**9.4. Объявление переменных: declare и typeset**

Инструкции **declare** и **typeset** являются [встроенными](http://www.opennet.ru/docs/RUS/bash_scripting_guide/c5358.html#BUILTINREF) инструкциями (они абсолютно идентичны друг другу и являются синонимами) и предназначена для наложения ограничений на переменные. Это очень слабая попытка контроля над типами, которая имеется во многих языках программирования. Инструкция **declare** появилась в Bash, начиная с версии 2. Кроме того, инструкция **typeset** может использоваться и в ksh-сценариях.

**ключи инструкций declare/typeset**

-r *readonly* (только для чтения)

declare -r var1

(**declare -r var1** аналогично объявлению **readonly var1**)

Это грубый эквивалент констант (const) в языке C. Попытка изменения таких переменных завершается сообщением об ошибке.

-i *integer*

declare -i number

# Сценарий интерпретирует переменную "number" как целое число.

number=3

echo "number = $number" # number = 3

number=three

echo "number = $number" # number = 0

# Строка "three" интерпретируется как целое число.

Примечательно, что допускается выполнение некоторых арифметических операций над переменными, объявленными как integer, не прибегая к инструкциям [expr](http://www.opennet.ru/docs/RUS/bash_scripting_guide/x6646.html" \l "EXPRREF) или [let](http://www.opennet.ru/docs/RUS/bash_scripting_guide/c5358.html" \l "LETREF).

-a *array*

declare -a indices

Переменная indices объявляется массивом.

-f *functions*

declare -f

Инструкция **declare -f**, без аргументов, приводит к выводу списка ранее объявленных функций в сценарии.

declare -f function\_name

Инструкция **declare -f function\_name** выводит имя функции function\_name, если она была объявлена ранее.

-x [export](http://www.opennet.ru/docs/RUS/bash_scripting_guide/c5358.html" \l "EXPORTREF)

declare -x var3

Эта инструкция объявляет переменную, как доступную для экспорта.

var=$value

declare -x var3=373

Инструкция **declare** допускает совмещение объявления и присваивания значения переменной одновременно.

**Пример 9-20. Объявление переменных с помощью инструкции declare**

#!/bin/bash

func1 ()

{

echo Это функция.

}

declare -f # Список функций, объявленных выше.

echo

declare -i var1 # var1 -- целочисленная переменная.

var1=2367

echo "переменная var1 объявлена как $var1"

var1=var1+1 # Допустимая арифметическая операция над целочисленными переменными.

echo "переменная var1 увеличена на 1 = $var1."

# Допустимая операция для целочисленных переменных

echo "Возможно ли записать дробное число 2367.1 в var1?"

var1=2367.1 # Сообщение об ошибке, переменная не изменяется.

echo "значение переменной var1 осталось прежним = $var1"

echo

declare -r var2=13.36 # инструкция 'declare' допускает установку свойств переменной

#+ и одновременно присваивать значение.

echo "var2 declared as $var2" # Допускается ли изменять значение readonly переменных?

var2=13.37 # Сообщение об ошибке и завершение работы сценария.

echo "значение переменной var2 осталось прежним $var2" # Эта строка никогда не будет выполнена.

exit 0 # Сценарий завершит работу выше.

|  |  |  |
| --- | --- | --- |
| [Назад](http://www.opennet.ru/docs/RUS/bash_scripting_guide/x4462.html) | [К началу](http://www.opennet.ru/docs/RUS/bash_scripting_guide/index.html) | [Вперед](http://www.opennet.ru/docs/RUS/bash_scripting_guide/x4788.html) |
| Подстановка параметров | [Наверх](http://www.opennet.ru/docs/RUS/bash_scripting_guide/c3270.html) | Косвенные ссылки на переменные |