дочка, бабуся, дідусь, чоловік, дружина, онук, онучка, теща, свекруха. Передбачити автоматичне коректування всіх родинних зв'язків при занесенні одиничної інформації про нову людину.

# Реалізація на FRL

## Огляд функцій

| Синтаксис        | Семантика   |
|------------------|---|
| (PASSERT fn bd)  | Визначається або переопределяється процедура з ім'ям <b>fn</b> і тілом <b>bd</b> . Ім'я процедури заноситься в стек <b>*PROCEDURES*</b> . Результат - ім'я процедури. |
| (PASSERTQ fn bd) | Те ж що і <b>PASSERT</b> , але при зверненні аргументи не обчислюються.   |

**(PASSERTQ ADD () ...)**

Додавання одиної інформації в мережу фреймів із зазначенням імені, прізвища, віку, статі, імені матері і батька, імені чоловіка (дружини) персонажа. Передбачене автоматичне коректування всіх родинних зв'язків. Якщо будь-яка з полів порожньо, необхідно ввести символ \*.

**(PASSERTQ ADD ())**

; Блок введення інформації про людину

**(PRINC "PNAME: " ) (SETQ NAME**

**(READ INPUT) )**

```
(FINSTANTIATE FAMILY NAME)
(FPUT NAME PNAME $VALUE NAME)
(PRINC "PSERNAME: " ) (SETQ SERNAME
(READ INPUT) )
(FPUT NAME PSERNAME $VALUE SERNAME)
(PRINC "AGE: " ) (SETQ AGE
(READ INPUT) )
(FPUT NAME PAGE $VALUE AGE)
(PRINC "POL (M/F): " ) (SETQ POL
(READ INPUT) )
(FPUT NAME PPOL $VALUE POL)
(PRINC "FATHER: " ) (SETQ FATHER
(READ INPUT) )
; Додаємо в батьківський фрейм інформацію
про дитину
(COND ((NOT (EQUAL FATHER '*))
```

```

        (
            (FPUT      NAME      PFATHER      $VALUE
FATHER)
            (FPUT      FATHER    PCHILDREN    $VALUE
NAME)
        ))
    )
    (PRINC "MOTHER:          ") (SETQ MOTHER
(READ INPUT))
; Додаємо в материнський фрейм інформацію
про дитину
    (COND ((NOT (EQUAL MOTHER '*))
        (
            (FPUT      NAME      PMOTHER      $VALUE
MOTHER)
            (FPUT      MOTHER    PCHILDREN    $VALUE

```

```
        NAME)
    ))
)
; Додаємо в інформацію про чоловіка /
дружину
(COND ((EQUAL POL 'M)
      (
        (PRINC "WIFE: ")
      (SETQ WIFE (READ INPUT))
        ; Додаємо у фрейм дружини
інформацію про чоловіка
        (COND ((NOT (EQUAL WIFE
'*))
              (
                (FPUT NAME PWIFE
$VALUE WIFE)
```



```

                                (FPUT      WIFE
                                PHASBAND $VALUE NAME)
                                ))
                                )
                                )
                                )
                                (T
                                (
                                (PRINC  "HASBAND:      ")
(SETQ HASBAND  (READ INPUT))
                                ; Додаємо у фрейм чоловіка
інформацію про дружину
                                (COND ((NOT (EQUAL HASBAND
'*)))
                                (
                                (FPUT      NAME

```



```
(PASSERTQ GETFATHEROF (PERSON)
```

```
(COND
```

```
; Перевірка існування фрейму
```

```
( (NULL (MEMBER PERSON (CDR
```

```
*FRAMES*)))
```

```
ERROR!
```

```
_WRONG
```

```
_PERSO
```

```
N_NAME
```

```
!)
```

```
; Отримання імені батька персонажа
```

```
(T (CAR (FGET PERSON PFATHER)))
```

```
)
```

```
)
```

(PASSERTQ GETMOTHEROF (person) ...)

Визначення імені матері персонажа **person**.

(PASSERTQ GETMOTHEROF (PERSON)

(COND

; Перевірка існування фрейму

( (NULL (MEMBER PERSON (CDR  
\*FRAMES\*))) )

ERROR!

WRONG\_PERS  
ON\_NAME!)

; Отримання імені матері персонажа

(T (CAR (FGET PERSON PMOTHER)))

)

)



(PASSERTQ GETSONOF (person) ...)

Визначення імен всіх синів персонажа **person**.

(PASSERTQ GETSONOF (PERSON)

(COND

; Перевірка існування фрейму

((NULL (MEMBER PERSON (CDR

\*FRAMES\*)))

ERROR!

WRONG\_PERS  
ON\_NAME!

(T

(SETQ R '())

; Отримання списку всіх дітей

(SETQ S (FGET PERSON

PCHILDREN) )

(LOOP

; Повертаємо результат  
після досягнення кінця списку

((NULL S) (REVERSE R))

; Додаємо в результат  
тільки хлопчиків

(COND ((EQUAL (CAR (FGET

(CAR S) PPOL)) "M")

(SETQ R (CONS (CAR S)

R)))

)

; Отримуємо залишок списку

(SETQ S (CDR S))

)

)

)

)



```
(PASSERTQ GETDAUGHTEROF (person) ...)
```

Визначення імен всіх дочок персонажа **person**.

```
(PASSERTQ GETDAUGHTEROF (PERSON)
```

```
(COND
```

```
; Перевірка існування фрейму
```

```
( (NULL (MEMBER PERSON (CDR
```

```
*FRAMES*)))
```

```
ERROR!
```

```
_WRONG
```

```
_PERSO
```

```
N_NAME
```

```
!)
```

```
(T
```

```
(SETQ R ' ( ) )
```

```
PCCHILDREN) )
; Отримання списку всіх дітей
(SETQ S (FGET PERSON
(LOOP
; Повертаємо результат
після досягнення кінця списку
((NULL S) (REVERSE R))
; Додаємо в результат
тільки дівчаток
(COND ((EQUAL (CAR (FGET
(CAR S) PPOL)) "F"))
(SETQ R (CONS (CAR S)
R)))
)
; Отримуємо залишок списку
(SETQ S (CDR S))
```

)  
)  
)  
)

(PASSERTQ GETCHILDRENOF (person) ...)

Визначення імен всіх дітей персонажа **person**.

(PASSERTQ GETCHILDRENOF (PERSON)

(COND

; Перевірка існування фрейму

((NULL (MEMBER PERSON (CDR

\*FRAMES\*)))

ERROR!

WRONG\_PERS  
ON\_NAME!)

; Отримання списку всіх дітей

(T (FGET PERSON PCHILDREN))

)

)

```
(PASSERTQ GETGRANDFATHEROF
(person) ...)
```

Визначення імен всіх дідусів персонажа  
person.

```
(PASSERTQ GETGRANDFATHEROF (PERSON)
(COND
; Перевірка існування фрейму
( (NULL (MEMBER PERSON (CDR
*FRAMES*)))
ERROR!
_WRONG_PERS
ON_NAME!)
(T
(SETQ R ' ( ) )
```

; Отримання імені дідуся по  
материнській лінії

```
(SETQ R (CONS (CAR (FGET (CAR (FGET PERSON  
PMOTHER)) PFATHER)) R))
```

                  ; Отримання імені дідуся по  
батьківській лінії

```
(SETQ R (CONS (CAR (FGET (CAR (FGET PERSON  
PFATHER)) PFATHER)) R))
```

```
(REVERSE R)
```

```
)
```

```
)
```

```
)
```

```
(PASSERTQ  
(person) ...)
```

```
GETGRANDMOTHEROF
```

Визначення імен всіх бабусь персонажа  
**person.**

```
(PASSERTQ GETGRANDMOTHEROF (PERSON)
```

```
(COND
```

```
; Перевірка існування фрейму
```

```
((NULL (MEMBER PERSON (CDR
```

```
*FRAMES*)))
```

```
ERROR!  
_WRONG  
_PERSO  
_NAME  
!)
```

```
(T
      (SETQ R '())
      ; Отримання імені бабусі по
материнській лінії
      (SETQ R (CONS (CAR (FGET (CAR (FGET PERSON
PMOTHER)) PMOTHER)) R))
      ; Отримання імені бабусі по
батьківській лінії
      (SETQ R (CONS (CAR (FGET (CAR (FGET PERSON
PFATHER)) PMOTHER)) R))
      (REVERSE R)
    )
  )
)
```



```
(PASSERTQ GETGRANDPARENTSOF
(person) ...)
```

Визначення імен всіх прабатьків персонажа  
**person.**

```
(PASSERTQ GETGRANDPARENTSOF (PERSON)
(COND
; Перевірка існування фрейму
((NULL (MEMBER PERSON (CDR
*FRAMES*)))
```

```
ERROR!
WRONG_PERS
ON_NAME!)
```

```
(T
(SETQ R '())
; Отримання імені бабусі по
```

материнській лінії

```
(SETQ R (CONS (CAR (FGET (CAR (FGET PERSON  
PMOTHER)) PMOTHER)) R))
```

; Отримання імені дідуся по  
материнській лінії

```
(SETQ R (CONS (CAR (FGET (CAR (FGET PERSON  
PMOTHER)) PFATHER)) R))
```

; Отримання імені бабусі по  
батьківській лінії

```
(SETQ R (CONS (CAR (FGET (CAR (FGET PERSON  
PFATHER)) PMOTHER)) R))
```

; Отримання імені дідуся по  
батьківській лінії

```
(SETQ R (CONS (CAR (FGET (CAR (FGET PERSON  
PFATHER)) PFATHER)) R))
```

```
(REVERSE R)
```

```
)  
)  
)  
(PASSERTQ GETWIFEOF (person) ...)
```

Визначення імені дружини персонажа **person**.

```
(PASSERTQ GETWIFEOF (PERSON)  
  (COND  
    ; Перевірка існування фрейму  
    ( (NULL (MEMBER PERSON (CDR  
*FRAMES*))) ) )
```

**ERROR!**  
**WRONG\_PERS**  
**ON\_NAME!**

```
; Отримання імені дружини
```

(T (CAR (FGET PERSON PWIFE)))

)

)

(PASSERTQ GETHASBANDOF (person) ...)

Визначення імені чоловіка персонажа **person**.

(PASSERTQ GETHASBANDOF (PERSON)

(COND

; Перевірка існування фрейму

((NULL (MEMBER PERSON (CDR

\*FRAMES\*))

ERROR!

WRONG\_PERS  
ON\_NAME!)

; Отримання імені чоловіка

(T (CAR (FGET PERSON PHASBAND)))

)

)



```
(PASSERTQ GETGRANDSONOF (person) ...)
```

Визначення імен всіх онуків (чоловіків)  
персонажа **person**.

```
(PASSERTQ GETGRANDSONOF (PERSON)
```

```
(COND
```

```
; Перевірка існування фрейму
```

```
((NULL (MEMBER PERSON (CDR
```

```
*FRAMES*)))
```

```
ERROR!
```

```
WRONG_PERS  
ON_NAME!
```

```
(T
```

```
(SETQ R ' ()))
```

```
; Отримання списку всіх дітей
```

```

                (SETQ S (FGET PERSON
PCHILDREN) )
                (LOOP
                    ; Повертаємо результат
після досягнення кінця списку
                    ((NULL S) (REVERSE R) )
                    ; Отримання списку всіх
онуків
                    (SETQ S1 (FGET (CAR S)
PCHILDREN) )
                    (LOOP
                        ((NULL S1) T)
                        ; Додаємо в результат
тільки хлопчиків
                        (COND ((EQUAL (CAR
(FGET (CAR S1) PPOL)) "M")

```





```
(PASSERTQ GETGRANDDAUGHTEROF
(person) ...)
```

Визначення імен всіх онучок персонажа  
**person.**

```
(PASSERTQ GETGRANDDAUGHTEROF (PERSON)
(COND
; Перевірка існування фрейму
( (NULL (MEMBER PERSON (CDR
*FRAMES*)))
```

```
ERROR!
_WRONG
_PERSON
N_NAME
!)
```

```

(T
  (SETQ R '())
  ; Отримання списку всіх дітей
  (SETQ S (FGET PERSON
PCHILDREN) )
  (LOOP
    ; Повертаємо результат
    після досягнення кінця списку
    ((NULL S) (REVERSE R))
    ; Отримання списку всіх
    онуків
    (SETQ S1 (FGET (CAR S)
PCHILDREN) )
    (LOOP
      ((NULL S1) T)
      ; Додаємо в результат

```

тільки дівчаток

```
(COND ((EQUAL (CAR  
(FGET (CAR S1) PPOL)) "F")  
      (SETQ R (CONS  
(CAR S1) R)))  
      ; Отримуємо залишок  
      списку онуків  
(SETQ S1 (CDR S1))  
      )  
      ; Отримуємо залишок списку  
      дітей  
(SETQ S (CDR S))  
      )  
      )  
      )
```

)

```
(PASSERTQ GETGRANDCHILDRENOF
(person) ...)
```

Визначення імен всіх онуків персонажа **person**.

```
(PASSERTQ GETGRANDCHILDRENOF (PERSON)
(COND
; Перевірка існування фрейму
( (NULL (MEMBER PERSON (CDR
*FRAMES*)))
```

```
ERROR!
_WRONG_PERS
ON_NAME!)
```

```
(T
(SETQ R '())
; Отримання списку всіх дітей
```

```
(SETQ S (FGET PERSON
PCHILDREN) )
      (LOOP
        ; Повертаємо результат
після досягнення кінця списку
        ((NULL S) (REVERSE R))
        ; Отримання списку всіх
онуків
        (SETQ S1 (FGET (CAR S)
PCHILDREN) )
        (LOOP
          ((NULL S1) T)
          ; Додаємо в результат
всіх
          (SETQ R (CONS (CAR
S1) R))
```

```
                ; Отримуємо залишок  
                списку онуків  
                (SETQ S1 (CDR S1))  
            )  
        ; Отримуємо залишок списку  
        дітей  
        (SETQ S (CDR S))
```

```
)
```

```
)
```

```
)
```

```
)
```



(PASSERTQ GETTESHAOF (person) ...)

Визначення імен теці персонажа **person**.

(PASSERTQ GETTESHAOF (PERSON)

(COND

; Перевірка існування фрейму

((NULL (MEMBER PERSON (CDR

\*FRAMES\*)))

ERROR!

WRONG\_PERS  
ON\_NAME!

; Отримуємо ім'я теці

(T (CAR (FGET (CAR (FGET PERSON

PWIFE)) PMOTHER)))

)

)

(PASSERTQ GETSVEKROVOF (person) ...)

Визначення імен свекрухи персонажа **person**.

(PASSERTQ GETSVEKROVOF (PERSON)

(COND

; Перевірка існування фрейму

( (NULL (MEMBER PERSON (CDR

\*FRAMES\*)))

ERROR!

\_WRONG

\_PERSON

\_NAME

!)

; Отримуємо ім'я свекрухи

(T (CAR (FGET (CAR (FGET PERSON

PHASBAND) ) PMOTHER) ) )

)

)

## *Приклади використання*

Нехай маємо:

```
(DEFNAMEQ IVAN
  (PNAME ($VALUE (IVAN) ) )
  (PSERANAME ($VALUE (IVANCHENKO) ) )
  (PAGE ($VALUE (75) ) )
  (PPOL ($VALUE (M) ) )
  (PWIFE ($VALUE (MARIA) ) )
  (PCHILDREN ($VALUE (IGOR) (DARYNA)
(MYKOLA) ) )
)
```

(DEFNAMEQ MARIA

(PNAME (\$VALUE (MARIA) ) )

(PSENAME (\$VALUE (PETRENKO) ) )

(PAGE (\$VALUE (68) ) )

(PPOL (\$VALUE (F) ) )

(PHASBAND (\$VALUE (IVAN) ) )

(PCHILDREN (\$VALUE (IGOR) (DARYNA)

(MYKOLA) ) )

)

(DEFNAMEQ PETRO

(PNAME (\$VALUE (PETRO) ) )

(PSENAME (\$VALUE (SEMENCHENKO) ) )

(PAGE (\$VALUE (84) ) )

(PPOL (\$VALUE (M) ) )

(PWIFE (\$VALUE (KATERYNA) ) )

(PCHILDREN (\$VALUE (OLEG) (VERA) ) )

)

(DEFNAMEQ KATERYNA

(PNAME (\$VALUE (KATERYNA) ) )

(PSENAME (\$VALUE (GARMASH) ) )

(PAGE (\$VALUE (86) ) )

(PPOL (\$VALUE (F) ) )

(PHASBAND (\$VALUE (PETRO) ) )

(PCHILDREN (\$VALUE (OLEG) (VERA) ) )

)



(DEFNAMEQ IGOR

(PNAME (\$VALUE (IGOR) ) )

(PSENAME (\$VALUE (IVANCHENKO) ) )

(PAGE (\$VALUE (51) ) )

(PPOL (\$VALUE (M) ) )

(PMOTHER (\$VALUE (MARIA) ) )

(PFATHER (\$VALUE (IVAN) ) )

)

(DEFRAMEQ DARYNA

(PNAME (\$VALUE (DARYNA) ) )

(PSENAME (\$VALUE (IVANCHENKO) ) )

(PAGE (\$VALUE (48) ) )

(PPOL (\$VALUE (F) ) )

(PMOTHER (\$VALUE (MARIA) ) )

(PFATHER (\$VALUE (IVAN) ) )

)

(DEFNAMEQ MYKOLA

(PNAME (\$VALUE (MYKOLA) ) )  
(PSENAME (\$VALUE (IVANCHENKO) ) )  
(PAGE (\$VALUE (46) ) )  
(PPOL (\$VALUE (M) ) )  
(PMOTHER (\$VALUE (MARIA) ) )  
(PFATHER (\$VALUE (IVAN) ) )  
(PWIFE (\$VALUE (VERA) ) )  
(PCHILDREN (\$VALUE (JEVGEN) (OLYA) ) )

)

(DEFRAMEQ VERA

(PNAME (\$VALUE (VERA) ) )  
(PSENAME (\$VALUE (SEMENCHENKO) ) )  
(PAGE (\$VALUE (44) ) )  
(PPOL (\$VALUE (F) ) )  
(PMOTHER (\$VALUE (KATERYNA) ) )  
(PFATHER (\$VALUE (PETRO) ) )  
(PHASBAND (\$VALUE (MYKOLA) ) )  
(PCHILDREN (\$VALUE (JEVGEN) (OLYA) ) )

)

(DEFNAMEQ OLEG

(PNAME (\$VALUE (OLEG)))

(PSENAME (\$VALUE (GARMASH)))

(PAGE (\$VALUE (48)))

(PPOL (\$VALUE (M)))

(PMOTHER (\$VALUE (KATERYNA)))

(PFATHER (\$VALUE (PETRO)))

)

(DEFRAMEQ JEVGEN

(PNAME (\$VALUE (JEVGEN) ) )

(PSENAME (\$VALUE (SEMENCHENKO) ) )

(PAGE (\$VALUE (20) ) )

(PPOL (\$VALUE (M) ) )

(PMOTHER (\$VALUE (VERA) ) )

(PFATHER (\$VALUE (MYKOLA) ) )

)

(DEFRAMEQ OLYA

(PNAME (\$VALUE (OLYA) ) )

(PSENAME (\$VALUE (SEMENSHENKO) ) )

(PAGE (\$VALUE (14) ) )

(PPOL (\$VALUE (F) ) )

(PMOTHER (\$VALUE (VERA) ) )

(PFATHER (\$VALUE (MYKOLA) ) )

)

## Сделаем несколько запросов к исходной системе фреймов:

Мама Вери: (GETMOTHEROF VERA) —>  
KATERYNA

Папа Олега: (GETFATHEROF OLEG) —>  
PETRO

Сини Івана: (GETSONOF IVAN)  
—> (IGOR MYKOLA)

Дочери Катерини: (GETDAUGHTEROF KATERYNA)  
—> (VERA)

Діти Марії: (GETCHILDRENOF MARIA) —> (IGOR  
DARYNA MYKOLA)

Сини Євгена: (GETSONOF JEVGEN)



—> NIL

Бабусі Євгена: (GETGRANDMOTHEROF JEVGEN)

—> (KATERYNA MARIA)

Дедусі Олі: (GETGRANDFATHEROF OLYA) —>  
(PETRO IVAN)

Прабатьки Олі: (GETGRANDPARENTSOF OLYA) —>  
(KATERYNA PETRO MARIA IVAN)

Жена Івана: (GETWIFEOF IVAN)

—> MARIA

Чоловік Марії: (GETHUSBANDOF MARIA)

—> IVAN

Онук Петра: (GETGRANDSONOF PETRO)

—> (JEVGEN)

Онука Петра: (GETGRANDDAUGHTEROF PETRO) —>  
(OLYA)

Онуки (усі) Петра: (GETGRANDCHILDRENOF PETRO) —  
> (JEVGEN OLYA)

Теща Миколи: (GETTESHAOF MYKOLA)  
—> KATERYNA

Свекруха Вери: (GETSVEKROVOF VERA) —>  
MARIA

## Додамо членів родини:

(add)

NAME: ILYA  
SERNAME: SEMENCHENKO  
AGE: 22  
POL (M/F): M  
FATHER: MYKOLA  
MOTHER: VERA  
WIFE: \*

(add)

NAME: DARYNA  
SERNAME: SIDORENKO  
AGE: 23  
POL (M/F): F  
FATHER: \*

MOTHER:

\*

HASBAND:

ILYA

**Комбінуя запити, можемо отримувати відповіді на складні питання.**

Наприклад, визначимо вік тещі батька Олі:

**(FGET (GETTESHAOF (GETFATHEROF OLYA)) PAGE)**

**—> (86)**

**Наступна лекція буде присвячена  
прикладу створення бази знань на мові  
Prolog.**