CSC209 Week 3 Notes

Hyungmo Gu

May 11, 2020

Shell Programming 4 of 6

- I/O redirection
  - **<:** sends *stdin* from file to variable or command

```
>>> var=‘<sample.txt‘
>>> echo $var
hello!
hi!
it’s a pleasure to meet you

```

  - **>>:** appends line to file
  - **2>&1:** redirects *stderr* to the same file as *stdout*

```
>>> cat hello > output.txt 2>&1
>>> cat output.txt
cat: hello: No such file or directory

```

- File Descriptors
  - **0:** Standard Input
  - **1:** Standard Output
    - Channels all output

```
cat sample.txt
hello!
hi!
it’s a pleasure to meet you
>>> cat sample.txt> output.log
>>> cat sample.txt1> output.log # identical
>>> cat output.log
hello!
hi!
it’s a pleasure to meet you

```

  - **2:** Standard Error
    * Channels all error output

```
    >>> cat 2> error.log
```

- *$#*
  - Returns number of command line arguements

```
>>> cat arg_number_check_example.sh
if [ $# -ne 2 ]
then
    echo usage: arg_number_check_example.sh x y >& 2
    exit 1
fi

expr $1 + $2
>>> arg_number_check_example.sh 2
usage: arg_number_check_example.sh x y
```

- *$**
  - Means all command line arguements
  - all arguements passed are treated as one

```
>>> sh dollar_star_example.sh hello world hi
cat: hello world hi: No such file or directory
```

- *$@*
  - Also means all command line arguements
  - each arguement sparated a space is treated independently
  - Works like for loop

```
>>> sh dollar_at_example.sh hello world hi
cat: hello: No such file or directory
cat: world: No such file or directory
cat: hi: No such file or directory
```

- *${x}*
  - Works like template literal in javascript
  - Works with arguements as well!
  - Is useful when using in loop

```
>>> cat dolloar_curly_x_example.sh
a="hello"
cat ${a}2
>>> sh dolloar_curly_x_example.sh
cat: hello2: No such file or directory
```

```
>>> x = "hello"
>>> sed -n ${x}p file
sed: 1: "hellop": extra characters at the end of h command
```

Introduction to arrays in C 1 of 3

-