

• ~~terminal~~ Show application → terminal

wsl 100% bo 11

• In terminal →

* want to know where you are?

⇒ type `pwd` → enters . tadaaa
↓
point working directory

* How to navigate around this operating system?

⇒ * contents of that directory

⇒ `ls`.

* want to go inside of a directory (file)

⇒ `cd Pictures`.

* How do we go back?

⇒ `cd ..`

* How to create a directory?

⇒ `mkdir new1`

* How to create a file inside of a directory?

⇒ `cd new1` enters

⇒ `cat > a.txt` enters

⇒ `hellow` --- enter

⇒ now press control Z together and tadaaa it ends

* checking if the file has been created

⇒ `ls`

* How to manipulate the file?

copy a file to another

1) create a new file → `mkdir new2`

2) see if the file has been

1) check if the file has been created → `ls`

1) now copy → `cp`

where you want to copy
which file you want to copy

11 cd new

11 ls.

Q How to delete a file?

11 rm ~~a.txt~~

→ the file name you want to remove

11 ls [→ 11 cd.. ls] → backing for next process

Q How to move a file to another place

11 mv a.txt new2

↑ which file. ↑ where

11 ls.

11 cd new

11 ls.

Q How to remove a directory

11 rm ~~new~~ → new directory to be removed at woff *

↑ name of the directory

11 ls.

Q If a file already exist with the exact same name checking

11 mv -i source destination < too <

Q Replace the destination file with a new file

11 mv -u source destination.

Q hidden file/directory finding

11 ls -a. (Press control h to show)

Q Want to list files or find files based on the first alphabet

11 ls D.*

↑ that alphabet

11 ls D* (for two words)

Page of know how self notes

• `ls -l` → give ls with a lot more information
 access changing

type	user			group		others		<code>ls -l</code>
	r	w	x	r	-x	r	-x	
d								

↓
to owner
↓
to group
↓
to others

`ls -l`

r → read
w → write
x → execute

* Symbolic mode

• adding permission

{
 "chmod" → adding write permission for
 which file/group
 "ls -l"

removing permission {
 "chmod o-rw a.text" → removing read permission from
 others
 "ls -l"

over write → {
 "chmod g=r a.text"
 "ls -l"

multiple permission at a time

"chmod g=rwx a.text"

* absolute mode

r → 4 w → 2 , x = 1

 adds numbers for multiple add permission

"chmod 764 a.text"

764 →
 user (4+w+x) → 4+2+1
 group (4+2, r+w) → 4+2
 others (1) → 1

• Creating links

* Creating a soft link

ll ln -s a.text b.text
↑
name of original file. ↑
name of the shortcut file.

* Creating a hard link

ll ln a.text c.text

↑
hard link file of the a.text

shell scripting

=> text editors

=> creating directory : `mkdir newl` → name : `... .sh` → click save
(sh → shell, java → java, c → c, python → py)

* executing file. (saved desktop)

=> ~~test~~ /test.sh

=> `ls -l`

=> `chmod u+x test.sh`

=> ./test.sh

* creating a shell file using terminal.

=> nano hello-world.sh

=> echo "Hello world" (Printing)

=> Ctrl + X

=> Y

=> Enter

=> now executing

=> ./hello-world.sh

=> bash hello-world.sh. (alters function of chmod u+x text.sh)

→ full code → (when we use bash)

`#!/bin/bash`

`echo "Hello world"`

* How to declare variable

`#!/bin/bash`

`a=5`

`b="my name"`

`echo $b $a` → will print the value of b and a

* How to take input

#!/bin/bash

echo "Enter a name: "

read name.

(printing) echo "The name of the user is \$name"

* arithmetic calculation

#!/bin/bash

echo a=4

b=2

c=\$((a + b))

echo \$c

* How to do this a^b

#!/bin/bash

a=5

b=2

c=\$((a ** b))

echo \$c

* How to use if else condition

#!/bin/bash

echo "Enter a number: "

read a

echo "Enter another number: "

read b

if [\$a -gt \$b];

then echo "\$a is greater than \$b"

else if [\$a -lt \$b];

then echo "\$a is less than \$b"

gt → greater than

elif → else if

lt → less than

fi → ending of if

else

echo "\$a and \$b are equal"

fi

- * How to check a ~~exist~~ file or directory exist or not by the name which has given by the user

=> echo "Enter a name: "

read a

if [-d \$a]

then echo "It is a directory"

elif [-f \$a]

then echo "It is a file"

else

echo "not a file or directory"

fi

| checking directory -d.
" file -f.

* cases

echo "Enter a name: "

read name

case \$name in

"Abul") echo "The name is Abul";;

"Kashem") echo "The name is Kashem";;

*) echo "Some--";;

esac

* for loop

for ((i=0 ; i<10 ; i++))

do

echo \$i

done

* for each loop

```
my_list = (a b c d)
for i in $(my_list[*]); do
    echo $i
done
```

* while loop

```
i=0
while [ $i -lt 10 ]; do
    echo $i
    i=$((i+1))
done
```

* How to write function/method

```
function hello-world () {
    echo "Hello"
}
hello-world.
```

* you have to use **local** $a=5$ ~~for~~ inside a method if you don't want the value to be changed outside your method

* How to pass parameters to a method

```
adder () {
    echo $(( $1 + $2 ))
}
adder 3 4
```

How to return a value from a method/function

adder() {

 result = \$((\$1 + \$2))

 return \$result

}

adder 5 4

echo \$?

sd of p[0] of b[0][0]

called via shell

each command has its own

[env]

0 function

allowing local variable [sd]

How built-in vars

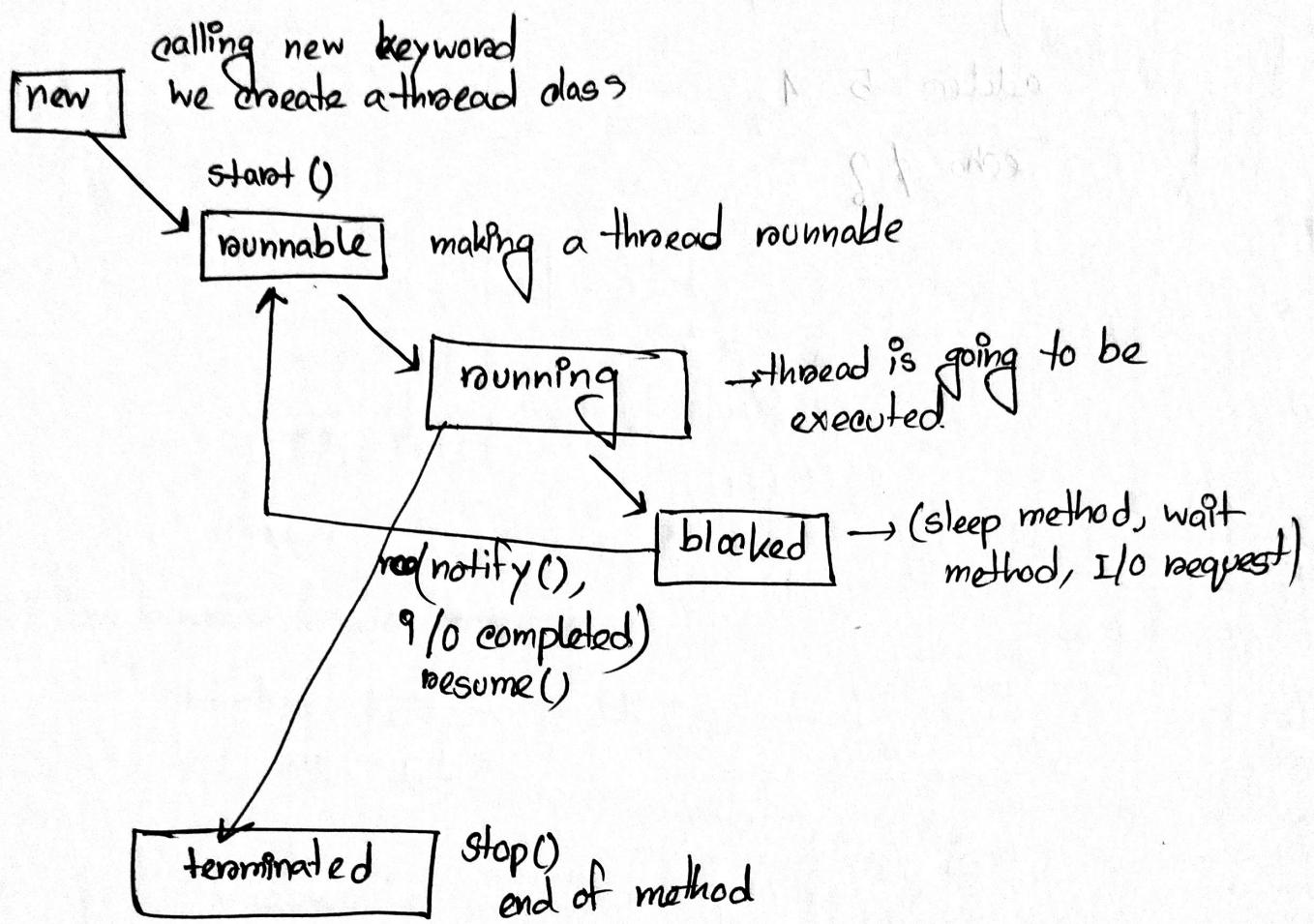
import of i b[0][0]

b[0][0] to b[0][0]

• Threading

Heap Space → static, class variables

Stack → instant variables, references



• public class ThreadTest {
 public static void main (String [] args) {
 S. O. SPrint ("");
 S. O. PIn (Thread. currentThread (). getName ());
 }
}

• class MyThread extends Thread {

@Override.

 public void run () {
 super.run ();
 }

public class ThreadTest {
 public static void main (String [] args) {
 MyThread myThread = new MyThread ();
 myThread.start ();
 }

• class MyThread (String extends Thread {

 public MyThread (String name) {
 super (name);
 }

 public void run () {

 S. O. PIn (Thread. currentThread (). getName ());
 }

public class ThreadTestof

public static void main(String[] args) {

MyThread myThread = new MyThread("My Thread 1");

myThread.start();

}

}

} basement shelves basementH cells

shelves()

} Oxia live sibling

Oxia spouse

{ }

} testbasement cells sibling

} (parent Ispade) non live sibling sibling

} (Observe) basement shelves basementH

} (tertite, basement) parent

{

} basement shelves parent) basementH cells

} (parent) sibling

(parent) spouse

{

} O mat live sibling

} (parent), Observe shelves, basement) n19 0.2

{ }